Research on a Novel Influence Maximization Algorithm Based on Community Structure

Jing Chen$^{1,2,3}$ and Jiangchuan Liu$^1$

$^1$ College of Information Science and Engineering, Yanshan University, Qinhuangdao 066004, China
$^2$ Key Laboratory for Computer Virtual Technology and System Integration of Hebei Province, Qinhuangdao 066004, China
$^3$ Software Engineering Laboratory in Hebei Province, Qinhuangdao 066004, China
Email: xychenjing@ysu.edu.cn

Abstract. With the research of influence maximization algorithm, many researchers have found that the existing algorithm has the problem of overlapping influence of seed nodes. In order to solve the problem of overlapping influence of seed nodes, this paper proposes an IMCS algorithm based on community structure. Firstly, we divide the community through the central node, and the quality of community division is ensured by defining community fitness and node contribution. Then through the analysis of the community division results, the seed node selects the one with the largest degree. Since most of the nodes activated by seed nodes of different communities also belong to different communities, this method solves the problem of overlapping influence to a certain extent. The experimental results show that the effectiveness of the IMCS algorithm is verified under the real network, cooperative network and artificial network, and the IMCS algorithm has a better effect than IEIR and Degree algorithms in most networks under the IC model.

Keyword. Seed node; influence maximization; community division; overlapping influence; community structure; independent cascade model.

1. Introduction

At present, social networks cover almost all forms of network services based on human social activities. Facebook, Twitter and Wechat have become important platforms for people to establish contacts and communicate with each other. With the help of social networks, how to quickly receive and publish information and accelerate the process of information diffusion is a hot research topic in social network analysis. Because social networks have relatively static snapshots of community structure in the process of formation and evolution, the results make the links between nodes in the community closer. In order to maximize influence propagation, this paper proposes an improved algorithm IMCS based on community structure. The algorithm IMCS aims at solving the overlap influence of seed nodes based on the results of community detection. By defining community fitness, contribution and community merger coefficient, the community is divided, and the seed nodes are selected based on community structure. Different seed nodes diffuse in different communities, applying the proposed algorithm IMCS in this paper, not only the scope of influence is greatly improved in small-scale social networks, but also the stable results of community partition are obtained. The paper is organized as follows:
In second part, we discussed and analyzed the related research work of influence maximization. In third part, the selection of seed nodes and the process of community partition are realized by defining the fitness and contribution of nodes, and the influence maximization algorithm IMCS is described and analyzed. In fourth part, the proposed algorithm IMCS is validated in real data sets and artificial data sets, and the advantages of the algorithm IMCS are verified by comparing and analyzing different algorithms. Finally, the full text is summarized, and the further research work is discussed.

2. Related Work

In recent years, a lot of research work has been done by domestic and foreign researchers on maximizing influence. At present, there are three main ways to solve the problem of maximizing influence: greedy algorithm, heuristic algorithm and community structure-based algorithm.

Richardson et al. used the network value of customers to carry out viral marketing. The experiments have proved the advantages of this method, and then raised the problem of maximizing influence, which was reduced to an algorithm problem [1, 2]. Kempe further proposed a greedy hill climbing algorithm to solve the problem of maximizing influence and proved that the problem is NP-hard [3]. Leskovec proposed the CELF algorithm, which greatly reduced the number of times to evaluate the seed influence range by mining the submodularity of the influence function and improved the time efficiency hundreds of times compared with the greedy algorithm [4]. Goyal made improvements based on the CELF algorithm and proposed the CELF ++ algorithm, which has a greater performance improvement than the CELF algorithm [5].

In addition to improving the greedy algorithm, many scholars have begun to use heuristic algorithms to improve efficiency. Chen et al. improved the traditional degree algorithm and proposed DegreeDiscount algorithm, which has a considerable effect compared with the traditional degree algorithm [6]. When a neighbor node of a node is selected as a seed node, there should be a certain discount when calculating the degree. Cao proposed CCA algorithm based on the characteristics of core number hierarchy and influence radius, which is more efficient than many other heuristic algorithms in the large IC model [7]. Huang proposed a folded Gibbs sampling algorithm training model, which is based on heuristic algorithm to discover influential nodes [8].

Maximizing influence is a very scalable problem. With the increasing amount of data on social networks, many researchers consider the community structure to solve the problem of maximizing influence. Tang et al. proposed a linear threshold model framework and a hybrid power greedy algorithm (HPG) in consideration of network structure and propagation characteristics, which is superior to the local optimal greedy algorithm [9]. Guo proposed an InfR algorithm and InfG model, which defined the mutual activation probability based on node edge attributes, that is, the measure of node influence, thereby mining its core node members and affecting coverage and the influence coverage rate is more than 90% [10]. Shang et al. proposed the CoFIM algorithm. Firstly, seed nodes are expanded between different communities, and then they are propagated between independent communities [11]. Han et al. explored the real change of the network by exploring some communities, and proposed a practical framework and an adjustment mechanism to enhance influence maximization [12]. Qiu proposed a hybrid algorithm PHG, which combines three stages. Different methods are used in different stages to find the node with the largest marginal effect [13]. Shen proposed a model which combines heuristic seed selection method and edge prediction method based on graph sparseness [14]. Li et al. proposed a community-based seed selection method, but the research object is a social network, which cannot correctly detect the community structure of other real networks [15]. Singh et al. proposed CoIM algorithm, which divides the network into many subnetwork, and then finds out the influential users from the subnetwork, so as to reduce the time complexity. They made a improvements based on CoIM algorithm, and proposed a C2IM algorithm which combines NDF technology and seed selection algorithm. They also focus on the balance between algorithm quality and time efficiency [16, 17].
3. Algorithmic Description and Implementation

3.1. Community Detection and Seed Node Selection

Reihaneh et al. proposed a community equation based on a central node in 2010 [18]. A community can be composed of a central node and its surrounding nodes, in a social network, the central node is a node with great influence. The degree can roughly evaluate the influence of the node, and it is generally considered that the node with large degree has a greater influence. Therefore, the node with a large degree is taken as the central node in this paper.

Equation (1): If $C_v$ is a local community, and the fitness of $C_v$ is:

$$M(C_v) = \frac{C^{in}_v}{C^{out}_v}$$

where $C^{in}_v$ is the inner edge of the community and $C^{out}_v$ is the outer edge of the community.

Equation (2): Community contribution of node $u$:

$$q(u) = \frac{u(C_v)}{D(u)}$$

where $u(C_v)$ is the number of edges of node $u$ and community $C_v$, and $D(u)$ is the degree of node $u$.

Equation (3): The set of neighbor nodes for community $C_v$:

$$Nei(C_v) = N(x_1) \cup N(x_2) \cup \ldots \cup N(x_n) - C_v$$

where $x_1, x_2, \ldots, x_n \in C_v$, $N(x_i)$ represent the neighbor of node $x_i$.

Given a network $G(V,E)$, where $V$ is the set of nodes and $E$ is the set of edges, make $D = \emptyset$, then the community detection steps based on the central node are described as follows:

Step 1: For each node $v \in V - D$, the node $v \in V$ with the highest degree of selection is used as the central node, the community in which it is located is $C_v$. The neighbor $u \in Nei(v)$ of $v$ is added to $u \in Nei(v)$, and the community fitness $M(C_v)$ of the community $C_v$ is calculated according to formula (1).

Step 2: For each node $k \in C_v$, mark $k$ as the node to be deleted by the community. Calculating the fitness $M'$ after deleting $k$ by the community, according to formula (1). If $M' > M(C_v)$, delete node $k$ from the community, otherwise it will not be deleted. The remaining nodes form community $C_v$. Repeat this step until no node is deleted from community $C_v$.

Step 3: Get the local community $C_v$ with node $v$ as the center node, and add the node in $C_v$ to set $D$.

Step 4: Find the neighbor set $Nei(C_v)$ of $C_v$ according to formula (3). For each node $u$ of neighbor set, calculate the contribution degree of $u$ node to $C_v$ community is calculated according to formula (2). If the contribution degree is greater than the given threshold $\theta$ (empirical value), add node $u$ to community $C_v$, and update $C_v$. Repeat this step until no nodes join the community.

Step 5: Add the nodes in $C_v$ to set $D$ and repeat steps 1 to 4 until all nodes belong to a community.

Take figure 1 as an example to illustrate the above execution process.
Firstly, node 1 with the largest degree of selection is taken as the central node, node 1 and its neighbor node 2, 3, 4, 5, 6 are taken as a community $C_1$, and the local fitness is calculated according to equation (1), $M(C_1) = 7/10$. Taking node 2 as an example, the fitness of the community after deleting it from community $C_1$ is calculated. Because of $M(C_1') > M(C_1)$, node 2 should be deleted from the community. There are no nodes in the remaining nodes that make the community fitness increase after deletion, so $\{1, 3, 4, 5, 6\}$ is obtained. The neighbor set $Nei(C_1) = \{2, 11, 12, 7, 8, 9, 10\}$ of $C_1$ is calculated according to equation (3), and the community contribution degree of each neighbor is calculated according to equation (2). If the threshold is set to 0.5, only 7, 8, 9 and 10 of $Nei(C_1)$ can be added to $C_1$, thus, the community $C_1 = \{1, 3, 4, 5, 6, 7, 8, 9, 10\}$ with node 1 as the central node is obtained. Finally, perform the fifth step above to find all the small communities.

Because small communities will be generated in the processing of the above-mentioned, the following community merger and optimization schemes are given in this paper.

Equation (4): Community consolidation factor $C(C_i, C_j)$:

$$C(C_i, C_j) = \frac{|C_i \cap C_j|}{\min(\text{len}(C_i), \text{len}(C_j))} + W \frac{M(C_i, C_j)}{\min(C_i^{\text{in}}, C_j^{\text{in}})}$$

(4)

where $C_i$ and $C_j$ are two communities respectively, $|C_i \cap C_j|$ indicates the number of common nodes in the two communities, $\min(\text{len}(C_i), \text{len}(C_j))$ indicates the number of nodes of the two communities with fewer nodes, $W$ is the balance coefficient. $M(C_i, C_j)$ indicates the number of connected links between the two communities except the common node, and $\min(C_i^{\text{in}}, C_j^{\text{in}})$ indicates the number of internal edges of the two communities with fewer internal edges. If two communities are greater than a given threshold $\omega$, then merge the two communities.

As shown in figure 2, if 1, 2, 3, and 3, 4, 5 are regarded as two communities respectively, the number of common nodes of the two communities is 1, and the number of nodes in the two communities is the same. Therefore, choosing any community with 3 nodes, the first value in formula (4) is 1/3. Except for common neighbors, the number of edges between two communities is 1, while the number of edges in the community is the same. When a community with 3 internal edges is randomly selected, the value of the second term in equation (4) is 1/3*0.8. Assuming that $w = 0.8$ (0.8 is an empirical value), the merging coefficient of the two communities is 1/3+0.8*1/3 = 0.597. Finally, the merging threshold is used to determine whether to merge the two communities.

Figure 1. The social network graph with 12 nodes.
Figure 2. Computation of merge coefficient of social networks.

For the communities divided in the previous section, the merging coefficient of the two communities is calculated. If the merging coefficient is greater than the given threshold, the merged communities will form new communities, delete the merged communities and update the community set. Repeat this step until no community can be merged. In order to maximize the influence, it is necessary to optimize the membership community of overlapping nodes and the community composed of a single node. The optimization strategy is to divide overlapping nodes or single nodes into communities where most neighbors belong. If the number of neighbors is the same, then random allocation can be done.

How to select seed nodes will affect the results of influence maximizing. Selecting the seed node based on the result of dividing the community in this paper. The method is described as follows: If there are more seed nodes than the number of communities, we select multiple nodes with the largest degree from each community, otherwise, on the basis of the above results of community merger and optimization, a node with the largest degree is selected for each community until the specified number of seed nodes are selected.

3.2. The Description of Influence Maximization Algorithm Based on Community Structure

Input: Social networking G=(V, E), \( \theta , k, \omega \);
Output: Seed set R.

Community division pseudocode:
1. While(len(Q)!=0) //Q is an empty set
2. for each \( v \in V - Q \)
3. Calculating the Degree \( D(v) \) of Node \( v \)
4. End for
5. Choosing the greatest degree node \( u=\{v|\text{max}(D(v))\} \) as the central node
6. Storing the neighbors of \( v \) and \( v \) into the set \( H[u] \) as a community
7. Calculating Community Fitness \( M(H[u]) \) of Community \( H[u] \)
8. Calculating community fitness \( M(H[u]-i) \) after removing node \( i \)
9. If(M(H[u]-i)>M(H[u]))
10. A.add(i)
11. Else If(A!={})
12. H[u].remove(i)
13. End If
14. While(True)
15. Calculating Neighbor Sets \( \text{Nei}(H[u]) \) of Community \( H[u] \)
16. Calculating the community contribution \( q(i) \) of node \( i \) to \( H[u] \)
17. If(q(i)>\( \theta \))
18. Adding nodes whose contribution is greater than a given threshold to set A1
19. Adding partitioned communities to collection R
20. End If
21. End While

**Community merge pseudocode:**

22. Calculating the merger coefficients of the two communities according to formula (4).
23. If $M(C_i, C_j) > \omega$ then merging the two communities
24. for i range(k)
25. for c in R
26. Choosing the community $c_{\text{max}} = \{c | \text{max(num(c))}\}$ with the largest number of nodes
27. Choosing the greatest degree nodes $u = \{u | \text{max(D(u))}\}$ in $c_{\text{max}}$, and adding to seed set S.
28. End For
29. End For

4. Experimental Results and Analysis

The main purpose is to verify two aspects in this paper. The first is to verify the quality of community partition. The second is to verify the impact of IMCS algorithm.

### 4.1. Verify the Validity of Community Partition

Three real social networks were used in the experiment: karate (Taekwondo Club), dolphins (Dolphin Network) and football (American University Rugby League Network). The specific information of the three social networks is shown in table 1.

| Name   | Number of nodes | Number of edges |
|--------|-----------------|-----------------|
| karate | 34              | 78              |
| dolphins | 62           | 159             |
| football | 115          | 613             |

The proposed algorithm is compared with GN and CDMM-LPA. In the experiment, modularity $Q$ and standardized mutual information NMI were used as the evaluation indicators of community detection, which were $= 0.4$, $= 0.5$, $= 0.8$. The comparison results of modularity $Q$ of the three algorithms are shown in table 2.

| Name   | GN   | CDMM-LPA | IMCS |
|--------|------|----------|------|
| karate | 0.360| 0.399    | 0.371|
| dolphins | 0.379| 0.490    | 0.378|
| football | 0.600| 0.581    | 0.596|

It can be seen from the results shown in Table 2 that the three algorithms have little difference in modularity, and most of the social networks have modularity partitioning results ranging from 0.3 to 0.7. Because the result of community partition with large modularity is not necessarily the best, there will be resolution problem if only modularity is used as evaluation index. Therefore, it is necessary to combine NMI to judge the validity of community detection. The results of NMI comparison of the three algorithms are shown in table 3.

| Name   | GN   | CDMM-LPA | IMCS |
|--------|------|----------|------|
| karate | 0.732| 0.617    | 0.837|
| dolphins | 0.704| 0.549    | 0.777|
| football | 0.880| 0.890    | 0.892|
It can be seen from the results shown in Table 3 that IMCS algorithm has obvious advantages over the other two algorithms. The reason is that in most networks, the influence of a node can only be transmitted to the second-level neighbors, but there are also close and loose differences between these nodes, and the probability that the nodes beyond the second-level neighbors will be affected is very small. Therefore, this paper focuses on choosing nodes with close ties within level 2 neighbors. The fitness of the neighbor nodes defined is used to select the first-level neighbor nodes which are more closely related to each other, and to select the nodes which are more closely related to the first-level neighbor through the community contribution of the nodes. Therefore, the selected nodes within the secondary neighborhood are closely related, and most of the nodes associated with evacuation are neglected. In summary, the experimental results prove the effectiveness of this method. The analysis of the experimental results shows that in most networks, the value of $\theta$ is between [0.4-0.45], while the best value is 0.5 for $\omega$, and 0.8 for $W$. In dolphins networks ($\omega=0.5, W=0.8$), Q is 0.489 when theta is 0.45, and Q is 0.378 when theta is 0.4. Therefore, the result of community division, Q value and NMI value change with the change of $\theta$ value. The smaller the value of $\theta$, the larger the size of the small community.

4.2. Verify the Effect of the Algorithm

Three artificial network datasets with 2000 nodes and four real cooperative network datasets (from http://snap.stanford.edu/data/) were used in the experiment. The specific information of the four real cooperative network datasets is shown in Table 4.

Table 4. Cooperative network information.

| Name  | Number of nodes | Number of edges | Average aggregation coefficient |
|-------|-----------------|-----------------|-------------------------------|
| GrQc  | 5242            | 14496           | 0.5296                        |
| HepTh | 9877            | 25998           | 0.4714                        |
| HepPh | 12008           | 118521          | 0.6115                        |
| CondMat | 23133         | 93497           | 0.6334                        |

In IC (independent cascade) model, the propagation probability $p = 0.06$ is set to compare the effects of IMCS, Degree and IRIE algorithms. Degree algorithm is to select the node with the largest degree in the network. IRIE algorithm is based on Influence Ranking Influence Estimation. It is the best comprehensive algorithm in the current influence maximization algorithm ($\theta = 0.45, \omega = 0.5, W=0.8, p=0.06$). The experimental results on artificial network datasets with parameters of $\mu = 0.1$, $\mu = 0.2$, $\mu = 0.3$ are shown in figures 3-5.

From figures 3 and 4, it can be seen that when the number of seed nodes reaches 50, the influence of IRIE and IMCS algorithm is better than that of Degree algorithm. In figure 3, the influence of IMCS algorithm is 7.43% and 31.40% higher than that of IRIE algorithm and Degree algorithm respectively, while the influence of IRIE algorithm is 18.24% higher than that of Degree algorithm.

In figure 4, Degree algorithm is better than IRIE algorithm before 45 nodes. IRIE algorithm is better than Degree algorithm when the number of seed nodes exceeds 45. When the number of seed nodes reaches 50, IMCS has the best influence, which is 7.96% and 12.05% higher than IRIE algorithm and Degree algorithm respectively, while the influence of IRIE algorithm is 3.79% higher than that of Degree algorithm. In figure 5, the influence of IMCS algorithm is 10.85% and 8.03% higher than that of IRIE and Degree algorithm respectively, while the effect of Degree algorithm is 2.60% higher than that of IRIE algorithm. In addition, with the increase of the number of seed nodes, the influence of IRIE algorithm is always lower than that of Degree algorithm, but the gap decreases gradually, and the influence of 50 nodes may be better than that of Degree algorithm.
The experimental results on GrQc, HepPh, HepTh and CondMat are shown in figures 6-9. In CondMat network, the IMCS algorithm is slightly worse than the IEIR algorithm, and the IMCS algorithm has the best impact results in other networks.

The influence of IMCS algorithm in figure 6 is 5.14% and 155.77% higher than that of IRIE algorithm and Degree algorithm, respectively. In figure 7, the influence of IMCS algorithm is almost the same as that of IRIE algorithm, which is 7.03% higher than that of Degree algorithm. From figures 6 and 7, we can know that the influence of Degree algorithm is basically unchanged when the number of seed nodes is more than 5, which shows that most of the nodes in the network tend to cooperate with the nodes with high degree.

In figure 8, the effect of IMCS algorithm is 12.55% and 3.39% higher than that of IRIE and Degree algorithm respectively, and the influence of Degree algorithm is 8.86% higher than that of IRIE algorithm. This shows that the moderate nodes in the network do not cooperate with the high-degree nodes only, and the influence overlap between the high-degree nodes is small. When \( P = 0.06 \), Degree has some advantages. In figure 9, the influence of IMCS algorithm is basically the same as that of IRIE algorithm, which is about 6.48% higher than that of Degree algorithm.
As the experimental results show that IMCS algorithm has a better influence than IRIE and Degree algorithm in most networks. The reason is that the nodes in the community are usually very closely linked, and the nodes are easy to be activated. These activated nodes are relatively not easy to be activated by the nodes in other communities, so they can be solved to a large extent. The problem of overlapping influence has been solved and the scope of influence has been improved. Because the time complexity of IMCS algorithm is relatively high, the algorithm has high application value in small-scale undirected weightless networks.

5. Conclusions
In this paper, we proposed an IMCS algorithm based on non-overlapping community structure, which applies a non-overlapping community discovery method based on central nodes to the problem of influence maximization. Through the community division of the selection of the central node, each community is effectively excavated, and the seed nodes are selected within the divided communities to maximize the influence. The validity of the IMCS algorithm is verified by experiments with real and artificial network data. Under the IC model, the IMCS algorithm in most networks has a better effect than other traditional algorithms. However, on some networks, the IMCS algorithm takes a long time to run. The algorithm in this paper still has many areas for improvement and further research. For example, the IMCS algorithm is only suitable for small-scale undirected and unweighted networks, and has a high time complexity. The next step will be based on how to reduce the time complexity of
the algorithm and apply it to large-scale directed and weighted network. In addition, applying influence maximization algorithms in dynamic community discovery is also a challenge.

References
[1] Domingos P and Richardson M 2001 Mining the network value of customers Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining pp 57-66.
[2] Richardson M and Domingos P 2002 Mining knowledge-sharing sites for viral marketing Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining pp 61-70.
[3] Kempe D, Kleinberg J and Tardos É 2003 Maximizing the spread of influence through a social network Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 137-146.
[4] Leskovec J, Krause A, Guestrin C, et al. 2007 Cost-effective outbreak detection in networks Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining pp 420-429.
[5] Goyal A, Lu W and Lakshmanan L V S 2011 Celf++: Optimizing the greedy algorithm for influence maximization in social networks Proceedings of the 20th International Conference Companion on World Wide Web pp 47-48.
[6] Chen W, Wang Y and Yang S 2009 Efficient influence maximization in social networks Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining 199-208.
[7] Cao J X, Dong D, Xu S, et al. 2015 A k-core based algorithm for influence maximization in social networks Chinese Journal of Computers 38 (2) 238-248.
[8] Huang H, Shen H, Meng Z, et al. 2019 Community-based influence maximization for viral marketing Applied Intelligence 49 (6) 2137-2150.
[9] Tian J T, Wang Y T and Feng X J 2011 A new hybrid algorithm for influence maximization in social networks Jisuanji Xuebao (Chinese Journal of Computers) 34 (10) 1956-1965.
[10] Guo J S, Yang H B, Wu K, et al. 2013 Influence optimization model based on community structure Journal of Computer Applications 33 (09) 2436-2439.
[11] Shang J, Zhou S, Li X, et al. 2017 CoFIM: A community-based framework for influence maximization on large-scale networks Knowledge-Based Systems 117 88-100.
[12] Han M, Yan M, Cai Z, et al. 2017 Influence maximization by probing partial communities in dynamic online social networks Transactions on Emerging Telecommunications Technologies 28 (4) e3054.
[13] Qiu L, Jia W, Yu J, et al. 2019 PHG: A three-phase algorithm for influence maximization based on community structure IEEE Access 7 62511-62522.
[14] Shen X, Mao S and Chung F L 2019 Cross-network learning with fuzzy labels for seed selection and graph sparsification in influence maximization IEEE Transactions on Fuzzy Systems (99) 1-1.
[15] Li X, Cheng X, Su S, et al. 2018 Community-based seeds selection algorithm for location aware influence maximization Neurocomputing 275 1601-1613.
[16] Singh S S, Singh K, Kumar A, et al. 2018 CoIM: Community-based influence maximization in social networks International Conference on Advanced Informatics for Computing Research pp 440-453.
[17] Singh S S, Kumar A, Singh K, et al. 2019 C2IM: Community based context-aware influence maximization in social networks Physica A: Statistical Mechanics and Its Applications 514 796-818.
[18] Khorasgani R R, Chen J and Zaïane O R 2010 Top leaders community detection approach in information networks 4th SNA-KDD Workshop on Social Network Mining and Analysis.