Digital Communications and Networks xxx (xxxx) xxx

Contents lists available at ScienceDirect

Digital Communications and Networks

journal homepage: www.keaipublishing.com/dcan

Identifying influential nodes in social networks via community structure and influence distribution difference

Zufan Zhang [a,b,c], Xieliang Li [a,b,c], Chenquan Gan [a,b,c,*]

[a] School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China
[b] Chongqing Key Laboratory of Mobile Communications Technology, Chongqing, 400065, China
[c] Engineering Research Center of Mobile Communications, Ministry of Education, Chongqing, 400065, China

ARTICLE INFO

Keywords:
Social network
Community detection
Influence maximization
Network embedding
Influence distribution difference

ABSTRACT

This paper aims to effectively solve the problem of the influence maximization in social networks. For this purpose, an influence maximization method that can identify influential nodes via the community structure and the influence distribution difference is proposed. Firstly, the network embedding-based community detection approach is developed, by which the social network is divided into several high-quality communities. Secondly, the solution of influence maximization is composed of the candidate stage and the greedy stage. The candidate stage is to select candidate nodes from the interior and the boundary of each community using a heuristic algorithm, and the greedy stage is to determine seed nodes with the largest marginal influence increment from the candidate set through the sub-modular property-based Greedy algorithm. Finally, experimental results demonstrate the superiority of the proposed method compared with existing methods, from which one can further find that our work can achieve a good tradeoff between the influence spread and the running time.

1. Introduction

With the rapid development of network technology and popularization of the Internet, social applications, like WeChat, Weibo and Snapchat, gradually generate a large amount of network data [1]. Many practical applications need to mine valuable information from the network data. Since then, the study of Influence Maximization (IM) has become a heated topic concerning the social network, especially in recent years, because it plays a key role in a wide range of fields, such as rumor control [2], network monitoring [3], information recommendation [4], and so on. Formally, the goal of IM is to identify the most influential network seed nodes as soon as possible and maximize the influence through the interaction between the seed nodes and other nodes simultaneously. Unfortunately, with the rapid growth of network scale, how to effectively solve the IM problem in reality, especially for the large-scale social networks, is very challenging [5].

Inspired by the "viral marketing" problem, the authors of Ref. [6] firstly collated and summarized the IM work. On this basis, Ref. [7] claimed that the IM issue is a non-deterministic problem with polynomial complexity, which is Non-deterministic Polynomial hard (NP-hard). Specifically, the authors proposed the Greedy algorithm to search seed nodes and proved that the optimal solution of IM could be approximated in the range of a factor. However, one disadvantage of the Greedy algorithm is that the time complexity is high because it must implement the Monte Carlo simulation [8] over the entire network to compute the marginal influence increment. To remedy this defect, Ref. [9] selectively updated the marginal impact increments of partial nodes utilizing the sub-modular characteristics of influence diffusion. Meanwhile, Ref. [10] presented the NewGreedy and MixGreedy algorithms, which further reduce the computational overhead by performing pruning on the network. However, the time complexity of the entire network-based algorithms is still too high to satisfy the requirements of timeliness of large-scale social networks [11]. Since the typical IM problem is NP-hard, heuristic algorithms, [12–15] attempted to directly pick up seed nodes according to some central measures, such as degree and PageRank [16]. But these central metrics are all related to the IM to some extent, ignoring the actual process of influence spreading under diffusion models. None of these algorithms provide theoretical assurance for the reliability of the results.

Real social networks, especially large-scale social networks, often have a distinct community structure in which nodes are closely connected. But it is noteworthy that the nodes in different communities are

* Corresponding author. School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China.
E-mail addresses: zhangzf@cqupt.edu.cn (Z. Zhang), xieliangli2016@163.com (X. Li), gcq2010cqu@163.com (C. Gan).

https://doi.org/10.1016/j.dcan.2020.04.011
Received 10 December 2019; Received in revised form 13 March 2020; Accepted 24 April 2020
Available online xxxx

often sparsely connected. Since communities are usually much smaller than the entire network, searching for seed nodes in each community can effectively reduce the computational overhead [17,18]. The CGA (Community-based Greedy Algorithm) [19] firstly divided the network into several communities and used the MixGreedy algorithm to find seed nodes in each community. However, the CGA ignores the bridge or hub nodes that connect different communities. Since bridge or hub nodes can spread influence among different communities, it is unreasonable to limit the search scope of these nodes in the community. For solving this problem, Ref. [20] selected seed nodes from the interior and the boundary of each community to reduce the loss of influence spread caused by community division. Nevertheless, in real social networks, the influence of nodes generally follows the power law distribution, and it is difficult for many nodes with limited influence to become seed nodes [21,22]. Hence, many unnecessary Monte Carlo simulation processes can be avoided by using the influence distribution characteristics of nodes, by which the efficiency of the algorithm will be improved while ensuring the influence spread.

To effectively solve the IM issue in real social networks, this paper identifies influential nodes via the community structure and the influence distribution difference, which can achieve a good tradeoff between the influence spread and the running time. The proposed method mainly consists of two parts: the network embedding-based community detection and the community structure-based IM. The former is to divide the social network into several high-quality communities. The latter is to divide the IM solution into a candidate stage and a greedy stage. In the candidate stage, for forming the candidate node set, the heuristic algorithm is used to select the highly influential nodes from the interior and the boundary of each community. In the greedy stage, the sub-modular property-based Greedy algorithm is used to determine the seed nodes with the largest marginal influence increment from the candidate node set. Finally, some experiments are performed to illustrate our superior performance.

The main contributions of our work are summarized as follows.

1) This paper proposes an IM method that can identify influential nodes via the community structure and the influence distribution difference.
2) The network embedding-based community detection approach is developed to divide the social network into several high-quality communities.
3) The community structure-based IM algorithm is designed to obtain the tradeoff between the influence spread and the running time.

The rest of this paper is organized as follows. Section 2 introduces the problem definition and preparation. The network embedding-based community detection and the community structure-based IM are described in detail in Sections 3 and 4, respectively. In Section 5, some experiments are performed and discussed. Finally, Section 6 concludes this work.

2. Problem definition and preliminaries

This section introduces the formal representation of the IM and some preliminaries, which will be used in the following sections.

2.1. Problem definition

Generally, let graph $G = (V,E)$ represent the social network, in which V and E denote individuals and the relationships between individuals, respectively. Let $S \in V$ be the subset of the initial seed nodes for influence spread, and $f(S)$ denote the influence spread of S, which is the expected number of nodes that are eventually influenced by S under a certain stochastic diffusion model. In particular, Fig. 1 gives an illustration of influence spread. According to the definition of influence spread, the IM problem can be formulated as follows.

Definition 1 (Influence Maximization). For the given network $G = (V,E)$, diffusion model and size of seed nodes k, the target of the IM is to discover the k-size subset of nodes $S \in V$ and guarantee that influence spread $f(S)$ is maximal under the diffusion model.

2.2. Preliminaries

To estimate the influence spread $f(S)$ of seed nodes S, the diffusion model should be determined at first. The most widely used diffusion models are the Linear Threshold (LT) model and the Independent

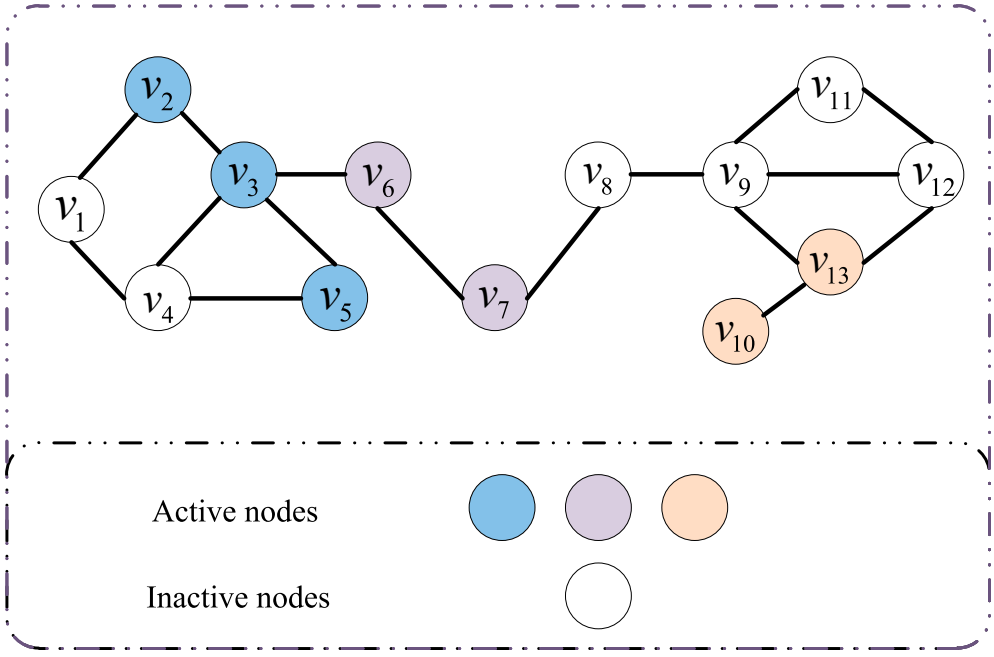

Fig. 1. An illustration of influence spread.

Cascade (IC) model, which represent different aspects of individual interaction. The former focuses directly on the threshold behavior in the influence spread, while the latter pays attention to the interaction and influence among individuals in social networks. This means that for the LT model, whether the individual is influenced depends on the overall results of all friends. Therefore, the LT model and the IC model are fundamentally different models.

IC model. Here, each edge $e = (u, v)$ is related to the probability that node v is activated by node u successfully. For the initial set of seed nodes S, the process of the influence spread is expanded in discrete timestamps with the following rules: At timestamp t, activated node u attempts to activate its inactive neighbor node v, then node v will become active at timestamp $t+1$ if it is activated successfully, otherwise, it will keep inactive. But node u has only one opportunity to activate node v, and this process will end until no more nodes are activated.

LT model. Here, weight b_{uv} denotes the influence of node u on node v. In addition, the following constraints should be satisfied:

$$\sum_{u \in N(v)} b_{u,v} \leq 1. \tag{1}$$

And each node v is also assigned with threshold θ_v, which indicates the time when node v is activated by its neighbor nodes. In general, the threshold θ_v is selected uniformly at random from 0 to 1. Similar to the IC model, the process of influence spread is expanded in the form of discrete timestamps. If the total weight satisfies the following threshold:

$$\sum_{u \in N(v)} b_{u,v} \geq \theta_v, \tag{2}$$

The status of nodes changes from "inactive" to "active". Since there is no clear definition to determine the threshold, the IC model is usually adopted to simulate the process of influence spread in most previous works.

Greedy algorithm. In either the IC or the LT model, the exact calculation of the optimal solution for the IM problem is NP-hard [7]. The goal of the Greedy algorithm is to iteratively choose the nodes with the maximal marginal influence increment as seed nodes. Let $f(v|S_i) = f(v \cup S_i) - f(S_i)$ denote the marginal influence increment of node v, where $f(S)$ is the influence spread of seed nodes S. Although the Greedy algorithm is conceptually simple, it requires significant computational overheads because Monte Carlo simulations are performed over the entire network to compute the marginal influence increment.

3. Network embedding-based community detection

In this paper, the network embedding-based community detection includes two stages: the community detection and the seed nodes selection. Firstly, the formal definitions of the community structure and the network embedding will be given. On this basis, the network embedding-based community detection will be introduced.

3.1. Community structure and network embedding

The community structure is one of the basic and important attributes of social networks since it can provide valuable insights into the dynamics of social networks. Furthermore, according to the work in Ref. [18], it can be defined as follows.

Definition 2 **(Community structure).** For a given network G, its community structure represents a group of nodes with higher edge density inside these nodes and lower edge density among groups. Modularity Q is often used to quantify the strength of the network community structure, and is defined as the fraction of edges, and its calculation formula is:

$$Q = \frac{1}{2|E|} \sigma(C_u, C_v) \sum_{u,v \in V} \left(e_{uv} - \frac{d_u d_v}{2|E|} \right), \tag{3}$$

where $|E|$ denotes the amounts of edges; $\sigma(C_u, C_v)$ means that if nodes u and v are in the same community, the value of the function is 1, otherwise its value is 0; d_u represents the degree of node u; e_{uv} refers to a direct edge between nodes u and v. In addition, modularity Q in the real social network is usually between 0.3 and 0.7. The larger the modularity is, the stronger the community structure of the social network.

The adjacency matrix is one of the intuitive representations for the network structure. However, for large-scale social networks, such a traditional network representation method usually makes the clustering task computationally expensive and intractable. The network embedding is able to extract low-dimensional and high-quality features, which is beneficial for the analysis of large-scale social networks. Formally, the network embedding is defined as follows.

Definition 3 (Network embedding). For a given network $G = (V,E)$, embed each node $v \in V$ into a low-dimensional representation space $\mathbb{Z}^d$, where $d \ll |V|$. This process is called network embedding. In $\mathbb{Z}^d$, the neighborhood relations among network nodes should be well preserved. Here, DeepWalk is used to learn the potential low-dimensional representation.

3.2. Community detection

As shown in Fig. 2, after obtaining the low-dimensional representation, the network is divided into several communities by means of the classic k-means algorithm [23], which is based on the following steps:

Step 1: Randomly select k clustering centers;

Step 2: Compute the similarity between each point and each center;

Step 3: Cluster the points where the similarity is less than the threshold;

Step 4: Update the cluster centers and repeat the second and third steps until the centers remain unchanged. As for how to get the value of k, the quality of community detection is measured by the modularity Q.

4. Community structure-based influence maximization

In this section, the IM based on the community structure and the influence distribution difference is first explored to enhance the time efficiency of the Greedy algorithm. Since the influence of social network

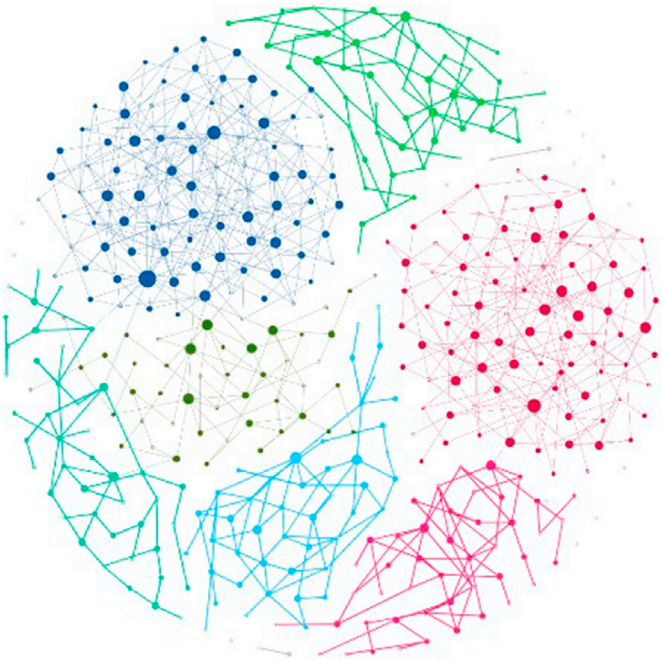

Fig. 2. Community detection.

nodes follows the power-law distribution, the IM solution can be deemed to have a candidate stage and a greedy stage. In the candidate stage, the heuristic algorithm is used to select the highly influential nodes from the interior and the boundary of each community to form the candidate node set. In the greedy stage, the sub-modular property-based Greedy algorithm is utilized to search seed nodes from the candidate set, where the influence spread scope of the candidate nodes is restricted to the communities of the nodes and their neighbors.

As shown in Fig. 3, social network G is divided into several non-overlapping communities. The first step is to solve the IM problem to determine the allocation strategy of the candidate nodes. As for how to determine the amount of high influential nodes from the candidate node set without losing influence spread, the proportion parameter is introduced for adjustment. Due to the positive correlation between the influence spread and the community size, the number of candidate nodes is allocated proportionally according to the community size. For a given network G with r communities, let C_s be the community division whose communities do not overlap each other, then

$$C_s = (C_1, C_2, \cdots, C_r). \tag{4}$$

Let U denote the set of candidate nodes. Then, there are two cases about the distribution of the number of U.

Case 1: If $r < \mid U \mid$, then the number of candidate nodes $|U_{C_i}|$ with respect to community C_i is calculated as:

$$|U_{C_i}| = \mu \times \frac{k \times |C_i|}{n}, i \in \{1, 2, \cdots, r\}, \tag{5}$$

where μ is the number of candidate nodes to be mined in different communities, n and k are the total number of nodes and seed nodes, respectively.

Case 2: If $r \geq \mid U \mid$, then the number of candidate nodes $|U_{C_i}|$ with respect to community C_i is:

$$|U_{C_i}| = \mu \times \frac{k \times |C_i|}{\sum_{l=1}^{q} |C_l|}, i \in \{1, 2, \cdots, q\}. \tag{6}$$

From the above analysis, U is proportional to the size of the internal and boundary nodes of each community.

4.1. Mining candidate nodes within the community

In the stage of mining the candidate nodes within the community, the LeaderRank algorithm, which is the improvement of PageRank [16] and can improve the convergence, is used to preliminarily calculate the influence of nodes within community. As depicted in Fig. 4, LeaderRank first assigns a unit of LR (LeaderRank value) to all nodes in the network except node g, and evenly distributes the LR to the neighbors of the nodes. From the viewpoint of mathematics, this process can be regarded as a random walk on a directed network with the stochastic matrix $\mathbf{P}$, in which $p_{ij} = \frac{e_{ij}}{k_i}$ is the probability of a random walker from i to j. Then the LRs of all nodes are updated depending on the following formula:

$$LR_i(t+1) = \sum_{j \in N_i} \frac{LR_j(t)}{k_j}, \tag{7}$$

where N_i represents the adjacent node set of node i, and k_j is the degree of node j, $LR_j(t)$ denotes the LR of node j at iteration t. The process iterates until it converges. Finally, the LR of the ground node is averagely assigned to other nodes, and the final value LR of node i is defined as:

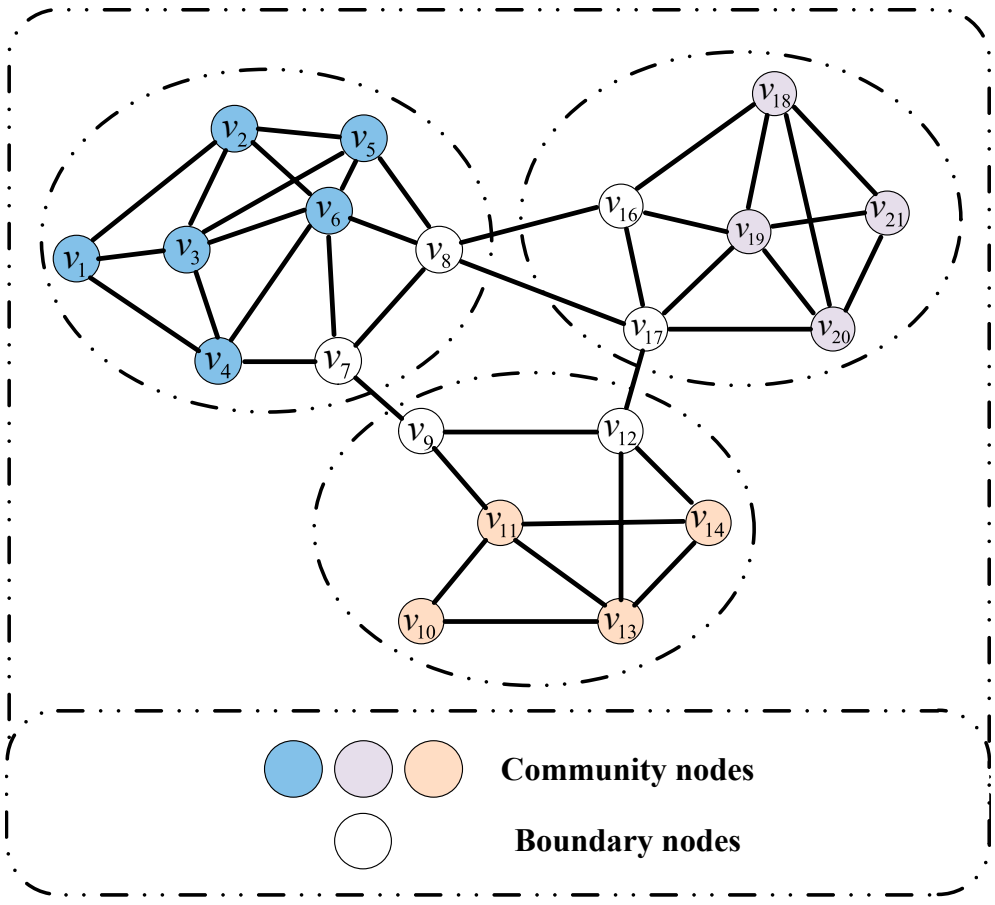

Fig. 3. Community structure.

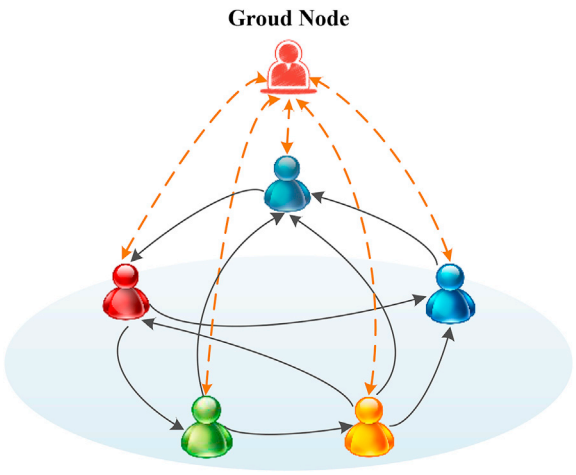

Fig. 4. LeaderRank.

$$LR_i = LR_i(t_c) + \frac{LR_g(t_c)}{N}, \tag{8}$$

where $LR_i(t_c)$ represents the LR value of node i at iteration t_c, $LR_g(t_c)$ represents the LR value of the ground node g at iteration t_c, and t_c is the point where the process iterates to convergence.

The above analysis implies that important nodes are more likely to get higher LR values rather than according to the degree of nodes. Therefore, the LeaderRank algorithm is used to preliminarily evaluate the influence of the interior nodes of the community. Then, according to the number allocation strategy of the candidate nodes, the nodes with high influence are selected from the interior of each community to form candidate nodes.

4.2. Mining candidate nodes within the boundary

As displayed in Fig. 5, the structural holes are the gaps among individuals which lack direct connections (e.g., node A-B and A-D). The position of the structural hole makes it serve as a bridge or "proxy" between other nodes. And it is well-known that the connection among communities is sparse, and structural holes with the advantage of weak ties are the key to the influence spread across communities. Hence, the structural holes are selected from the boundary of each community to form candidate nodes.

According to the definition of structural holes, the network constraint coefficient CT is used to measure the constraints imposed by forming the structural holes. And the calculation of CT is shown as:

$$CT_i = \sum_{j \in N(i)} \left(p_{ij} + \sum_q p_{iq} p_{qj} \right)^2, \tag{9}$$

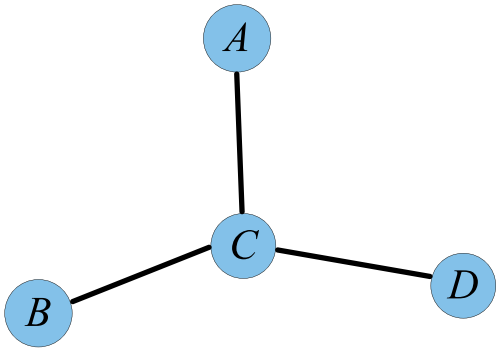

Fig. 5. The illustration of structural holes.

where node q is a common neighbor of nodes i and j, p_{ij} is the cost invested by maintaining the neighbor relationship in the direct way, and p_{iq} is the cost invested by maintaining neighbor relationship in the indirect way, as illustrated in Fig. 6. In the unweighted network, p_{ij} is usually equal to $\frac{e_{ij}}{\sum_{j \in N(i)} e_{ij}}$.

The hub nodes with many neighbor nodes are more likely to be structural holes. However, the network constraint coefficient only focuses on the network topology and ignores the community structure characteristics of large-scale social networks. Nodes in the social network of multiple communities can actually act as a "bridging" role for the communities, and information can be spread to most of the communities through these nodes. Hence, the network constraint coefficient combined with the community structure is defined as:

$$OC_v = \frac{10^{-CT_v} \odot Nb(v)}{\max OC}, \tag{10}$$

where CT_v and $Nb(v)$ are the network constraint coefficient and the neighbor set of node v, respectively; maxOC is the normalization factor. In the process of specific implementation, the value of OC is used to initially evaluate the potential influence of the boundary nodes. And then the boundary nodes with high influence are selected into the candidate node set according to the number allocation strategy of candidate nodes.

4.3. Community structure-based seed nodes selection

After selecting the candidate nodes from the interior and the boundary of communities, the sub-modular property-based Greedy algorithm is utilized to choose seed nodes from the set of candidate nodes. The community-based influence spread is defined as:

$$f_c(S) = f(C \cap S, G_C), \tag{11}$$

where $f(C \cap S, G_C)$ represents the number of activated nodes by the set of seed nodes $C \cap S$ in subnetwork G_C. Sine the influence scope of the candidate nodes is restricted within the communities where the nodes and their neighbors are located, the marginal influence increment is calculated as follows.

If the candidate nodes are located in the interior of the community, the marginal influence increment is calculated as:

$$\begin{aligned} f_c(u|S) &= f(u \cap (C_u \cap S), G_{C_u}) - f(C_u \cap S, G_{C_u}) \\ s.t. C_u &= \sum_{C \in C_s, C \cap u \neq \varnothing} C \end{aligned}. \tag{12}$$

If the candidate nodes are located on the boundary of the community, the marginal influence increment is calculated as:

$$\begin{aligned} f_c(u|S) &= f(u \cap (C_u \cap S), G_{C_u}) - f(C_u \cap S, G_{C_u}) \\ s.t. C_u &= \sum_{C \in C_s, C \cap nb(u) \neq \varnothing} C, \end{aligned} \tag{13}$$

where $nb(u) = \{v : (u, v) \in E\} \cup \{u\}$. The community-based influence

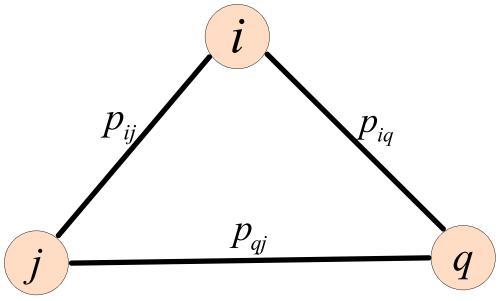

Fig. 6. The cost invested by maintaining the neighbor relationship.

Table 1
Seed nodes selection.

Input: Number of seed nodes k, candidate set U,
network G, community C_i;
Output: Seed nodes S;
1: Initialize S as null and calculate the marginal influence increment $f_c(v\|S_0)$ of each node v in U;
2: Select node v with the $f_c(v\|S_0)_{\max}$ into S_1, and remove v from U;
3: **for***j* from 1 to k **do**
4: Record aMax $= f_c(u\|S_j)$, aNode $= u$;
5: **for** each node $z \in$ U **do**
6: **if**$f_c(z\|S_{j-1}) >$ aMax, then
7: calculate $f_c(z\|S_j)$;
8: **if**$f_c(z\|S_j) >$ aMax, then
9: update aMax $= f_c(z\|S_j)$, aNode $=$ z;
10: **end if**
11: **end if**
12: **end for**
13: Add aNode into S_j and remove it from U;
14: **end for**

spread $f_c(S)$ satisfies the following inequations:

$$f_c(S) \geq 0, \tag{14}$$

$$f_c(S_1) \leq f_c(S_2), \tag{15}$$

$$f_c(S_1 \cup \{v\}) - f_c(S_1) \geq f_c(S_2 \cup \{v\}) - f_c(S_2), \tag{16}$$

where all $v \in V$ and $S_1 \subseteq S_2 \subseteq S$. To make full use of these characteristics for reducing time complexity, the proposed method only selectively updates the marginal influence increment of partial nodes, which can save lots of unnecessary Monte Carlo simulations without seed nodes selection. Thus, the main steps of the proposed method are illustrated in Table 1.

5. Experiments

To show the performance of the proposed method, some experiments, including comparisons, are implemented using Python on the Intel Xeon CPU with 2.7 GHz and 8 GB main memory running Windows 7.

5.1. Datasets

In our experiments, two real-world social networks (Cora and Citeseer) are adopted. The download link of Cora and Citeseer is as follows: http://www.cs.umass.edu/mccallum/data/and http://www.cs.umd.edu/sen/lbc-proj/data/citeseer.tgz. These two datasets all belong to citation networks, in which nodes and edges represent papers and the citation relationship between them, respectively. Tables 2 and 3 list the details of these two datasets.

As shown in Table 2, Cora consists of 2708 machine learning papers from seven classes, the citation network contains 5429 edges. Cora is regarded as an undirected graph because there are no link directions and self-links.

As shown in Table 3, Citeseer includes 3312 scientific publications from six classes, and there are 4732 edges between publications. Similar to Cora, link directions and self-links are also ignored. Citeseer is also regarded as an undirected graph. Table 4 further presents the details of statistical indicators in Cora and Citeseer, where d_{ave} and C are the average degree and the number of communities, respectively, and Q is the corresponding modularity.

5.2. Benchmark methods

To show the efficiency and effectiveness of the proposed method (called CBIMA), the following methods are compared with ours.

Table 2
Cora dataset.

Machine learning papers	Amounts
Case Based	351
Genetic Algorithms	217
Neural Networks	418
Probabilistic Methods	818
Reinforcement Learning	426
Rule Learning	298
Theory	180

Table 3
Citeseer dataset.

Scientific publications	Amounts
Agents	596
Artificial Intelligence	668
Databases	701
Information Retrieval	249
Machine Learning	508
Human Computer Interaction	590

Table 4
Details of statistical indicators in datasets.

Datasets	n	m	d_{ave}	Q	C
Cora	2708	5429	4.02	0.6424	7
Citeseer	3312	4732	2.86	0.4524	6

Greedy: Greedy is used to solve the IM problem. Since Greedy has very high performance guarantee and the proposed method aims to improve Greedy by utilizing the community structure, it is a good comparative method.

CELF: CELF is a state-of-the-art variant of Kempe's Greedy algorithm. CELF has the same problem with Greedy.

DegreeDiscount: DegreeDiscount is a heuristic algorithm based on degree centrality. Compared with degree heuristic algorithms, DegreeDiscount is able to avoid the aggregation of seed nodes to some extent.

ICRIM: ICRIM reduces the time complexity of Greedy by dividing the network into several independent communities and then searching for seed nodes within each community. In addition, different from CGA, ICRIM selects seed nodes from the interior and the boundary of each community to reduce the loss of influence spread caused by community division.

In the above-mentioned methods, Greedy and CELF are both greedy algorithms, DegreeDiscount belongs to the heuristic algorithm, and ICRIM is the community-based greedy algorithm. Here, the community-based heuristic algorithms are not selected as the comparison method, for the heuristic algorithms based on the entire network tend to have better performance.

5.3. Parameter settings

In the setting of simulation parameters, the size of seed nodes is 50. For the IC model, the probability that the activated nodes activate their neighbor nodes is the same. However, according to the homophily principle of the social network [24], the relationships between individuals are different, and the influence probability is closely related to the attractiveness of individuals. Therefore, it is reasonable to consider both the strength of association among individuals and the differences among individuals. The influence probability associated with edge $e = (u, v)$ is defined as:

$$p_{uv} = 2\frac{|nb(u)| \cap |nb(v)|}{|nb(u)| \cup |nb(v)|} \cdot \frac{|nb(u)| - 1}{|nb(v)| - 1}\ \overline{p}, \tag{17}$$

where $\overline{p}$ is the average propagation probability of the whole network, and the value of $\overline{p}$ is set to 0.1. Here, all original networks are undirected, but the diffusion models require that the networks should be directed, so each undirected edge is treated as two opposite directed edges [25,26].

5.4. Results analysis

Experiment 1 (Influence spread on Cora and Citeseer). The influence spread of different algorithms on Cora is demonstrated in Fig. 7. Specifically, Greedy and CELF both achieve the best performance. However, as discussed in the previous section, the Greedy algorithms are not suitable for large-scale social networks due to the high time complexity. In the case of suitable proportional parameters, the proposed method has the advantages of Greedy and CELF. Furthermore, it can be seen from Fig. 7 that the decrease of proportional parameter leads to the loss of influence spread. The influence spread of DegreeDiscount is always far less than other algorithms. This indicates that the heuristic algorithms only select seed nodes according to the centrality index of the network topology, which cannot offer good performance guarantees.

To further present the availability of the proposed method, experiments are conducted on Citeseer with poor quality of network partition (as shown in Table 3). Fig. 8 displays that both the running time and the influence spread of all algorithms have decreased since Citeseer is more sparse than Cora. Moreover, when the size of seed nodes is less than 30, the influence spread curves of all algorithms except the heuristic algorithm are relatively close, which indicates that the candidate stage hardly leads to the loss of influence spread. However, when the size of seed nodes increases, ICRIM and CBIMA are slightly worse than Greedy in terms of the influence spread. Since the community structure of Citeseer is weaker than Cora, the process of influence spread based on community structure deviates from the real situation.

Experiment 2 (Running time on Cora and Citeseer). Figs. 9 and 10 illustrate the running time of different algorithms on Cora and Citeseer, respectively. Here, the scale for *y*-axis in the figures is logarithmic. As illustrated in Fig. 9, Greedy and CELF take almost about 21,000 and 8300 s to find 50 seed nodes on Cora, which shows that Greedy algorithms are difficult to apply to large-scale social networks. In addition, since the influence spread of the heuristic algorithm DegreeDiscount is always far less than other algorithms, the experimental results do not show the running time of DegreeDiscount, which is usually within 10 s.

In Figs. 9 and 10, the running time of ICRIM and CBIMA on the public dataset is significantly lower than those of Greedy and CELF. This phenomenon can be explained as that since the scale of communities is much smaller than the entire network, searching for influential nodes in each community can effectively reduce the time complexity. More importantly, it can be seen from Fig. 9 that when the proportional parameter is 10, the running time of CBIMA is much lower than CELF and about 64% lower than that of ICRIM, which indicates that combining the community structure and the influence distribution difference can further reduce the time complexity. Furthermore, as shown in Fig. 10, similar experimental results have achieved on Citeseer. This also demonstrates that the proposed method is able to reduce the running time while ensuring the influence spread.

5.5. Case study

In order to demonstrate the effect of CBIMA, this subsection presents the case study of the distribution of seed nodes on Cora. Tables 5 and 6 show the specific information of the Top-5 seed nodes selected by CBIMA and DegreeDiscount, respectively. Cora belongs to the citation network, where the nodes and the edges represent papers and the citation relationship between papers, respectively, and the node has the corresponding class label (as illustrated in Cora). As shown in Table 5, the seed nodes selected by CBIMA are distributed in different communities, which avoids the problem of influence overlap. In addition, it can be seen that the most influential seed nodes come from some large-scale communities.

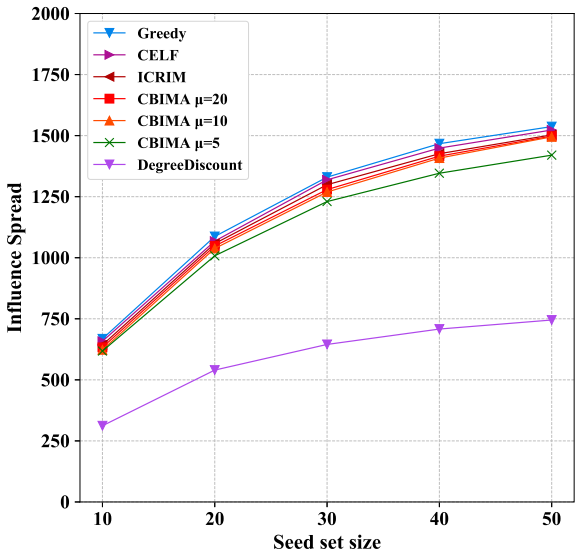

Fig. 7. Influence spread on Cora.

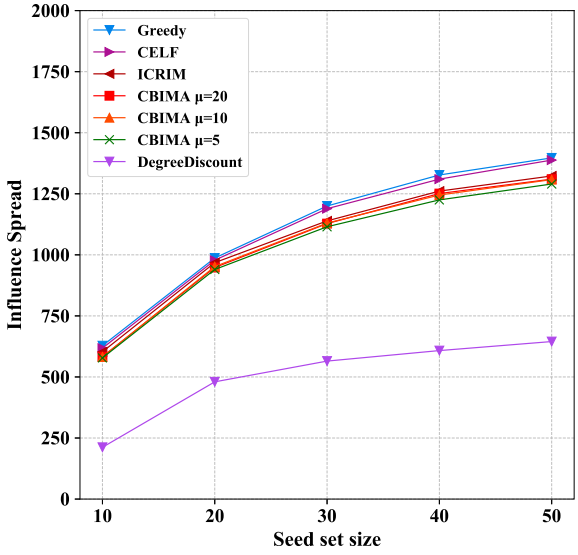

Fig. 8. Influence spread on Citeseer.

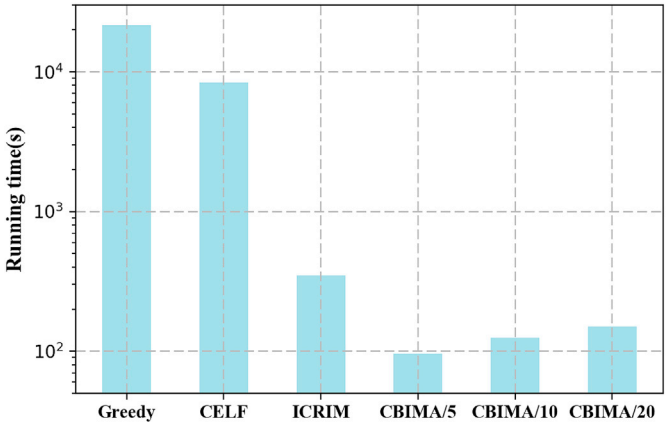

Fig. 9. Running time on Cora.

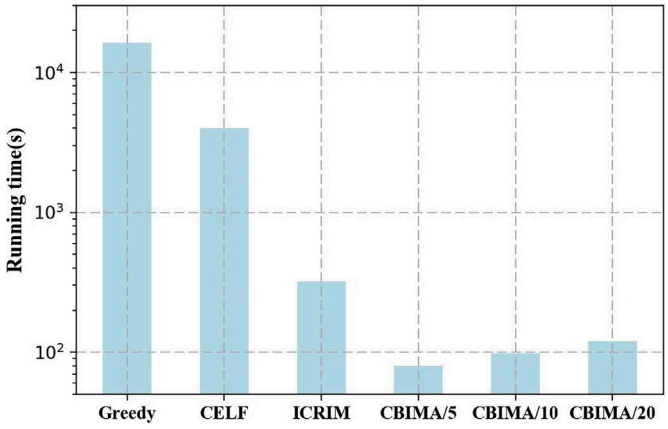

Fig. 10. Running time on Citeseer.

Table 5
Top-5 seed nodes of CBIMA.

Title	Class label
Feature subset selection with probabilistic estimation	Probabilistic Methods
Feature selection methods for classifications	Neural Networks
Induction of condensed determinations compression	Case Based
NP-Completeness searching for possible feature sets	Probabilistic Methods
Feature subset selection using a genetic algorithm	Genetic Algorithms

Table 6
Top-5 seed nodes of DegreeDiscount.

Title	Class label
Feature subset selection with probabilistic estimates	Probabilistic Methods
NP-Completeness searching for possible feature sets	Probabilistic Methods
Feature selection methods for classifications	Neural Networks
Numerical methods for optimization and equations	Probabilistic
Applications of machine learning and induction	Neural Networks

For example, the seed node "Feature subset selection as search with probabilistic estimates" is located in the "Probabilistic Methods" community.

Table 6 shows the specific information of the Top-5 seed nodes selected by DegreeDiscount. The heuristic algorithm only selects seed nodes according to some centrality metrics and ignores the actual process of influence spread, which may lead to the aggregation of seed nodes. The heuristic algorithm DegreeDiscount selects seed nodes according to the degree of nodes so that most seed nodes come from the community of "Probabilistic Methods" and "Neural Networks". And the aggregation of seed nodes will lead to the loss of influence spread. According to the above discussion, compared with the heuristic algorithms, the community structure-based IM can avoid the aggregation of seed nodes and achieve better performance in influence spread.

6. Conclusions

This paper explores a scheme for identifying influential nodes in social networks via the community structure and the influence distribution difference, in which the solution of the IM problem is divided into the candidate stage and the greedy stage. In the candidate stage, different heuristic algorithms are used to select candidate nodes from the interior and the boundary of each community. In the greedy stage, the submodular property-based Greedy algorithm is utilized to select the seed nodes with maximum marginal influence spread from the candidate set. Finally, experimental results demonstrate that, compared with the existing methods, the proposed method can ensure the influence spread while reducing the running time in large-scale social networks.

In future work, it is worth studying the effect of node content on the influence spread. One may apply the node classification method [27,28] and sentiment analysis methods [29,30] to identify the influential nodes. Since the user behavior [31] and social tie [32] play an important role in information diffusion, it is necessary to consider these factors for the node. In addition, with the successful application of deep learning technology in the Internet of Things [33,34], it is essential to explore how to use this technology to study the IM problem in Internet of Things.

Declaration of competing interest

The authors declare that there are no conflicts of interest.

Acknowledgements

The authors are grateful to the anonymous reviewers and the editor for their valuable comments and suggestions. This work is supported by Natural Science Foundation of China (Grant Nos. 61702066 and 11747125), Major Project of Science and Technology Research Program of Chongqing Education Commission of China (Grant No. KJZD-M201900601), Chongqing Research Program of Basic Research and Frontier Technology (Grant Nos. cstc2017jcyjAX0256 and cstc2018jcyjAX0154), Project Supported by Chongqing Municipal Key Laboratory of Institutions of Higher Education (Grant No. cqupt-mct-201901), Technology Foundation of Guizhou Province (QianKeHeJiChu[2020]1Y269), and New academic seedling cultivation and exploration innovation project (QianKeHe Platform Talents[2017]5789-21).

References

[1] Y. Sun, Y. Yuan, G. Wang, An on-line sequential learning method in social networks for node classification, Neurocomputing 149 (2015) 207–214.
[2] H. Han, X. Guo, Construction on framework of rumor detection and warning system based on web mining technology, in: 17th International Conference on Computer and Information Science, IEEE, Singapore, 2018, pp. 767–771.
[3] M. Ren, Y. Jiang, X. Guo, P2P networks monitoring based on the social network analysis and the topological potential, in: IEEE Conference on Communications and Network Security, IEEE, San Francisco, 2014, pp. 19–31.
[4] M. Ye, X. Liu, Exploring social influence for recommendation: a generative model approach, in: 35th International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, Portland, 2012, pp. 671–680.
[5] Y. Li, J. Fan, Y. Wang, Influence maximization on social graphs: a survey, IEEE Trans. Knowl. Data Eng. 30 (10) (2018) 1852–1872.
[6] P. Domingos, M. Richardson, Mining the network value of customers, in: 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, San Francisco, 2001, pp. 57–66.
[7] D. Kempe, J. Kleinberg, E. Tardos, Maximizing the spread of influence through a social network, in: 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, Washington, 2003, pp. 137–146.
[8] N. Ohsaka, T. Akiba, Y. Yoshida, et al., Fast and accurate influence maximization on large networks with pruned monte-carlo simulations, in: 28th AAAI Conference on Artificial Intelligence, AAAI, Québec, 2014, pp. 138–144.
[9] J. Leskovec, A. Krause, C. Guestrin, et al., Cost-effective outbreak detection in networks, in: 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, San Jose, 2007, pp. 420–429.
[10] W. Chen, Y. Wang, S. Yang, Efficient influence maximization in social networks, in: 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, Paris, 2009, pp. 199–208.
[11] E. Bagheri, G. Dastghaibyfard, A. Hamzeh, An efficient and fast influence maximization algorithm based on community detection, in: 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery, IEEE, Changsha, 2016, pp. 1636–1641.
[12] M. Kimura, K. Saito, R. Nakano, Extracting influential nodes on a social network for information diffusion, Data Min. Knowl. Discov. 20 (1) (2010) 70–97.

[13] K. Jung, W. Heo, W. Chen, IRIE: scalable and robust influence maximization in social networks, in: 12th IEEE International Conference on Data Mining, IEEE, Brussels, 2012, pp. 918–923.
[14] J. Kim, S. Kim, H. Yu, Scalable and parallelizable processing of influence maximization for large-scale social networks?, in: 29th IEEE International Conference on Data Engineering IEEE, Brisbane, 2013, pp. 266–277.
[15] S. Cheng, H. Shen, J. Huang, et al., IMRank: influence maximization via finding self-consistent ranking, in: 37th International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, Gold Coast, 2014, pp. 475–484.
[16] S. Chen, K. He, Influence maximization on signed social networks with integrated pagerank, in: International Conference on Smart City/SocialCom/SustainCom, IEEE, Chengdu, 2015, pp. 19–21.
[17] J. Li, Y. Yu, Scalable influence maximization in social networks using the community discovery algorithm, in: 6th International Conference on Genetic and Evolutionary Computing, IEEE, Kitakyushu, 2012, pp. 284–287.
[18] X. Li, X. Cheng, S. Su, et al., Community-based seeds selection algorithm for location aware influence maximization, Neurocomputing 275 (2018) 1601–1613.
[19] Y. Wang, G. Cong, G. Song, et al., Community-based greedy algorithm for mining top-k influential nodes in mobile social networks, in: 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, Washington, 2010, pp. 1039–1048.
[20] F. Ye, J. Liu, C. Chen, et al., Identifying influential individuals on large-scale social networks: a community based approach, IEEE Access 6 (1) (2018) 47240–47257.
[21] S. Loannidis, C. Augustin, On the strength of weak ties in mobile social networks, in: 2th ACM EuroSys Workshop on Social Network Systems, ACM, Nuremberg, 2009, pp. 19–25.
[22] G. Tong, W. Wu, S. Tang, et al., Adaptive influence maximization in dynamic social networks, IEEE/ACM Trans. Netw. 25 (1) (2017) 112–125.
[23] Y. Marom, D. Feldman, k-Means clustering of lines for big data, Neural Information Processing Systems (2019) 12797–12806.
[24] M. McPherson, L. Smith-Lovin, J. Cook, Birds of a feather: homophily in social networks, Annu. Rev. Sociol. 27 (1) (2001) 415–444.
[25] A. Goyal, F. Bonchi, L. Lakshmanan, et al., Learning influence probabilities in social networks, in: 3th International Conference on Web Search and Web Data Mining, ACM, New York, 2010, pp. 241–250.
[26] A. Goyal, W. Lu, L. Lakshmanan, SIMPATH: an efficient algorithm for Influence maximization under the linear threshold model, in: 11th IEEE International Conference on Data Mining, IEEE, Vancouver, 2011, pp. 211–220.
[27] J. Yang, T. Wang, L. Yang, et al., Preoperative prediction of axillary lymph node metastasis in breast cancer using mammography-based radiomics method, Sci. Rep. 4429 (2019) 1–11.
[28] Z. Zhang, X. Li, C. Gan, Multimodality fusion for node classification in D2D communications, IEEE Access 6 (2018) 63748–63756.
[29] Z. Zhang, Y. Zou, C. Gan, Textual sentiment analysis via three different attention convolutional neural networks and cross-modality compriseent regression, Neurocomputing 275 (2018) 1407–1415.
[30] C. Gan, L. Wang, Z. Zhang, et al., Sparse attention based separable dilated convolutional neural network for target entities sentiment analysis, Knowl.-Based Syst. 188 (2019) 1–10.
[31] C. Gan, X. Li, L. Wang, Z. Zhang, The impact of user behavior on information diffusion in D2D communications: a discrete dynamical model, Discrete Dynam Nat. Soc. 2018 (2018) 1–9.
[32] Z. Zhang, L. Wang, Social tie-driven content priority scheme for D2D communications, Inf. Sci. 480 (2019) 160–173.
[33] D. Wu, H. Shi, H. Wang, et al., A feature-based learning system for Internet of Things applications, IEEE Internet Things 6 (2) (2019) 1928–1937.
[34] P. Zhang, X. Kang, D. Wu, et al., High-accuracy entity state prediction method based on deep belief network toward IoT search, IEEE Wireless Commun. Lett. 8 (2) (2019) 492–495.