Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Influence maximization in social graphs based on community structure and node coverage gain

Zhixiao Wang [a,b], Chengcheng Sun [a], Jingke Xi [a], Xiaocui Li [c,*]

[a] School of Computer Science and Technology, China University of Mining and Technology, Xuzhou, Jiangsu, 221116, China
[b] Mine Digitization Engineering Research Center of the Ministry of Education, Xuzhou, Jiangsu, 221116, China
[c] Wuhan National Laboratory for Optoelectronics and School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, 430074, China

ARTICLE INFO

Article history:
Received 2 July 2020
Received in revised form 11 January 2021
Accepted 17 January 2021
Available online 22 January 2021

Keywords:
Social graph
Influence maximization
Overlapping community
Topological potential field
Node coverage gain

ABSTRACT

Influence maximization is an optimization problem in the area of social graph analysis, which asks to choose a subset of k individuals to maximize the number of influenced nodes at the end of the diffusion process. As individuals within a community have frequent contact and are more likely to influence each other, community-based influence maximization has attracted considerable attentions. However, this kind of works ignores the role of overlapping nodes in community structure, resulting in performance degradation in seeds selection. In addition, many existing community-based algorithms identify the final seeds only from the selected important communities, or they need to leverage the weights between local spread and global spread of a node. It is difficult to set suitable scales for important communities or to determine the weights for different spread. In this paper, we propose a novel influence maximization approach based on overlapping community structure and node coverage gain. Firstly, social graphs are partitioned into different overlapping communities by the algorithm of node location analysis in topological potential field. Secondly, a node coverage gain sensitive centrality measure is put up to evaluate the influence of each node locally within its belonging communities, which avoids the problem of local spread and global spread. Finally, seed nodes are directly selected by combining the detected community structure with the pre-designed strategy, without important communities identification. The comprehensive experiments under both the Uniform Independent Cascade model and the Weighted Independent Cascade model demonstrate that our proposed approach can achieve competitive influence spread, outperforming state-of-the-art works. Furthermore, our proposed approach exhibits stable performance on graphs with different scales and various structural characteristics.

1. Introduction

Social influence is the phenomenon that one's emotions or opinions can induce his/her friends to behave in a similar way [1]. The prevalence of social graphs has prompted much attention as they play an important role in information diffusion [2]. Influence Maximization (IM) aims to find a set of k influential nodes (called seed set) from social graphs to maximize the expected number of influenced nodes under a certain propagation model [3]. Due to its immense application potential in various domains, such as viral marketing, target advertisement, rumor control and social recommendation, IM has been extensively studied in the past decade.

In recent years, the problem of influence maximization has some variants, such as the context-aware IM [4], which combines the information of location [5], user interest [6], and topic [7] to improve the effectiveness. There are also some other variants such as budgeted IM [8], competitive IM [9] and fair IM [10]. However, in this paper, we only focus on the general IM problem which asks to select a subset of k individuals for initial activation to maximize the number of influenced nodes at the end of the diffusion process under a given spreading model.

Kempe et al. [11] has proven that IM is a NP-hard problem and they proposed a greedy algorithm with an approximation of $(1 - 1/e)$ to solve the problem. After that, many greedy/heuristic techniques [12–18] are proposed to provide near-optimal solutions. The rich-club phenomenon (a small number of nodes with large numbers of links tend to connect with each other) [19] is crucial to the IM problem. This phenomenon leads to the effect that if it only selects high-evaluated individuals as seeds, the final influence spread would be far from satisfying. In other words, high-degree nodes usually possess high individual influence while also tend to connect with each other. Their influenced

[*] Corresponding author.
E-mail address: lxc@cumt.edu.cn (X. Li).

https://doi.org/10.1016/j.future.2021.01.025
0167-739X/© 2021 Elsevier B.V. All rights reserved.

areas will thus be highly overlapped. Greedy algorithms [11–14,20,21] utilize Monte-Carlo simulations to ensure the performance, which is time consuming and limits its application on large-scale graphs. Heuristic algorithms [12,15,22] devise various measurements to estimate the influence spread of each node instead of running heavy Monte-Carlo simulations that heavily depends on the structure characteristics of graphs and cannot guarantee the performance.

Social graphs exhibit natural community structures, i.e., containing groups of nodes that have denser connections within each group and fewer connections between groups [23]. Because of the useful characteristics of community structure in social graphs, some attentions have been made to incorporate the role of communities in influence maximization problem [24–28]. Compared with traditional greedy/heuristic methods, community-based methods can reduce the computational time and increase the performance [29]. However, there still remain some unsolved problems for this kind of methods: (1) When partitioning a social graph into communities, most state-of-the-art community-based researches [24–28] did not take into consideration the role of overlapping nodes, i.e., they usually evaluate a node's influence only within its own community. In fact, the overlapping nodes are ties that connect different communities and build their spreading paths. The ignorance of these overlapping nodes will decrease the accuracy of seed selection and the performance of influence spread. (2) When evaluating an individual's influence spread or select a seed node, many state-of-the-art community-based works need some additional information beyond the graph itself. For example, the number of important communities and the scale of candidate seed nodes in [24] and [25], the weights of local (intra-community) influence and global (inter-community) influence in [26] and [27], etc. Actually, it is difficult to set suitable values in advance.

Motivated by the above observations, this paper proposes a novel influence maximization framework for social graphs based on Community structure and Node Coverage Gain, namely CNCG. Firstly, overlapping community structure is detected by the algorithm of node location analysis in topological potential field [30]. Secondly, a node coverage gain sensitive centrality measure is presented to estimate the influence spread of each node within belonging communities. Finally, seed nodes are directly selected from different communities based on the results of node influence evaluation.

The main contributions of this paper are summarized as follows:

(1) The proposed approach adopts the overlapping community structure for further identifying seed nodes, which could avoid the effect of the rich-club phenomenon at the inter-community level. Since overlapping nodes are the ties that connect different communities and also build their spreading paths, we point out that these overlapping nodes are important for the IM problem. Putting emphasis on these overlapping nodes would improve the accuracy of seed selection and the performance of influence spread, which is also verified by the experimental results.

(2) The proposed approach presents a node coverage gain sensitive centrality measure to evaluate the influence of each node locally within its own community, which could effectively avoid the effect of the rich-club phenomenon at the intra-community level. The proposed centrality only involves a node's neighbors within its belonging communities. Thus, the overlapping nodes are more easily to be selected as seeds in a natural manner since they belong to several communities. Moreover, the proposed approach does not need to leverage the weights between local and global spread, nor to identify important communities with a manually defined threshold, which achieves better generality and stability.

Experimental results under both the Uniform Independent Cascade model (UIC model) and the Weighted Independent Cascade model (WIC model) demonstrate the performance of our proposed approach. With the seed nodes selected by CNCG, we can get the satisfactory influence spread, outperforming the state-of-the-art methods. Furthermore, our proposed CNCG approach has stable performance on various social graphs.

The remainder of this paper is organized as follows. Section 2 reviews the related works. Section 3 describes our proposed approach in details. Section 4 presents the experimental results under both the UIC model and the WIC model. The final conclusions come to in Section 5.

2. Related works

Kempe et al. [11] first proved that the IM problem is NP-hard. They proposed the Greedy algorithm which obtains the approximate real influence for any given node set through a large number of Monte-Carlo simulations. They also proved that the IM problem is monotone submodular and Greedy is thus guaranteed to be within about $(1 - 1/e)$ of the optimal solution theoretically. However, the Greedy is very inefficient to deal with large networks. Later, Leskovec et al. [13] presented CELF algorithm which is 700 times faster than the Greedy. The reason is that they exploited the submodularity of the IM problem and pruned a large number of unnecessary simulations. Moreover, Goyal et al. [14] presented CELF++ to improve the runtime of CELF. The CELF++ extends CELF to include the marginal gain with the node that has the previous highest value. This approach achieves better pruning while only bringing limited improvements empirically [31]. To reduce the runtime of Greedy, some researchers use a large number of network snapshots rather than pruning the Monte-Carlo simulations. Chen et al. [12] first presented this idea and proposed NewGreedy which constructs a number of snapshots to approximate a node's real influence. The StaticGreedy proposed by Cheng et al. [21] produces those snapshots at first to perform all subsequent evaluations. They need fewer snapshots of the network compared with NewGreedy empirically. However, when a network is large, the number of snapshots that these algorithms require may heavily challenge the computer memory.

As the influence of a node could be evaluated easier and faster in a small subgraph than in the original full graph, community-based solutions have been developed for solving IM problem in recent years. This kind of approaches could bring down the problem into the community level by using community detection of the social graph as an intermediate step [32].

The first study toward the community-based solution is proposed by Wang et al. [20]. They detect communities based on information propagation and then select parts of communities for finding influential nodes. Later on, many community-based algorithms appear. To simplify seed selection, ComPath [24] identifies the most influential communities and candidate seed nodes. Final seed set is obtained based on the intra distance among the candidate seed nodes. A similar approach can be found in FSIM [25], which uses the betweenness centrality measure to select a limited number of communities as important communities. Then some nodes from important communities will be selected as candidate seed nodes. Different from ComPath, the final seed set of FSIM is obtained by node importance comparison. However, it is difficult to determine the proper values for the number of important communities and candidate seed nodes.

Bozorgi et al. [28] proposed another community-based method named INCIM, which also exploits the community structure of

Table 1
Notations used in this paper.

Notation	Definition
G	Social graph
V, E	Nodes and edges of G
n	Number of nodes in G
m	Number of edges in G
$\langle d\rangle$	The average degree of G
cc	The clustering coefficient of G
d_{max}	The max degree of G
$\phi(v_i)$	The topological potential of node v_i
C	The community structure of G
C_i	The ith community of in C
c	The total community number of C
v_i	A node in community C_i
$N(v)$	The neighbors of v
$N_i(v)$	The neighbors of v in community C_i
$Cover(v_i)$	The node coverage of v_i
$Gain(v_i)$	The node coverage gain of v_i
S	The seed set
$\bar{S}$	The set of non-seed nodes
M_{ci}	The node with maximum gain within the community C_i
M_{gr}	The node with maximum gain in the whole network which comes from C_r
k	The total number of seed nodes
p_c	The spreading threshold of a network under the UIC model

graph to find the influential communities. Compared with ComPath and FSIM, INCIM [28] finds influential nodes based on a combination of local (intra-community) and global (inter-community) influence evaluation. Later, Shang et al. [26] provided a community-based framework named CoFIM for IM problem, which divides the influence propagation process into two phases: seeds expansion and intra-community propagation. The first phase is the expansion of seed nodes among different communities, and the second phase is the influence propagation within communities. The overall influence of a seed node is evaluated by the combination of the influence during the two phases. To improve CoFIM [26], Shang et al. [27] proposed a similar framework IMPC with accelerating techniques, which divides the influence diffusion process into two phases: multi-neighbor potential based seeds expansion and intra-community influence propagation. They also provide an objective function to evaluate the overall influence as a combination of the influence during the two phases. For this kind of methods, the parameters weighting local and global influence (such as γ in CoFIM, α and β in IMPC) are difficult to predefine.

More importantly, the community-based methods mentioned above ignore the role of overlapping nodes in community structure. They usually evaluate a node's influence only within its own community. In fact, the overlapping nodes are ties that connect different communities and build their spreading paths. The ignorance of these overlapping nodes will decrease the accuracy of seed selection and the performance of influence spread.

3. Methods

In this section, we propose a novel influence maximization approach for social graphs. Firstly, we partition the input social graph into a series of communities with the algorithm of node location analysis in topological potential field. Secondly, we put up a node coverage gain sensitive centrality metric to evaluate the influence of each node within its local community. Finally, we select seed nodes from different communities by combining the detected community structure with the pre-designed seed nodes strategy. Detail descriptions of the proposed approach will be provided in the following subsection. Table 1 lists most of the notations used in this paper.

3.1. Community detection

There are many kinds of community detection algorithms. Due to the inherent advantages in terms of performance, the topological potential based methods have attracted considerable attentions [30,33]. We proposed an overlapping community detection algorithm based on node location analysis in the topological potential field in our previous work [30] which can detect the overlapping community structure accurately and provide overall good-standing performance. Therefore, we adopt this algorithm to partition social graphs into overlapping communities in this paper.

A social graph G can be denoted as $G = (V, E)$, where V represents the nodes in G and E refers to the edges that link these nodes. In many research fields, the topological potential field is used as a mathematical model to describe the non-contact interactions between objects [33]. Since the nodes of social graphs are not isolated but linked by edges, we use topological potential field model to describe the interactions among these nodes. Each node is regarded as a field source which creates a potential field around itself [30]. All nodes interact with each other, forming a field called the topological potential field.

Definition 1 (*Topological Potential*)**.** Given a social graph $G = (V, E)$ and its corresponding topological potential field, the topological potential of node v_i is defined as [30]:

$$\phi(v) = \sum_{u\in G} m(u) * e^{-(\frac{h_{vu}}{\sigma})^2} \tag{1}$$

where $\phi(v)$ represents the topological potential of v, n refers to the total number of nodes in G, $m(u)$ denotes the weight of u, h_{vu} indicates the number of hops between nodes v and u, and σ is an impact factor used to control the impact scope of nodes.

An object in Physics has its inherent mass properties while it does not work for network nodes in data science. Therefore, we associate node mass with its importance in the network and measure it via the PageRank algorithm [34] as:

$$m(v) = dp \sum_{u\in N(v)} \frac{m(u)}{|N(u)|} + (1 - dp) \tag{2}$$

where dp (the damping factor) is 0.85 according to [34] and $N(v)$, $N(u)$ denotes the neighbors of v and u respectively.

Besides, if $h_{vu} > \lfloor 3\sigma/\sqrt{2}\rfloor$, the topological potential component produced by node v_j on v_i will be very weak and can be ignored according to the three-sigma rule and the Gaussian function [30]. The impact factor σ controls the impact scope of each field source and thus determines the distribution of topological potential. If σ is too small, each field source can only impact a limited scope of nodes. When σ is close to 0, each source can only have an impact on themselves. If σ is too large, every source would impact all nodes in the network, which is not reasonable in the real-world scenarios. Therefore, an appropriate value is needed for truly reflecting the structural characteristics of the social graph. Similar to other works, potential entropy [30] is adopted to select the optimal value of the impact factor σ.

We have analyzed the characteristics of the topological potential field in our earlier work [30] and found that it presents a natural peak–valley structure. Nodes with large potential values always locate at relatively high positions of the field. Conversely, nodes with small potential values always locate at relatively low positions [30]. We can partition social graphs into overlapping communities based on node location analysis in the topological potential field. Generally speaking, nodes in the topological potential field are classified into three categories, namely, Peak, Slope, and Valley. The Peak are nodes whose potential is locally maximal. Analogically, The Valley are nodes whose potential is locally minimal. The nodes neither belong to Peak or Valley are Slope. Similar to a contour map, a Peak and its surrounding Slopes form a "hill", which corresponds to a community in network structure. Valleys are the borders of those "hills", which are thus the overlapping borders of communities. For more details about node location analysis, please refer to our earlier works in [30] and [33].

3.2. Node coverage gain evaluation

Based on the detected community structure, we can further identify the influential nodes. On one hand, Wang et al. [20] found that the difference between influence in its community and influence in the whole graph is small. Ok et al. [35] also pointed out that it is necessary to find seeds accurately for the local graphs rather than global graphs. On the other hand, the overlapping nodes are the ties that connect different communities and build their spreading paths. The ignorance of these overlapping borders will decrease the accuracy of influential node identification. Therefore, we proposed a novel centrality measure, which put emphasis on the overlapping nodes, to evaluate the influence of each node locally within its belonging communities.

Definition 2 (*Single Node Coverage*)**.** Given a social graph $G = (V, E)$, $C = \{C_1, C_2, C_3, \ldots\}$ is the community structure of G. The coverage of a single node v is defined as:

$$Cover(v) = \left\{u \in N(v) \middle| u \in \bigcup_{v \in C_i} C_i\right\} = \bigcup_{v \in Ci} N_i(v) \tag{3}$$

where $Cover(v)$ denotes the single node coverage of v, N_i refers to the neighbors of v that belong to community C_i.

Definition 3 (*Node Set Coverage*)**.** Given a social graph $G = (V, E)$, $C = \{C_1, C_2, C_3, \ldots\}$ is the community structure of G. There is a node set S consisting of nodes that may from different communities. The coverage of node set S is defined as:

$$Cover(S) = \left\{\bigcup_{v \in C_i} N_i(v) \middle| v \in S\right\} = \bigcup_{v \in S} Cover(v). \tag{4}$$

Definition 2 shows that the single node coverage is the set of a node's neighbors that share the same community. Thus, the overlapping nodes are emphasized in a natural manner since they belong to several communities. Definition 3 shows that the node set coverage is the union of these nodes' single coverage rather than adding up their numbers. According to the rich-club effect, high-evaluated nodes tend to connect with each other and thus share lots of common neighbors. This effect could be greatly mitigated when applying the node set coverage to select seed nodes.

Definition 4 (*Node Coverage Gain, Short for Gain*)**.** Given a social graph $G = (V, E)$ and the set of selected seeds S. The coverage gain of node v is defined as:

$$Gain(v) = \left|Cover(S) \bigcup Cover(v) \bigcap \bar{S}\right| - \left|Cover(S)\right| \tag{5}$$

where $Gain(v)$ denotes the node coverage gain of v_i^k, $Cover(S)$ refers to the coverage of seed set S, $Cover(v)$ represents the coverage of node v, $\bar{S}$ is the set of non-seed nodes, $\bar{S} = V \backslash S$.

The above three definitions reveal that the coverage of a node depends on its neighbors. Usually, the bigger a node's degree is, the larger its coverage will be. The gain of a node reflects the potential influence spread gain if added to the seed set. A larger coverage does not always imply a bigger gain. According to Definition 4, only the node with a big gain can be selected as a seed node to maximize the influence spread.

Take Fig. 1 as a simple example to illustrate the node coverage gain evaluation. In Fig. 1(a), the degree of *node* 15 is the biggest in this schematic graph, therefore, it is selected as seed preferentially. The coverage of *node* 15, i.e. $Cover\,(15) = \{10, 11, 12, 13, 14, 16, 17, 18, 20, 22, 23, 24\}$. Once *node* 15 is selected as seed node, the gain of other nodes will be updated. $Gain\,(22) = |\{19, 21\}| = 2$, as shown in Fig. 1(b). $Gain\,(5) = |\{1, 2, 3, 4, 6\}| = 5$, as shown in Fig. 1(c). Obviously, we should select the *node* 5 as the second seed rather than *node* 22, although the latter has very large degree. Thus, the rich-club effect can be avoided efficiently.

3.3. Seed selection

The community detection divides the social graph into a series of communities and then the gain of each node is evaluated via node coverage gain. For the seed selection, we utilize the results of community detection and node coverage gain evaluation to select appropriate nodes as seeds. In order to avoid the rich-club effect and maximize the influence spread, seed nodes should be selected among all local communities, rather than only from a few so-called important communities. The three-step seed selection strategy is described as follows:

Step 1: Each community has a largest-gain node within itself. If the gain values of these nodes are different from each other, the node with the maximum gain will be preferentially selected as a new seed.

Step 2: If there is more than one node having the same maximum gain in the whole network in Step 1, the node from the community where there does not exist any other seeds will be selected as a new seed.

Step 3: If there is more than one node having the same maximum gain in the whole network and each corresponding community has at least one seed in Step 2, the node from the community with the largest scale (i.e., number of nodes) will be selected as a new seed.

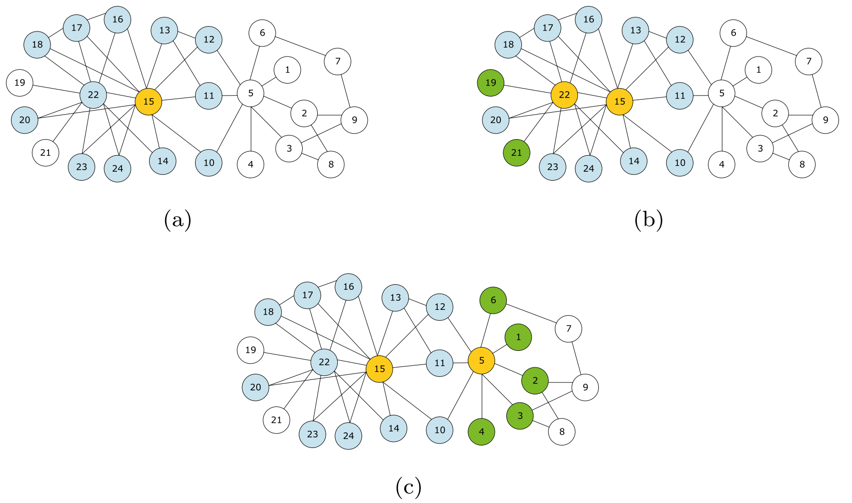
For the first step, if a node has the maximum gain value, it must be the most influential node and should be preferred as seed. For the second and third steps, we should give priority to the community with large scale and none seeds. It should be noted that this three-step strategy only selects one seed node each time. After a seed node is selected, the coverage gain of the remaining nodes will be updated for the next selection.

3.4. Algorithm description

The pseudo-code of our proposed algorithm is described as follows.

3.5. Complexity analysis

In Algorithm 1, line (1) is the initialization, lines (2)–(7) identify the node with maximum coverage gain of community level, the complexity is $O\left(n \left\langle d \right\rangle\right)$, n denotes the number of nodes in the social graph, $\left\langle d \right\rangle$ refers to the average degree of the social graph. Lines (8)–(16) select the k seed nodes iteratively. Suppose the social graph is divided into c communities and the number of nodes in the largest community is l. In the first iteration, the node with maximal degree is selected as seed. For the rest $k - 1$ times iterations, we need to recalculate the nodes' gain of one community at each iteration, the time complexity is $O((k-1)\, l \left\langle d \right\rangle)$. Thus, the total complexity of Algorithm 1 is $O\left(n \left\langle d \right\rangle\right) + O\left((k-1)\, l \left\langle d \right\rangle\right) = O\left((kl + n) \left\langle d \right\rangle\right)$.



(a)

(b)

(c)

Fig. 1. A simple example of node coverage gain evaluation.

Algorithm 1: Influence maximization based on community structure and node coverage gain

Input: Network $G = (V, E)$, Community structure C, the seed size k
Output: The set of seed nodes S

```
Initialize $S \leftarrow \emptyset$;
For each $C_i \in C$ do //$C_i$ refers to the $i^{th}$ community of $C$
    $M_{ci} = argmax\{Gain(v_i)|v_i \in C_i\}$;
    //$M_{ci}$ refers to the node with maximum coverage gain in $C_i$
    $U = U \bigcup \{M_{ci}\}$;
    //$U$ refers to the set of the nodes with maximum gain in each
    community
End for
For $i = 1$ to $k$ do
    select the seed node $M_{gr}$ from $U$ with the strategy in Section 3.3;
    //$M_{gr}$ refers to the node with maximum gain in whole network that
        from $C_r$
    $S = S \bigcup \{M_{gr}\}$;
    For each $C_i$ that $M_{gr}$ belongs to do
        update $M_{ci}$ in $U$;
        //finding a new node with the maximum coverage gain in $C_i$
    End for
End for
```

4. Experiments

4.1. Datasets and models

Datasets. In this paper, seven real world networks with different sizes and various structure characteristics are selected to evaluate the performance of our proposed approach.

(1) NetScience[1]: This is a co-authorship network of scientists in network theory and experiments.
(2) Yeast[1]: This is the Yeast's protein interaction network.
(3) Power[1]: This is a power-topology network of the Western States Power Grid of the United States.
(4) CaGrQc[1]: This is a collaboration network of arXiv General Relativity.
(5) NetHept[2]: This is a collaboration network of arXiv High Energy Physics - Theory.
(6) Brightkite[1]: This network contains user–user friendship relations from Brightkite, a location-based social graphing service provider where users shared their locations by checking-in.

Table 2
Detail information of seven real-world networks.

Networks	n	m	$\langle d\rangle$	d_{max}	cc	p_c
NetScience	1589	2742	3.4512	34	0.6378	0.322
Yeast	2361	6646	5.6298	64	0.1301	0.070
Power	4941	6594	2.6691	19	0.0801	0.437
CaGrQc	5242	14 496	5.5307	81	0.5296	0.090
NetHept	15 233	31 376	4.1195	64	0.4984	0.075
Brightkite	58 228	214 078	7.3531	1134	0.1723	0.015
Wordnet	146 005	656 999	8.9997	1008	0.6021	0.020

(7) Wordnet[2]: This network depicts the relations of lexical words from the WordNet dataset.

Detail information of these seven networks are described in Table 2. n, m refer to the number of nodes and edges, respectively; $\langle d\rangle$, d_{max} represent the average degree and the max degree, respectively; cc refers to the clustering coefficient; p_c [36] denotes the spreading threshold under the UIC model, which is determined as the position of the maximum of the susceptibility $\left\langle S^2\right\rangle / \langle S\rangle^2$ where $\langle S\rangle$, $\langle S_2\rangle$ represent the 1st and 2nd moment of the outbreak size distribution computed for random initial single spreaders.

In this paper, two spreading models are involved for evaluating the influence spread, i.e. the Uniform Independent Cascade model (UIC model for short) and the Weighted Independent Cascade mode (WIC model for short) [11].

In the UIC model, the spreading process starts with the initial set S consisting of k active nodes. Let A_i be the set of nodes that are activated in the ith round, and $A_0 = S$. When $i > 0$, all nodes in A_{i-1} will have only one chance to activate each of its inactive neighbors (which do not belong to A_j for $0 \le j \le i$) with a constant influence probability. If succeed, the activated neighbor will be merged to A_i. The whole process ends when A_i is empty.

The WIC model works just like the UIC model, only that the influence probability is not constant but the reciprocal of the degree of each being activated node. Thus, in the WIC model, nodes with a smaller degree are easier to be activated.

4.2. Baseline algorithms

Five state-of-the-art methods are selected as baseline algorithms, including Degree, LIR [16], DegreeDiscount(DD for short) [12], ProbDegree(ProbD for short) [17] and CoFIM [26]. In addition to the above five baseline algorithms, a simplified version of our

[1] http://snap.stanford.edu/data.
[2] http://konect.uni-koblenz.de/networks/.

proposed approach is also selected as baseline algorithm. The brief introduction of these algorithms are as follows.

Degree: A classic influence maximization algorithm which selects the $top - k$ largest degree nodes as the seeds.

LIR [16]: A simple and fast influence maximization algorithm which directly picks the local leader with biggest degree as seeds.

DD [12]: An improved version of the Degree algorithm. It adds a discount process to avoid the rich-club effect. If a node is selected as seed, the DD algorithm will discount the degree of its neighbors.

ProbD [17]: An improved version of the Degree algorithm which takes into account the degree of a node and its neighbor's. The influence of a node depends on its degree, neighbors' degree and the influence probability.

CoFIM [26]: A community-based framework for influence maximization on large-scale graphs. It contains two diffusion phases: the first phase is the expansion of seed nodes among different communities, and the second phase is the influence propagation within communities.

CELF++ [14]: A greedy-based method that improves CELF in terms of time cost. This method also guarantees that the influence spread is at least $(1 - 1/e)$ of the optimal results.

NCG: A simplified version of our proposed CNCG method which does not contain the step of community detection and the coverage gain of each node is evaluated within the whole graph, rather than locally within its own community. In other words, the whole network is viewed as a single community.

4.3. Tests under the UIC model

We evaluate the performance of our proposed CNCG approach under the UIC model and compare it with the baseline algorithms.

Firstly, we vary the size of seed nodes k from 5 to 50 to evaluate the influence spread of different algorithms. In all experiments, we run the Monte-Carlo simulation 10,000 times to obtain the near-stable influence spread. The influence probability is set with the spreading threshold p_c. According to [26], the parameter γ of CoFIM is set 3. Fig. 2 shows the corresponding results on seven networks, including NetScience, Yeast, Power, CaGrQc, NetHept, Brightkite and Wordnet. X-axis represents the seed nodes size k and Y-axis refers to the influence spread. For NetScience network (Fig. 2(a)), the influence spread of Degree, LIR, DD and ProbD algorithms are smaller than that of other four algorithms. For Yeast network (Fig. 2(b)), LIR exhibits the worst performance, while Degree and ProbD are not so well either. CNCG is even better than the CELF++ whose performance is theoretically guaranteed. In Power network (Fig. 2(c)), when the seed nodes size k is smaller than 20, Degree algorithm can maintain relatively stable performance. Otherwise, it cannot guarantee the accuracy of seed selection, decreasing the influence spread. In CaGrQc network (Fig. 2(d)), both Degree and DD show an obviously poor influence spread. In NetHept network (Fig. 2(e), CNCG shows a competitive performance with CELF++, outperforming the other algorithms when k is bigger than 35. In Brightkite network (Fig. 2(f)), when the seed nodes size k is bigger than 10, LIR algorithm cannot find enough local leaders as seed nodes, resulting in the smallest influence spread. In Wordnet network (Fig. 2(g)), CELF++ shows the best performance, with our CNCG comes the second.

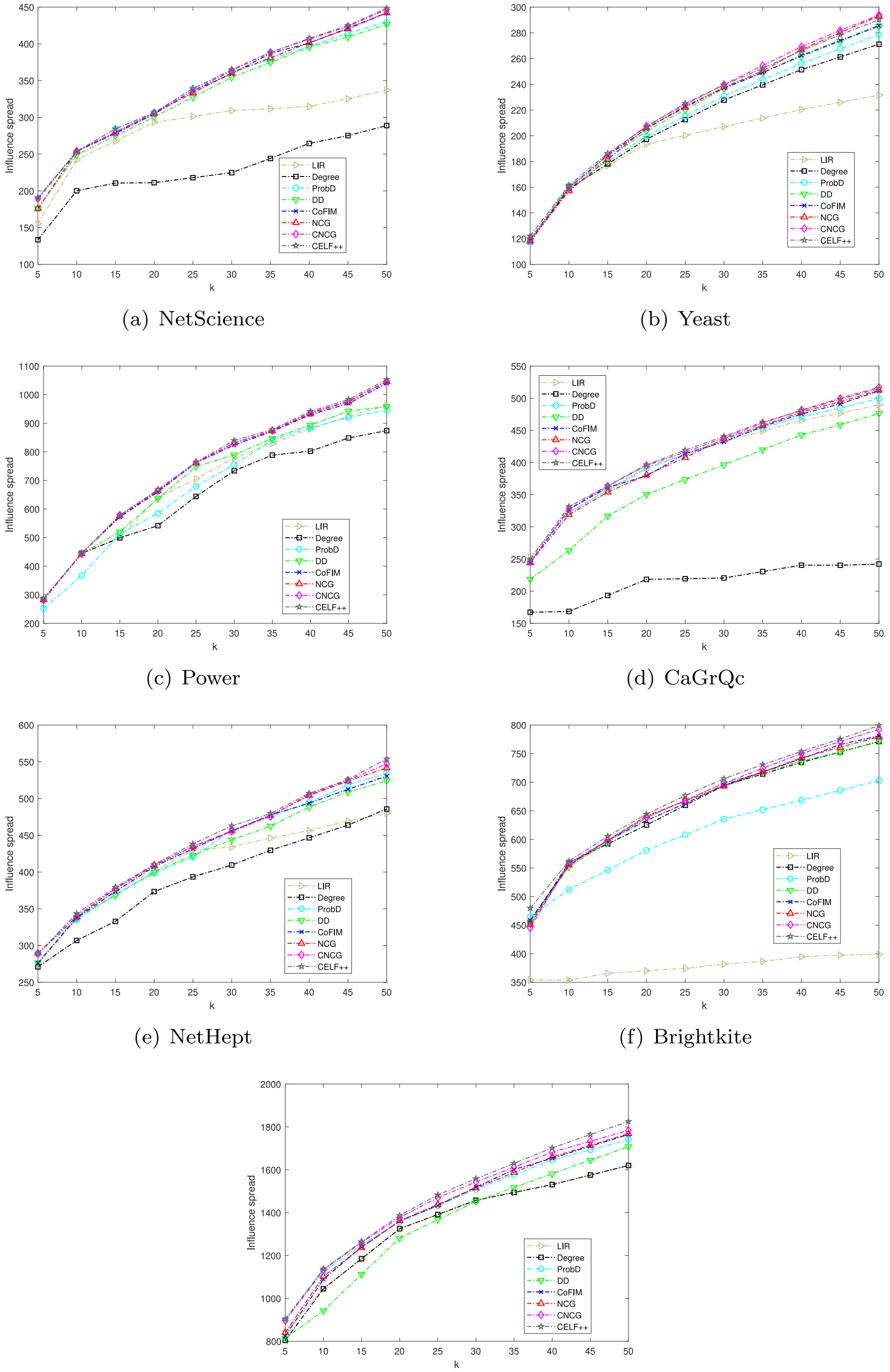
Based on the above experimental results, we found that: (1) The scales and structure characteristics of graphs can significantly affect the influence spread of LIR and Degree algorithms; (2) Both DegreeDiscount and ProbDegree are the improved versions of the Degree algorithm, therefore, they show a better performance than that of original Degree algorithm; (3) Compared with other six algorithms, both CELF++ and CNCG algorithms show stable performance with the variation of seed nodes size k; (4) CNCG exhibits a higher influence spread than that of NCG algorithms. The larger the community numbers in graphs, the greater the performance improvement.

Secondly, we evaluate the influence spread of different algorithms with the variation of the influence probability p. The seed nodes size k is set 50. According to the spreading thresholds p_c of different networks shown in Table 2, we range the influence probability of NetScience and Power networks between 0.05 and 0.5 with the step length 0.05. Likewise, the influence probability of Yeast and CaGrQc networks is set between 0.01 and 0.09 with the step length 0.01. The corresponding range of Brightkite and Wordnet networks is set between 0.005 and 0.05 with the step length 0.005. Fig. 3 shows the corresponding results on seven networks. X-axis represents the influence probability p and Y-axis refers to the influence spread.

Generally, the experimental results in Fig. 3 are similar to those in Fig. 2. With the variation of influence probability p, CNCG almost exhibits the best influence spread on seven networks, outperforming other algorithms. When the influence probability p is near the spreading threshold p_c of the corresponding network, the influence spread difference of eight methods is almost the most obvious, especially for NetHept network (Fig. 3(e)) whose p_c value is 0.075. An interesting phenomenon is that when the influence probability is around p_c, CELF++ shows good performance. However, when the probability is far from the p_c, the performance of CLEF++ gets worse, sometimes even lower than the classic DD algorithm. For example, as Fig. 3(a) shows, when p is between 0.3 and 0.4, CELF++ is obviously better than the other seven algorithms. In other cases, CELF++ is worse than CNCG (for $p > 0.4$) or even worse than most algorithms (for $p < 0.2$). It reveals that the performance of CELF++ (and other Greedy-based algorithms) is highly associated with the influence probability. It has to recalculate the seeds when the influence probability changes or its performance would not be guaranteed (sometimes even worse than some simple heuristic algorithms). However, our proposed CNCG algorithm selects seeds without knowing the influence probability, showing a better generality.

Thirdly, we evaluate the spreading speed of different algorithms, which can be revealed by the number of activated nodes at each time step of the diffusion process. The seed nodes size k is set 50. Fig. 4 shows the corresponding results on four networks, including NetScience, CaGrQc, Brightkite and Wordnet. X-axis represents the time step and Y-axis refers to the influence spread.

As shown in Fig. 4, the influence spread of eight methods increase steadily during the diffusion process. The bigger the time step, the larger the influence spread. Some degree-based methods, such as Degree and DegreeDiscount, tend to select the nodes with more neighbors as seeds, thus, they can obtain a higher influence spread at the beginning of the diffusion process. But, their spreading speed will slow down after several time steps. This phenomenon can also be found in CaGrQc, Brightkite and Wordnet networks. Because the scale of NetScience is very small, the influence spread of ProbDegree, DegreeDiscount, CoFIM, NCG and CNCG hardly increase anymore when the time step is bigger than 4. NCG and CNCG methods may not obtain the best influence spread at the initial time step, but they exhibit a faster growth afterwards than that of other methods. In CaGrQc network (Fig. 4(b)) and Brightkite network (Fig. 4(c)), the performance of CNCG will outperform other algorithms when the time step is bigger than 4 and 6. Even CELF++ would achieve high influence spread at the end of the spreading process just as Fig. 2 shows, its spreading speed is rather slow compared with the other algorithms.



(a) NetScience

(b) Yeast

(c) Power

(d) CaGrQc

(e) NetHept

(f) Brightkite

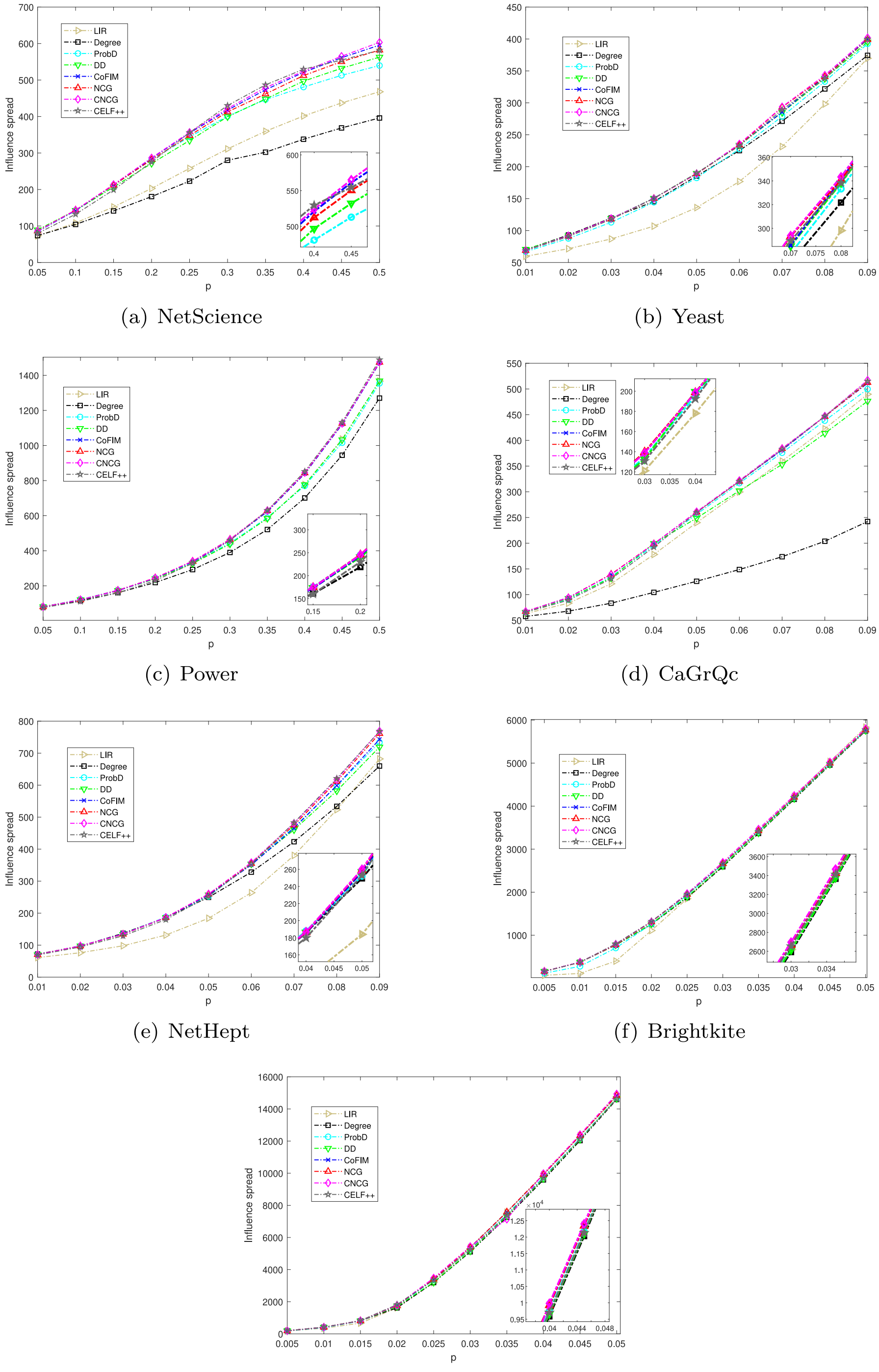
(g) Wordnet

Fig. 2. The influence spread of eight algorithms on seven networks with the variation of seed nodes size k under the UIC model.

4.4. Tests under the WIC model

In this subsection, we evaluate the influence spread of our proposed CNCG approach under the WIC model and compare it with the baseline algorithms. We still run the Monte-Carlo simulation 10,000 times to obtain the near-stable influence spread. The seed nodes size k ranges from 5 to 50. Fig. 5 shows the corresponding results on seven networks, including NetScience, Yeast, Power, CaGrQc, NetHept, Brightkite and Wordnet. X-axis represents the seed nodes size k and Y-axis refers to the influence spread.

The experimental results under the WIC model are similar to those under the UIC model. With the variation of seed nodes size k, CELF++ and CNCG exhibit the best influence spread on seven networks, outperforming other six algorithms. For simplicity, we will not repeat it here. What needs to be pointed out in particular is that: compared with UIC model, the influence spread of ProbD method declined significantly under the WIC model. The reason is described as follows: in WIC model, the influence probability that node u affects node υ is the reciprocal of υ's degree. When two large-degree nodes directly connect with each other, the influence probability between them is very small. In other words, WIC model is not sensitive to the rich-club phenomena and the rich-club effect is relatively weak. Thus, a large-degree node can still bring a considerable influence spread even if its neighbors



(a) NetScience (b) Yeast

(c) Power (d) CaGrQc

(e) NetHept (f) Brightkite

(g) Wordnet

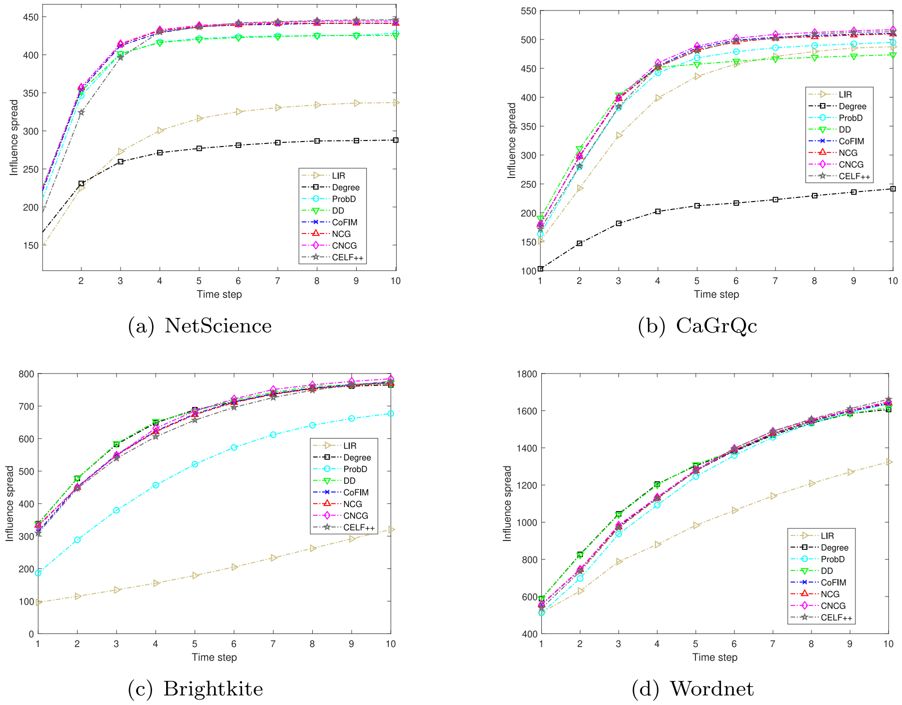
Fig. 3. The influence spread of eight algorithms on seven networks with the variation of influence probability under the UIC model, seed nodes size $k = 50$.

have already been selected as seeds. However, ProbD method will delete one node's neighbors directly after it is selected as seed. This simple strategy is not suitable for WIC model, leading to the degradation of influence spread. The similar phenomena has been found in our previous works [18] and [15].

As Fig. 5 shown, the influence spread of some compared algorithms is rather close on some networks. For simplicity, we only detail the numerical results of CaGrQc network (Fig. 5(d)) for a clearer comparison. Table 3 shows the influence spread of CoFIM, CELF++, NCG and CNCG methods on this network. We use the bold font to mark the maximum value. For CaGrQc network, CELF++ and CNCG are top-two algorithms for almost all seed nodes size k and CNCG is slightly better than CELF++ when k is bigger.

4.5. Test on overlapping nodes

In this subsection, we further analyze the significance of the overlapping nodes in our proposed approach by a case study on the NetHept network. As already introduced, NCG views the whole network as a single community. In other words, none of the seed nodes in NCG would benefit from the overlapping bonus in Definition 2. However, CNCG first detects the overlapping community structure and then calculates the node coverage. If



(a) NetScience

(b) CaGrQc

(c) Brightkite

(d) Wordnet

Fig. 4. The spreading speed of eight algorithms on four networks under the UIC model, $k = 50$.

Table 3
The influence spread of four algorithms on CaGrQc network.

k	5	10	15	20	25	30	35	40	45	50
CoFIM	**130.96**	203.02	298.91	371.97	436.02	501.52	556.30	608.82	656.53	704.02
CELF++	130.55	**210.52**	**299.26**	**372.53**	**439.76**	502.53	**559.49**	609.38	658.05	715.68
NCG	129.15	209.13	284.33	367.96	430.49	500.85	557.03	603.57	653.13	700.22
CNCG	129.13	206.98	292.70	371.06	437.44	**503.53**	558.95	**610.29**	**665.00**	**717.79**

a node is an overlapping node, its coverage would be the union from different communities, which has an advantage compared with nodes from a single community.

NetHept network has 15 233 nodes, while 5514 of them are identified as overlapping nodes in our proposed CNCG method. We compare the seed nodes selected by CNCG and NCG to reveal whether would CNCG select more overlapping nodes and thus improve the influence spread. With the increasing of k, we count how many overlapping nodes are selected as seed nodes by NCG and CNCG. Corresponding Results are shown in Table 4.

As shown in Table 4, when k is 50, NCG selects 6 overlapping nodes while CNCG selects 43 overlapping nodes. It implies that overlapping nodes are more frequently selected in our proposed CNCG method, which is in accordance with Definition 2. Besides, the influence spread of CNCG is almost always better than NCG on NetHept network (Figs. 2(e), 3(e) and 5(e)), no matter the influence probability or the spreading model. Especially when k is 5 under the WIC model, the influence spread of CNCG is 181.28 while that of NCG is only 160.83.

Therefore, putting emphasis on these overlapping nodes indeed improves the accuracy of seed selection and the performance of influence spread.

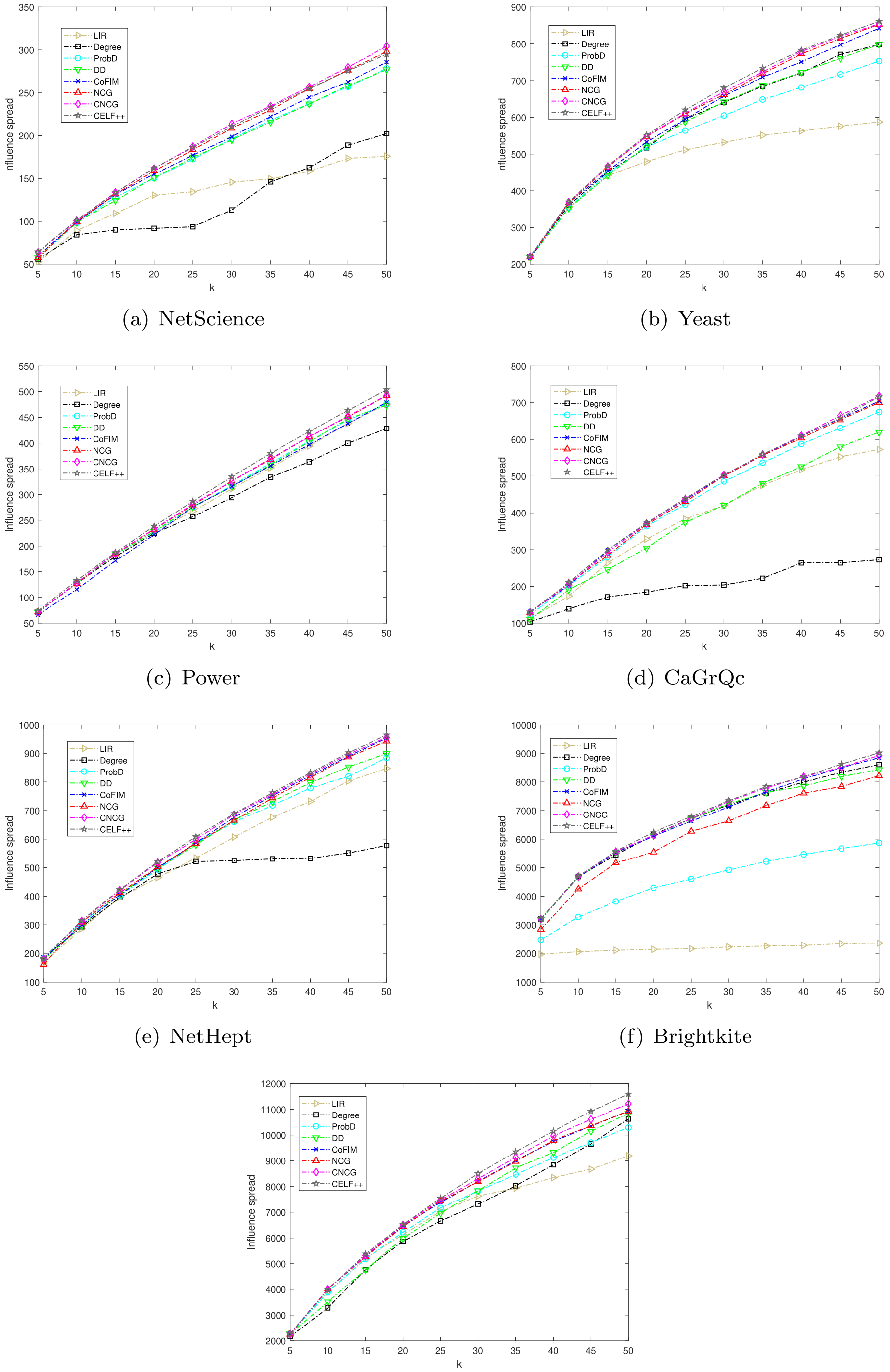
4.6. Running time comparison

In order to further compare the performance of these methods, Fig. 6 lists the running time of different algorithms on Brightkite and Wordnet networks. All methods are implemented in Matlab R2018b.

As shown in Fig. 6, the CELF++ is the most time-consuming. For CNCG method, it needs to calculate the shortest path between nodes when constructs the topological potential field, which takes the most time of the method. Therefore, CNCG needs more running time than other three heuristic methods (LIR, ProbD and DD) and slightly slower than CoFIM. However, compared with NCG which views the whole network as a single community, the overall time of CNCG is reduced significantly although we need to pay extra time to partition whole graphs into communities. The reason is described as follows: after community partition, the node coverage gain of each node is evaluated within its local community rather than the whole network, leading to a considerable decline of node coverage gain calculation complexity.

5. Conclusions

Developing effective algorithms for influence maximization is still a challenging problem. In this paper, we propose an influence maximization approach based on overlapping community detection and node coverage gain evaluation. We first partition the social graph into overlapping communities with an algorithm of node location analysis in topological potential field and then evaluate the influence of each node locally through node coverage gain. The seed nodes are identified by exploiting the detected community structure and pre-designed seed selection strategy. The experimental results under the UIC model and WIC model prove that our proposed approach can accurately identify the influential seeds to maximize the influence spread. Especially, we show that overlapping nodes are important for the IM problem. Putting emphasis on these overlapping nodes cold improve



(a) NetScience

(b) Yeast

(c) Power

(d) CaGrQc

(e) NetHept

(f) Brightkite

(g) Wordnet

Fig. 5. The influence spread of eight algorithms on seven networks with the variation of seed nodes size k under the WIC model.

Table 4
The number of overlapping nodes and influence spread on NetHept network.

k	5	10	15	20	25	30	35	40	45	50
NCG	0	1	2	2	3	4	5	6	6	6
UIC	289.78	339.08	377.21	408.89	434.90	455.91	477.06	503.97	523.79	541.72
WIC	160.83	309.08	412.47	502.27	585.33	665.39	743.90	815.91	887.39	942.36
CNCG	3	8	13	18	22	26	29	34	39	43
UIC	288.93	340.13	377.73	409.50	434.63	455.05	476.08	505.54	524.88	547.14
WIC	181.28	313.31	421.54	517.87	598.88	686.79	756.18	827.43	896.88	955.69

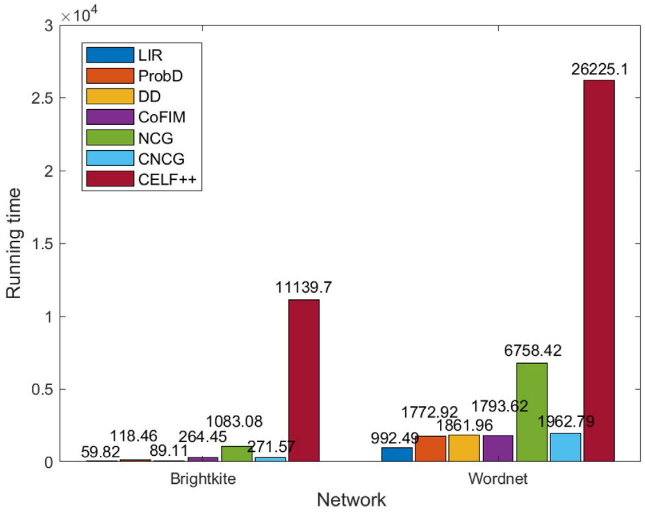


Fig. 6. Running time of different methods on Brightkite and Wordnet networks.

the accuracy of seed selection and the performance of influence spread. In the future, we will design a more efficient parallel algorithm to reduce the running time of the shortest path calculation between nodes, thus, we can further improve the efficiency of our proposed CNCG approach.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was supported by the Fundamental Research Funds for the Central Universities, China (grant No. 2019XKQYMS85).

References

[1] S. Peng, Y. Zhou, L. Cao, S. Yu, J. Niu, W. Jia, Influence analysis in social networks: A survey, J. Netw. Comput. Appl. 106 (2018) 17–32.

[2] M. Li, X. Wang, K. Gao, S. Zhang, A survey on information diffusion in online social networks: Models and methods, Information 8 (4) (2017) 118.

[3] Y. Li, J. Fan, Y. Wang, K.-L. Tan, Influence maximization on social graphs: A survey, IEEE Trans. Knowl. Data Eng. 30 (10) (2018) 1852–1872.

[4] S.S. Singh, A. Kumar, K. Singh, B. Biswas, C2im: Community based context-aware influence maximization in social networks, Physica A 514 (2019) 796–818.

[5] X. Li, X. Cheng, S. Su, C. Sun, Community-based seeds selection algorithm for location aware influence maximization, Neurocomputing 275 (2018) 1601–1613.

[6] H. Huang, H. Shen, Z. Meng, Community-based influence maximization in attributed networks, Appl. Intell. 50 (2) (2020) 354–364.

[7] S. Chen, J. Fan, G. Li, J. Feng, K.-l. Tan, J. Tang, Online topic-aware influence maximization, Proc. VLDB Endow. 8 (6) (2015) 666–677.

[8] S. Banerjee, M. Jenamani, D.K. Pratihar, Combim: A community-based solution approach for the budgeted influence maximization problem, Expert Syst. Appl. 125 (2019) 1–13.

[9] A. Bozorgi, S. Samet, J. Kwisthout, T. Wareham, Community-based influence maximization in social networks under a competitive linear threshold model, Knowl.-Based Syst. 134 (2017) 149–158.

[10] M. Khajehnejad, A.A. Rezaei, M. Babaei, J. Hoffmann, M. Jalili, A. Weller, Adversarial graph embeddings for fair influence maximization over social networks, in: Proceedings of the 29th International Joint Conference on Artificial Intelligence, 2020.

[11] D. Kempe, J. Kleinberg, É. Tardos, Maximizing the spread of influence through a social network, in: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2003, pp. 137–146.

[12] W. Chen, Y. Wang, S. Yang, Efficient influence maximization in social networks, in: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2009, pp. 199–208.

[13] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, N. Glance, Cost-effective outbreak detection in networks, in: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2007, pp. 420–429.

[14] A. Goyal, W. Lu, L.V. Lakshmanan, Celf++ optimizing the greedy algorithm for influence maximization in social networks, in: Proceedings of the 20th International Conference Companion on World Wide Web, 2011, pp. 47–48.

[15] X. Rui, X. Yang, J. Fan, Z. Wang, A neighbour scale fixed approach for influence maximization in social networks, Computing 102 (2) (2020) 427–449.

[16] D. Liu, Y. Jing, J. Zhao, W. Wang, G. Song, A fast and efficient algorithm for mining top-k nodes in complex networks, Sci. Rep. 7 (2017) 43330.

[17] D.-L. Nguyen, T.-H. Nguyen, T.-H. Do, M. Yoo, Probability-based multi-hop diffusion method for influence maximization in social networks, Wirel. Pers. Commun. 93 (4) (2017) 903–916.

[18] X. Rui, F. Meng, Z. Wang, G. Yuan, A reversed node ranking approach for influence maximization in social networks, Appl. Intell. 49 (7) (2019) 2684–2698.

[19] S. Zhou, R.J. Mondragón, The rich-club phenomenon in the internet topology, IEEE Commun. Lett. 8 (3) (2004) 180–182.

[20] Y. Wang, G. Cong, G. Song, K. Xie, Community-based greedy algorithm for mining top-k influential nodes in mobile social networks, in: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2010, pp. 1039–1048.

[21] S. Cheng, H. Shen, J. Huang, G. Zhang, X. Cheng, Staticgreedy: solving the scalability-accuracy dilemma in influence maximization, in: Proceedings of the 22nd ACM International Conference on Information & Knowledge Management, 2013, pp. 509–518.

[22] K. Jung, W. Heo, W. Chen, Irie: Scalable and robust influence maximization in social networks, in: 2012 IEEE 12th International Conference on Data Mining, IEEE, 2012, pp. 918–923.

[23] S. Rani, M. Mehrotra, Community detection in social networks: Literature review, J. Inf. Knowl. Manage. 18 (02) (2019) 1950019.

[24] K. Rahimkhani, A. Aleahmad, M. Rahgozar, A. Moeini, A fast algorithm for finding most influential people based on the linear threshold model, Expert Syst. Appl. 42 (3) (2015) 1353–1361.

[25] E. Bagheri, G. Dastghaibyfard, A. Hamzeh, Fsim: A fast and scalable influence maximization algorithm based on community detection, Int. J. Uncertain. Fuzziness Knowl.-Based Syst. 26 (03) (2018) 379–396.

[26] J. Shang, S. Zhou, X. Li, L. Liu, H. Wu, Cofim: A community-based framework for influence maximization on large-scale networks, Knowl.-Based Syst. 117 (2017) 88–100.

[27] J. Shang, H. Wu, S. Zhou, J. Zhong, Y. Feng, B. Qiang, Impc: Influence maximization based on multi-neighbor potential in community networks, Physica A 512 (2018) 1085–1103.

[28] A. Bozorgi, H. Haghighi, M.S. Zahedi, M. Rezvani, Incim: A community-based algorithm for influence maximization problem under the linear threshold model, Inf. Process. Manage. 52 (6) (2016) 1188–1199.

[29] M. Jalayer, M. Azheian, M.A.M.A. Kermani, A hybrid algorithm based on community detection and multi attribute decision making for influence maximization, Comput. Ind. Eng. 120 (2018) 234–250.

[30] W. Zhi-Xiao, L. Ze-chao, D. Xiao-fang, T. Jin-hui, Overlapping community detection based on node location analysis, Knowl.-Based Syst. 105 (2016) 225–235.

[31] A. Arora, S. Galhotra, S. Ranu, Debunking the myths of influence maximization: An in-depth benchmarking study, in: Proceedings of the 2017 ACM International Conference on Management of Data, 2017, pp. 651–666.

[32] Y. Li, J. Fan, Y. Wang, K.-L. Tan, Influence maximization on social graphs: A survey, IEEE Trans. Knowl. Data Eng. 30 (10) (2018) 1852–1872.

[33] Z. Wang, Z. Li, G. Yuan, Y. Sun, X. Rui, X. Xiang, Tracking the evolution of overlapping communities in dynamic social networks, Knowl.-Based Syst. 157 (2018) 81–97.

[34] S. Brin, L. Page, The anatomy of a large-scale hypertextual web search engine, Comput. Netw. ISDN Syst. 30 (1998) 107–117.

[35] J. Ok, Y. Jin, J. Shin, Y. Yi, On maximizing diffusion speed in social networks: impact of random seeding and clustering, in: The 2014 ACM International Conference on Measurement and Modeling of Computer Systems, 2014, pp. 301–313.

[36] F. Radicchi, C. Castellano, Fundamental difference between superblockers and superspreaders in networks, Phys. Rev. E 95 (1) (2017) 012318.

Zhixiao Wang received his Ph.D. degree in the Department of Computer Science and Engineering at Tongji University in 2011. Currently, he served as a professor at School of Computer Science and Technology in China University of Mining and Technology. He has published more than 30 papers in international conferences and journals. His research interests include complex network, social network analysis and data mining.

Chengcheng Sun is a Ph.D. candidate at School of Computer Science and Technology, China University of Mining and Technology. His research interests include social network analysis, graph representation learning and data mining.

Jingke Xi is an Associate Professor of College of Computer Science and Technology, China University of Mining and Technology. He received his Ph.D. degree at China University of Mining Technology in 2012. His research interests include community detection and data mining.

Xiaocui Li received the B.S. degree in computer science and technology, and the M.S. degree in computer application technology from the China University of Mining and Technology, Xuzhou, China, in 2010, and 2013, respectively. She is currently pursuing the Ph.D. degree in data mining and privacy protection from the Huazhong University of Science and Technology, Wuhan, China. Her current research interests include clustering algorithm, machine learning. She has published papers in WWWJ, DASFAA and NCAA.