Jing and Liu *EURASIP Journal on Wireless Communications and Networking* (2019) 2019:142
https://doi.org/10.1186/s13638-019-1405-5

EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

Context-based influence maximization with privacy protection in social networks

Dong Jing and Ting Liu*

Abstract

As the increase of requirements of accessing and sharing information of people, large social networks have appeared. The influence maximization over social network has been a popular research topic, whose goal is to maximize the expected range of influence by selecting seed nodes to sending information and encouraging nodes in social network to report the messages. On the other side, privacy concerns have became more and more important, both automated and manual efforts are utilized to protection privacy of users. Under the mechanisms of privacy protection, the nodes in social network will not act as they did in the setting of no privacy protection. As far as we know, there are no previous works considering this problem. In this paper, we consider the influence maximization problem with privacy protection mechanisms in social networks.

One challenge is that how to abstract the relations between users and information to identify which kinds of information should be protected by the privacy-related mechanisms. A context-based solution is proposed in the paper to face the challenge above and solve the influence maximization problem. First, a context-based information diffusion model (IDC for short) is proposed. Then, the corresponding influence maximization problem (IM-IDC for short) under IDC model is formally defined. Then, the methods about context extraction, influence estimation, and redundant contexts identification are introduced. The IM-IDC problem is shown to be NP-hard, and an efficient approximation algorithm based on greedy strategy is proposed and analyzed theoretically. Finally, experimental results show that our method is efficient.

Keywords: Influence maximization, Privacy, Context

1 Introduction

Recently, as the increase of requirements of accessing, sharing, and sending information of people, social networks have appeared. The developments of techniques of social network have huge increase and rapid popularization in the whole world. They have changed the communication ways of people. Typical social network applications include Friendster, Twitter, Facebook, Sina Weibo, and so on. As the used of mobile cell phones increase, more and more people are involved in the social network. According to the reports, until 2015, the size of mobile phones users in the world has reached 4.45 billion and 42.9% of them are using smart phones (about 2 billion). With the reports from China, the size of mobile smart phone users has reached 0.5 billion in 2014. As the size and power increases of smart mobile devices, more and more online social applications are being used in mobile environments, social networks become larger and larger.

In the social networks, since the way of diffusing information changed, many human activities need to adapt to such change. A key application is using social networks for viral marketing as shown in [1]. Different from traditional methods for marketing, viral marketing can utilize the "word-of-mouth" advantages of social networks and diffuse advertising information more efficiently. It has attracted lots of research interests. During the market procedure, people with high influence will be expected to send the advertising information to. People with higher influence can affect other people with more chances. Influence maximization problem is one of the most popular topics in the area of influence research in social networks. It has been formally investigated in [2] and obtained lots of attentions from many researchers (such as [3–5]). However, there are still important challenges

*Correspondence: tliu@ir.hit.edu.cn
Department of Computer Science and Technology, Harbin Institute of Technology, 92 West Dazhi Street, 150001 Harbin, China

not solved in real applications of influence maximization problem when facing more practical scenarios. One of them is that the influence ability between nodes in real world may be affected by privacy protection mechanisms. Since data privacy has been more and more important, not only automatic tools but also manual efforts are taken to protect privacy of users. In that cases, even if one user is influenced by the information very much, because of privacy considerations, he may not repost the message. In details, essentially, under the privacy protection mechanisms in social applications, people are allowed to select *what they want to send* or *what they are saying about* by filtering messages by labels or adding labels to the message. In these cases, it is more feasible to consider a context-based or label-based privacy protection and the probability that node u affects v on special context c rather than the probability that u affects v. That is, the information diffusion procedure is affected by the privacy protection mechanisms based on contexts essentially. As far as we know, there are only few works that consider the influence maximization problem under multiple factors or related problems under privacy, no previous works focus on the influence maximization problem under context-based privacy protection mechanisms.

In the general model, a social network is composed of nodes and edges like general networks. Each node is a social actor and the edge between two nodes represent the relations between them (e.g., following, followed-by). The procedure of information diffusion in general social networks can be explained by the followings. Assume that some node A has accepted some information I (e.g., A has bought some product introduced by I), A will have chances to influence some node B within its neighbors. The *chance* is modeled by an influence probability between A and B. For example, if the probability between them is 0.9, B will accept I and transfer I to its neighbors also in the next step. In this model, the influence maximization problem is to find a special seed nodes set S such that the expected number of nodes affected by S can be maximized. Here, the influence probability is a simple and clear method to describe the ability of influence between two nodes, and it is usually assumed to be a constant one if a special social network is considered. However, in real applications of social networks, the influence probability may depend on the context and may be affected by the privacy considerations.

Let us consider an example in practical applications. Assume that there is a user x of Twitter, y is a friend of x. In real life, x is a college teacher for computer science. At some time, x bought two books, one is about programming and the other one is about literature. Then, x sent two tweets about the two books with pictures and comments. When y saw the tweet about programming book, if x did not publish the information about his career, y is possible to guess that x belongs to computer science area. Even if x has shared the information of his interests by adding labels such as *computer science*, y may guess that x is doing a job needing lots of knowledge by analyzing the tweet about literature book. Also, on the other side, considering that y is a close friend of x, y may decide to repost the tweet about programming, since y trust x in computer science area because of the major of x. But when y saw the tweet about literature book, the probability that y reposts the tweet becomes lower because he does not think x as a professional one for literature. Here, different from the cases assumed by traditional models for information diffusion, privacy protection mechanisms may affect the information diffusion procedure and the way of affecting influence probability may depend on the categories of the information. On the other side, even if x and y are close enough to be insensitive to the private information of each other, it cannot be expected that y always reposts messages from x in the same high probability. It will also depend on which kind of information is diffusion. Furthermore, if we can identify those kinds of information protected by privacy considerations, or interested by other users, we can refine the model more.

Actually, the examples above show two important challenges of privacy considerations in the information diffusion model. The first one is how to identify which kinds of information should be protected or specially interested by other nodes. The otehr one is how to solve the influence maximization problem under multiple kinds considerations.

In this paper, we address the problem of maximizing influence with privacy considerations in social networks. Obviously, because of privacy considerations, not all affected nodes will try to affect others by reposting the information, one possible case is that they are influenced but stop to repost the information. The main challenge is that how to abstract the relations between users and information to identify which kinds of information should be protected by the privacy-related mechanisms. A context-based solution is proposed in the paper to face the challenge above and solve the influence maximization problem. To solve the influence maximization problem efficiently, we study its computational complexities and design efficient algorithms. The main contributions can be summarized as follows:

1 We identify the influence maximization problem under privacy protection mechanisms as new challenges of information diffusion in social networks. To overcome them, we propose new information diffusion models to support context-based privacy description and formulate the new influence maximization problem-based on new model.

2 We show the hardness for context-based influence maximization problem. It is achieved by proving that classic influence maximization problems is a special case of the new problem.
3 We design efficient approximation algorithms for the new influence maximization problem. By showing the monotone and submodular properties of the new problem, a $(1 - 1/e)$ approximation algorithm can be obtained.
4 The experimental results on real datasets show that the proposed method can efficiently solve the information diffusion problem with privacy considerations in social networks.

The rest of the paper are organized as follows. In Section 2, some preliminaries and new definitions about the information diffusion model under privacy considerations will be introduced. Then, in the Section 3, the methods about context extraction, influence estimation, and redundant contexts identification are introduced. Then, in Section 4, theoretical analysis and approximation algorithms for influence maximization problems are introduced. Extensions and optimizations about the algorithms is shown in Section 5. Experimental results are shown in Section 6. In Section 8, the related works are discussed. Finally, Section 9 concludes the paper.

2 Context-based information diffusion model with privacy consideration

In this section, general information diffusion models are introduced first, then, to consider the effects of privacy protection, a context-based information diffusion model is proposed, finally, we give the formal definition of the corresponding influence maximization problem.

2.1 General information diffusion models

In this paper, information diffusion can be described as the propagating procedure of information over some network. A network is usually denoted by a graph $G(V, E)$. Here, V is the node set where each node represents one person or entity, and E is the edge set where each edge represents the relation (cooperation, friends, enemies, and so on) between two nodes. Each node is associated with *active* or *inactive* state. Intuitively, the active state means that the node has been affected. The active set of nodes may affect the nodes in inactive set and the influence ratio can describe the strength of that affection. If some inactive node is affected by some *active* node so much that the inactive becomes *active*, such a process is called *activation*. Intuitively, for some node v, the more of the neighbors of v are activated, more likely v will be activated. After then, v will affect more nodes further. As such procedures repeat, more and more nodes will become active. The procedure of activation cannot be reversed: one node can transform from inactive state to active state, but not vice versa. To design proper theoretical model to describe information diffusion in real world, the key is to explain how the interactions between nodes work. Next, we introduce two popular information diffusion models.

Linear threshold model. Given a network $G(V, E)$, let $N(v)$ be the set $\{u|(u, v) \in E\}$. For each $(u, v) \in E$, a threshold value b_{uv} is utilized to represent the degree of influence from u to v. For each node v, it is satisfied that $0 \leq \sum_{u \in N(v)} b_{u,v} \leq 1$. During the procedure of information diffusion, another threshold value θ_v with respect to each node v is used to control the diffusion of information. In detail, at some instant time, let $A(v)$ be the set of v's neighbor nodes which have been active. If $\sum_{u \in A(v)} b_{u,v} \geq \theta_v$, v will become active. In this model, when node u tries to activate its neighbor v and fails, the influence $b_{u,v}$ is remembered and will be accumulated in the following activating steps. In other words, the influence from u to v will not be ignored, even if the activation is failed. As we will see in the following part, the influence is treated differently in other models. The whole procedure of information diffusion in linear threshold model can be described as follows. First, an initial active node set S_0 will be activated. Then, in the ith step of information diffusion, based on the active nodes in S_{i-1}, the influence for each node in $V \setminus S_{i-1}$ will be computed. According to the influence computed and the θ_v for each node v, all nodes satisfying $\sum_{u \in A(v)} b_{u,v} \geq \theta_v$ will be put in S_i. Repeat these steps until no more nodes can become active.

Independent cascade model. Independent cascade model is a probabilistic model. Instead of b_{uv} in linear threshold model, this model uses p_{uv} to describe the probability that u can activate v in a single activation. The whole procedure of information diffusion under independent cascade model can be described as follows. First, an initial node set S_0 will be set to be active. Then, in the ith step, every node will try to activate their neighbors. In detail, for each node $u \in S_{i-1}$ and node $v \in V \setminus S_{i-1}$, if $(u, v) \in E$, v will be activated once in probability p_{uv}. If v indeed becomes active, it will be added to S_i and not be further considered in current step. Repeat this procedure until that no new nodes are added. It should be noted that p_{uv} is only determined by u and v and is independent with other node pairs. In this model, each edge (u, v) will be considered only one time. Once it fails, this edge will never be considered. In [2], an extended model in which p_{uv} will be decreased as time goes by.

2.2 Context-based privacy protection mechanism

It has been a huge requirement to protect information privacy, especially in the area of social network. Without privacy protection mechanism, sensitive information of users may be easy to be obtained by illegal applications. More and more operation systems tend to provide core mechanism to protect information privacy when users try to send out data to the network, the protection mechanism is transparent to users. Also, general users learn more and more knowledge about how to protect their privacy by sending only insensitive information.

There have been lots of previous works focusing on the techniques of privacy protection mechanisms, such as perturbation [6], randomization [7], k-anonymity [8], and difference privacy [9]. Most of them have one common feature, that is to hide single user data or answer from other ones. The main intuitive idea is to let the item protected seem to be no difference from other items.

In the procedure of information diffusion in social network, the actions of reposting some message may leak private information of users. In fact, when some user A receives some message M, the actions A taking will answer the query "is A interested in M?." If A decides to repost M, the answer is *yes*, otherwise, the answer is *no*. By collecting such answers, private information (e.g., interests, majors) of A can be obtained using information techniques. Essentially, during this procedure, to provide the privacy protection mechanisms, it needs to modify the answers (the decisions about whether to repost the message received) in certain way such that A will not show obvious difference from other users. Therefore, the privacy protection mechanisms will affect the probability that A can activate other users, that is there are cases A will not repost M to avoid leaking private information even if A has been activated by M. To describe the procedure of diffusing information under privacy protection mechanism, we propose a *context* -based representation of privacy protection mechanism first, then the corresponding information diffusion model is introduced in the following part.

Here, a context set $C = \{c_1, c_2, \ldots, c_m\}$ is used to specify which kind of class the information belongs to. We consider a privacy protection model as follows. Each user is attached to several contexts; they can represent his interests which can be obtained based on tags labeled by himself. Also, each message is attached with contexts, which can be obtained by getting the tags labeled by the produced user or analyzing it by language processing algorithms. Intuitively, when a user A receives some message M, even if A indeed is interested in M and wishes to share M with other uses in the network, it is very possible that A determines not to repost M because of privacy considerations. There are mainly three kinds of such considerations as follows:

- First, if the context sets of M and A are similar, A tends to repost M in high probability. In this case, even if attackers know that A reposts M, since the interest tags are public information to all users, only little private information will be leaked. This protection strategy is that A only gives the answers which others know A will give.
- Second, if many neighbors of A repost M, A tends to repost M in high probability. In this case, even if attackers know that A reposts M, since many other users also do the same actions, intuitively, only *non-private* information will be leaked. This protection strategy is that A only gives the answers which other ones will also give.
- Third, if there is some item in the context set of M which seems to be different from the context set of A, A will not repost M. This protection strategy is that A will not give the answers that the potential attackers do not know A will give.

2.3 Context-based information diffusion with privacy protection strategies

In this part, to integrate privacy protection strategies with the procedure of information diffusion, a context-based information diffusion model (IDC for short) is proposed. In IDC model, the social network can be represented by a graph $G = (V, E)$. Here, V is the node set and E is the directed edge set which represents the influence relationship between nodes in the network. Intuitively, if there is an edge $(u, v) \in E$, it says that v can be influenced by u. That is, if u has been influenced, v also may be influenced through the edge (u, v).

In the IDC model, a finite context set $C = \{c_1, c_2, \ldots, c_m\}$ is used to represent the context information. Intuitively, to support privacy protection strategies during information diffusion, the relationships between users and messages must be identified. However, it is impossible to build a model for each piece of message, a better solution is to put messages into different categories by analyzing their characteristics. The context information is just used to identify which category the message belongs to. For example, the contexts of a tweet like "LeBron James has agreed to a four year, 154 million dollars contract with Los Angeles Lakers" may be {LeBron James, Los Angeles Lakers, Basketball}. Also, we use context information to represent what kinds of information some node is interested in. For example, in real applications such as tweet, the tags of users can be viewed as the contexts. Ideally, each item in C should be *totally* different from others, such that we can use a C with minimum size to import context information to the procedure of information diffusion. Each node v in V may be associated with a context set $C_v \subseteq C$, which represent v is sensitive or interested those information generated by the contexts in C_v. In IDC model, also, for each special message M, different from general information diffusion models, the information M should also be specified with a context set $C_M \subseteq C$, which means that information M is related with all context in C_M.

In general information diffusion model, for each edge (u, v) in the network, a probability or threshold is assigned to represent the influence, which is usually defined to be a function $p : E \mapsto [0, 1]$. Differently, the influence function p is extended to be a function set $P = \{p_1, \ldots, p_m\}$, where each function corresponds to a special context c and for edge (u, v) the value of p^c_{uv} is the measure of influence of u on v. Therefore, for each two nodes u and v satisfying $(u, v) \in E$, given special information I, whether v will be influenced by u is determined by the contexts of I, the influence probability functions of different contexts maybe different. For example, suppose A and B are two tweet users, it is a usual case that *A trusts B* in *IT* area but not in *music* area, so the probability that v is influenced by some IT information re-tweeted by u is much higher than by some music-related information.

Context-based information diffusion model. Formally speaking, in IDC model, an information diffusion model can be described by a 5-tuple $\langle G, C, P, U, \Theta \rangle$, where $U : V \mapsto 2^C$ is defined to be the function to compute the context set for each user and $\Theta = \{\theta_i | 1 \leq i \leq |V|\}$ satisfying $0 \leq \theta_i \leq 1$. The procedure of information diffusion in IDC model can be explained as follows.

In IDC model, given a network $G = (V, E)$, a context set C, two functions P and U, a threshold set Θ, a message I with context set $C_I \subseteq C$ and a seed node set A, the information diffusion process working in discrete time can be explained as follows. Here, we use t_0, t_1, $\cdots$, t_n to represent the discrete times. For each node $v \in V$, there are three different states: inactive, active, and progressive.

- Initially, at time t_0, all nodes in A will become active and inserted into the set Z, and all other nodes will be initialized to be inactive.
- At time t_i, each active node u which is just activated will determine whether to go from active to *progressive* in following steps. First, u will continue with the next step with probability
$$p_1(u, I) = \frac{|C_u \cap C_I|}{|C_u|}, \text{where } C_u = U(u), \quad (1)$$
otherwise, u will keep in active state later. Then, the following threshold value will be calculated
$$f_1(u) = \frac{|\hat{N}_u| + 1}{|N_u| + 1}, \quad (2)$$
where N_u is the set of neighbor node of u and $\hat{N}_u$ is the set of progressive nodes in N_u. If $f_1(u) \geq \theta_u$, u will become progressive, otherwise, u will keep in active later.
- At time t_i, after changing the states of some nodes from active to progressive, all nodes which just became progressive will try to activate their neighbor nodes in inactive state. In detail, suppose node u just became progressive and v is an inactive neighbor node of u before t_i. For each context $c \in C_I$, v will be influenced by u in the probability p^c_{uv}. If, for all contexts in C_I, v has been influenced by some node (maybe by other nodes in previous steps), v will become active. Then, v will be added to the set Z.
- The procedures above iterate until no new nodes can be added into Z. Finally, Z will be the influenced set of A under the IDC model $\langle G, C, P, U, \Theta \rangle$. During the whole procedure, it should be noted that the node state can transform from inactive to active, from active to progressive, but not vice versa.

The intuitive idea of context-based information diffusion model can be explained as follows. There are several factors affecting the actions of some node on special information, each factor can be abstracted by the relationship between node and information. The relationship is abstracted by context, e.g., tag information in real applications. For special information I, if I is interesting enough on all related contexts to node u, u will accept I and become active. If I is "similar" enough with u and there are enough neighbor users which accept I, u will tend to become progressive and try to activate other users.

Let us consider an example of IDC model shown on the top left corner of Fig. 1. Assume that we have an ID model $\langle G, C, P, U, \Theta \rangle$ for some network. The network G is shown in Fig. 1, which include five nodes and seven edges. The context set C is

$$\{\text{basketball, LeBron James, Michael Jordan}, \ldots\}.$$

We use c_1, c_2, c_3, and so on to represent them. The function set $P = \{p^1, p^2, p^3, \ldots\}$ can be represented as follows, without loss of generality, focusing on the first two contexts in this example, we only describe p^1 and p^2.

$$p^1_{AB} = 0.6; \quad p^2_{AB} = 0.9;$$
$$p^1_{AC} = 0.7; \quad p^2_{AC} = 0.8;$$
$$p^1_{CD} = 0.1; \quad p^2_{CD} = 0.9;$$
$$p^1_{BD} = 0.9; \quad p^2_{BD} = 0.1;$$
$$p^1_{BE} = 0.8; \quad p^2_{BE} = 0.02;$$

The definition of function U is as follows:

$$U(A) = U(B) = U(C) = \{c_1, c_2, c_3\}$$
$$U(D) = \{c_2, c_4, c_5, c_7\}, U(E) = \{c_2\}$$

The definition of threshold set Θ is as follows:

$$\theta_A = 0.1; \theta_B = \theta_C = 0.5; \theta_D = \theta_E = 0.4$$

Given $\{A\}$ as the seed node set, assume that a message m associated with contexts $C_I = \{c_1, c_2\}$, an example of information diffusion procedure is shown in Fig. 1. In the first step t_0, node A will be initialized to be active. Then,

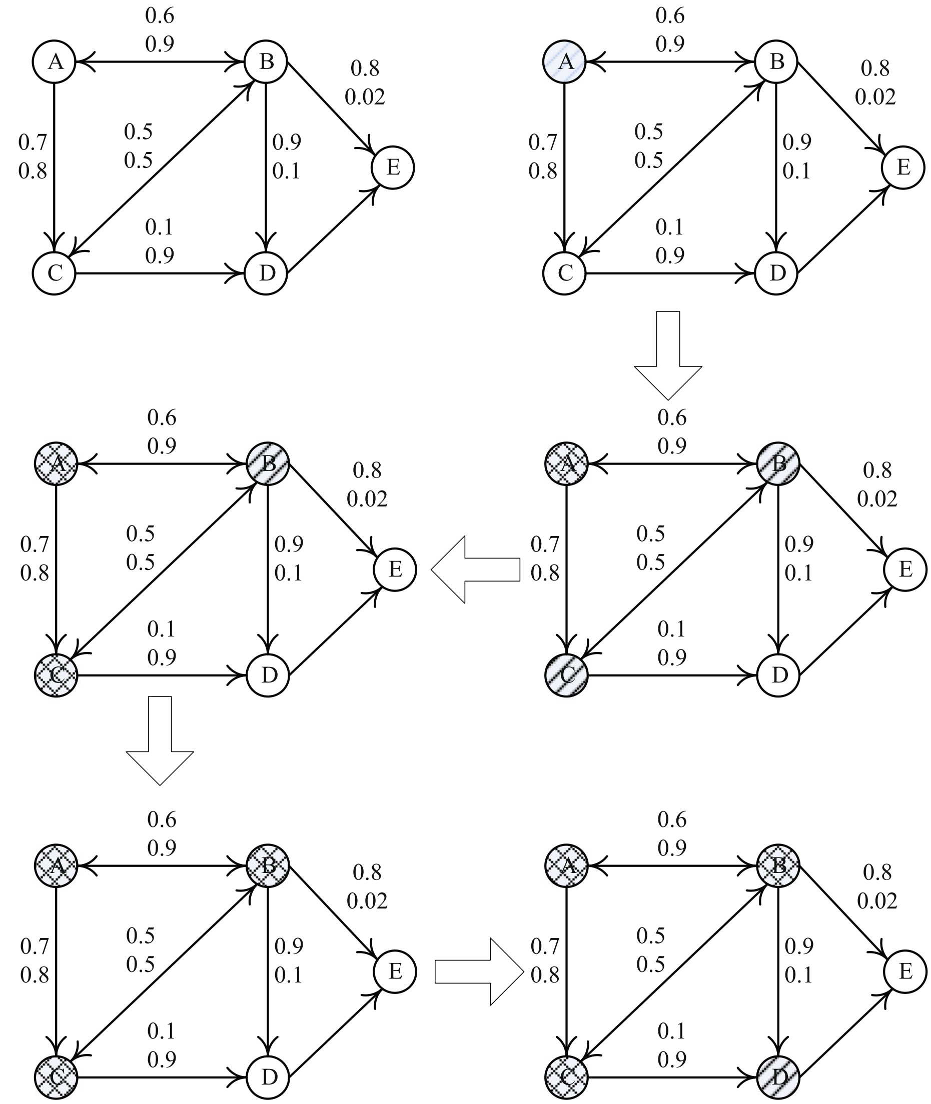

Fig. 1 An example of IDC model and the procedure of information diffusion in IDC model

$p_1(A,m) = \frac{|C_I \cap U(A)|}{|U(A)|} = \frac{2}{3}$, A will continue to calculate $f_1(A) = \frac{1}{3}$ with probability $\frac{2}{3}$. Suppose A successes to compute $f_1(A)$, since $f_1(A) > \theta_A = 0.1$, A will become progressive in the next step. In the second step t_1, two edges connected with A will be processed. For the edge (A, C), since $p^1_{AC} = 0.7$ and $p^2_{AC} = 0.8$, node C will be tried to be activated in two steps. Assume that the first random value generated is 0.85, because $0.85 > p^1_{AC} = 0.7$, C will be activated in the first context. Also, assume that C is also activated in the second context. Then, C will become active. For the edge (A, B), because $p^1_{AB} = 0.6$ and $p^2_{AB} = 0.9$, the node B will be tried to be activated by two contexts. Assume that the two random values generated for the two contexts are 0.7 and 0.9, the node B will become active also. Then, we have

$$p_1(B,m) = \frac{|C_I \cap U(B)|}{|U(B)|} = \frac{2}{3} \text{and}$$

$$p_1(C,m) = \frac{|C_I \cap U(C)|}{|U(C)|} = \frac{2}{3}.$$

Assume that B and C are both success in this step. Then, we have

$$f_1(B) = \frac{|\{A\}| + 1}{|\{A, C, D, E\}| + 1} = \frac{2}{5} \text{and}$$

$$f_1(C) = \frac{|\{A\}| + 1}{|\{A, B, D\}| + 1} = \frac{2}{4}.$$

Because $f_1(B) < \theta_B = 0.5$, B will stay in active state, and C will become progressive observing that $f_1(C) \geq \theta_C = 0.5$. In the third step t_2, C will try to activate the node D on two contexts in the probability $p^1_{CD} = 0.1$ and $p^2_{CD} = 0.9$.

Assume that D is activated on the second context, but D will not become active since only one context is activated. Now, since C has been progressive in the last step, we have

$$f_1(B) = \frac{|\{A, C\}| + 1}{|\{A, C, D, E\}| + 1} = \frac{3}{5}.$$

Therefore, B will become progressive. In the forth step t_3, the node B will try to activate D and E. Assume that D is activated on the first context, then D will become active. Also, since $p^2_{BE} = 0.02$ and $p^1_{BE} = 0.8$, assume that only the first context of E is activated. In the fifth step t_4, assume that the node D fails to continue calculate $f_1(D)$, D will stay in the state active in the following steps. Then, E will not be activated. Finally, the nodes $\{A, B, C, D\}$ are influenced by the seed set $\{A\}$.

2.4 Influence maximization problem on IDC model

The goal of the original influence maximization problem is to find a node subset S such that the expected nodes which are influenced by S is maximized. Obviously, to define the influence, a method measuring the benefit obtained by diffusing information over S should be given first. Based on the two classical models of information diffusion, the corresponding procedures of diffusing information are probabilistic. Therefore, the influence maximization problems are usually studied under the semantics of possible worlds.

The space of all possible worlds corresponding to a given IDC model can be determined by the following steps. Let Ω be the set of all different possible worlds of given IDC model. As shown in the section above, each possible world can be defined by a special procedure of information diffusion. That is, even if almost all steps in the information diffusion procedure of IDC model are probabilistic, once fixed series of steps are given, all operations can be determined. Therefore, each unique diffusion procedure is related with a possible world. Then, the probability of the possible world $X \in \Omega$ can be represented by $\mathbf{Pr}(X)$.

To be simple, each possible world can be represented by a deterministic-induced subgraph of the whole network. According to the information diffusion procedure of IDC model, all probabilistic choices are indeed the operations of tossing coins based on p_1 and $\{p^1, p^2, \dots\}$ for the contexts. Then, the probability of each possible induced graph $\hat{G}$ can be calculated by the following formula

$$\prod_{e \in S} \mathbf{Pr}(e) \prod_{e \in \overline{S}} (1 - \mathbf{Pr}(e)) \prod_{v \in T} U(v) \prod_{v \in \overline{T}} (1 - U(v)) \qquad (3)$$

where S is the set of edges selected in $\hat{G}$, and T is the set of nodes selected by the probability $p_1(\cdot)$ in $\hat{G}$. It can be verified that there exists efficient algorithms to check whether the induced graph $\hat{G}$ is feasible to obtain in IDC model. For each feasible induced graph, we can determine an information diffusion procedure to conduct it. It should be noted that in IDC model, two different processes may reach the same possible world. Therefore, the probabilities of each single process and possible world are different. Intuitively, we can define a standard information diffusion such that it can be one-to-one mapped to the induced graphs. As shown in the following part, such standard procedure can be simulated efficiently, and the problem can be solved by randomized estimation.

Based on the observations above, we can give the definition of influence function. Usually, we use the function $\delta(\cdot)$ to represent the influence range of given seed node set. That is, given seed set A, $\delta(A)$ will be the nodes which become active after diffusing the information based A. Observing the above procedure of information diffusion, for each single process, we have $\delta(A) = Z$. In fact, within the example shown above, each possible world G_i is indeed a deterministic single diffusion procedure, and the vertex set of G_i is just Z which can be influenced by A. However, the diffusion process is a probabilistic one, we need a definition based on possible world semantics.

Definition 1 (Influence function) *Given an IDC model $\langle G, C, P, U, \Theta \rangle$, an information context set C_I and the seed node set A, let $\{G_1, \dots, G_m\}$ be the set of all possible worlds. The influence function δ can measure the expected value of influence of A on G. For special A, $\delta(G, A, \theta)$ is defined to be $\sum \mathbf{Pr}(G_i) \cdot |V_{G_i}|$and it is also denoted by $\delta(A)$ for simplicity.*

Based on the definition of influence function, we can give the formal definition of influence maximization problem on IDC model.

Definition 2 (Influence maximization on IDC model) *Given an IDC model $\langle G, C, P, U, \Theta \rangle$, an information context set C_I and an integer $k > 0$, the question is to find a subset A satisfying $|A| \le k$ and the size $\delta(A)$ is maximized.*

In the following parts, we will use IM-IDC (influence maximization for information diffusion with contexts) to represent the influence maximization problem on IDC model.

3 Representing the contexts in diffusing information

In this section, we explain how to use those representations to describe the procedure of information diffusion.

3.1 Context tags for information and nodes

For each node v in the network, as shown in the section above, we use C_v to represent the context information of v. In a practical application, the context information can be collected by selecting the *tags* of interested topics. For example, most of social applications, such as Tweet, and

Sina Weibo, allow users to select interested tags which will be used to do the recommendation, indeed the tags can represent which kind of information will be read and retransmit. Second, the tags can also be obtained by analyzing the profiling information of users. For example, the user profiles in Sina Weibo also contains some important information of users such as living place and age, and the profiles also contain tag information which may affect the information diffusion, one of the key observations is that people usually are interested in things or topics related to their local events; by analyzing some more *customized* information, such as *self-descriptions* of users, utilizing techniques of natural language processing, we can also identify the tags which represent the factors affecting the procedure of information diffusion. Finally, the tags for nodes can also be *copied* from one node to another one. The structural information in social networks can be used to infer the tags of nodes. By finding clusters in the network, we can identify which nodes may contain similar tags. Intuitively, the nodes within one same cluster tend to contain similar tags; if the clustering algorithms take the information related to some special tags into consideration, the clustering result will provide a strong evidence for predicting the tags for nodes.

For information m in the network, such as one tweet, we use C_m to represent the contexts related to m. In real applications, we can obtained the contexts of m by following methods. First, most social platforms allow users to give the *highlights* or *labels* for the information produced in the network, for example, Sina Weibo allows users to use special symbols (e.g., #) to add topic labels for the message created by them. Second, a simple strategy is to use the users' tag for the information produced by them. Finally, the tags can also be generated by analyzing the information by sophisticated synthetic and semantic processing techniques for natural languages.

3.2 Determining the influence probabilities for contexts

According to the definition of contexts in the above section, during the procedure of information diffusion, the influence probability functions are different for different contexts. Essentially, general techniques for predicting influence probabilities in social network without context considerations can be utilized here to determine the influence probabilities. Intuitively, for each special context c, it can be done by using the predicting techniques over information diffusing records related to context c. As long as enough data for diffusing information can be collected, the influence probabilities can be predicted accurately. Since the goal of this paper is to design efficient algorithms for the problem of influence maximization, without loss of generality, we assume the influence functions of contexts are given in advance. In fact, even if only the influence functions for a special set of tags are given, when new tags appear, we can still build the influence probabilities for new tag incrementally by analyzing the procedure of information diffusion related to the corresponding tags.

4 Approximation algorithm for IM-IDC problem

In this section, first the computational complexity of IM-IDC problem is studied which indicates that it is intractable and is not expected to have exact algorithm in polynomial time. Then, an approximation algorithm for IM-IDC problem is designed.

4.1 Complexity of IM-IDC problem

Since the influence maximization problem on classic models are usually NP-*hard*, we can obtain the following result by simply restricting the definition of IM-IDC problem.

Theorem 1 *IM-IDC problem is NP-hard.*

Proof The theorem can be proved by observing that classical influence maximization problems on the IC model in [2] is a special case of IM-IDC problem. The detail can be analyzed as follows.

For IM-IDC problem, we can prove that they are NP-hard by making a direct reduction from the classical influence maximization problem on IC model in [2]. Given a classical influence maximization instance $I = \langle G, p, k\rangle$, we can build an instance of IM-IDC problem $I' = \langle G', P, C, U, \Theta, C_I, k'\rangle$ by the following steps.

- Let G' in the instance I' be same as the network graph G in I.
- Let $C = \{c\}$.
- For each node $u \in V_{G'}$, let $U(u) = \{c\}$, $\theta_u = 0$.
- Since there is only one context in C, let $P' = \{p'\}$, where for each edge (u, v) let $p'_{uv} = p_{uv}$.
- Let $C_I = \{c\}$, $k' = k$.

Obviously, the construction of I' can be finished in polynomial time, and it is easy to verify that we can find a subset A such that $|A| = k$ and $\delta(A) > x$ if and only if A is also a solution of IM-IDC satisfying $|A| = k'$ and $\delta(A) > x$. That is there are bijective maps between the solutions of I and I'. Therefore, IM-IDC problem is NP-hard. □

The result above indicates that it is impossible to design algorithms in polynomial time unless P=NP. Therefore, we need heuristic algorithms later. In fact, the problem is much harder than the analysis in Theorem 1. As shown in [1], the problem of computing $\delta(\cdot)$ under classic IC model has already been $\sharp P$-hard. That is, to compute an influence measure in a feasible way (e.g., on possible worlds semantics) has been already very hard. Of course, this result is meaningful only when we assume that all algorithms solving the IM-IDC problem will invoke a procedure to solve the subproblem of computing influence measures. Observing that no existing algorithms can avoid computing or estimating the influence measure directly, the result of $\sharp P$-hard indicates that the IM-IDC problem is much harder than we think in usual.

4.2 Efficient approximation algorithm

According to Theorem 1, since there are no efficient deterministic algorithms for the IM-IDC problem, therefore, in this part, an approximation algorithm is given. The main idea of this part is to design greedy algorithm with performance guarantees. A popular method is to utilize the monotonicity and submodularity properties of measuring functions. Given a function $\delta(\cdot) : 2^V \rightarrow R$, δ is called to be *monotone* if and only if $\delta(S_1) \leq \delta(S_2)$ for any $S_1 \subseteq S_2$, it is called to be *submodular* if $\delta(S_1 \cup x) - \delta(S_1) \geq \delta(S_2 \cup x) - \delta(S_2)$ for any $S_1 \subseteq S_2$. Informally speaking, suppose we are trying to find an optimal subset $S' \subseteq S$, the optimization goal is measured by a function f, a greedy algorithm has performance guarantee if f is *monotone* and *submodular*. In practical, for influence maximization problem, as shown in [2], monotone and submodular properties allow us to develop greedy algorithms to achieve $(1 - 1/e - \epsilon)$ approximation ratio.

We proposed an algorithm based on greedy idea which can produce approximation algorithms with ratio $1 - 1/e$ as shown by [10]. The algorithm is shown as Algorithm 1. The algorithm ApproIM-IDC takes $M = \langle G = (V, E), P, C, U, \Theta \rangle$, an information context set C_I, and integer $k > 0$ as the input parameters. First, two variables S and *cur* are initialized to be an empty set and 0, respectively, where S is a set for storing the optimal seed nodes, and *cur* is used to record the influence obtained by the algorithm during the whole procedure (line 2–3). Then, algorithm ApproIM-IDC iterates over the integer k. At each time, ApproIM-IDC selects one node u greedily (line 4–16). Intuitively, ApproIM-IDC tries to maximize the benefit obtained locally by selecting u, which may lose the chance to get the global optimal solution. As shown by the analysis in the following parts, the solution of ApproIM-IDC has approximation performance guarantees. The variable Δ_v is used to represent the influence benefit obtained by adding v to S, that is $\delta(S \cup v) - \delta(S)$. The function CalInf is invoked to calculate the influence (line 8), which will be explained in the following part. Finally, the set S will be returned by ApproIM-IDC as the approximation of optimal seed node set (line 17).

Algorithm 1 ApproIM-IDC

Input: $M = \langle G, P, C, U, \Theta \rangle$, the information context set C_I and a positive integer k
Output: The seed node set S

```
 1: function ApproIM-IDC($M, C_I, k$)
 2:     $S \leftarrow \emptyset$
 3:     $cur \leftarrow 0$
 4:     for $i \leftarrow 1$to$k$ do
 5:         $dinf \leftarrow 0$
 6:         $u \leftarrow$ null
 7:         for $v \in V \setminus S$ do
 8:             $\Delta_v \leftarrow$ CalInf($S \cup v, M, C_I$)$-cur$
 9:             if $\Delta_v > dinf$ then
10:                 $dinf \leftarrow \Delta_v$
11:                 $u \leftarrow v$
12:             end if
13:         end for
14:         $S \leftarrow S \cup u$
15:         $cur \leftarrow cur + dinf$
16:     end for
17:     return $S$
18: end function

19: function TossCoin($p$)
20:     $coin \leftarrow 0$
21:     $r \leftarrow$ getRandom(0,1)
22:     if $r \leq p$ then
23:         $coin \leftarrow 1$
24:     end if
25:     return $coin$
26: end function

27: function CalInf($S, M, C_I$)
28:     $influence \leftarrow 0$
29:     for $i \leftarrow 1$to$n$ do
30:         Initialize $T$, $L$ and $R$ to be empty sets
31:         $influence \leftarrow influence \times \left(1 - \frac{1}{i}\right)$
32:         for each node $v \in V \setminus S$ do
33:             $v.state \leftarrow$ inactive
34:             for each context $c \in C_I$ do
35:                 $v.con[c] \leftarrow$ false
36:             end for
37:         end for
38:         for each node $v \in S$ do
39:             $v.state \leftarrow$ active
40:             $T \leftarrow T \cup v$
41:             $R \leftarrow R \cup v$
42:             $L \leftarrow L \cup v$
43:         end for
44:         while $L \neq \emptyset$ do
45:             $u \leftarrow$ one node $u \in L$
46:             $N_u \leftarrow$ neighbors of $u$ in $G$
47:             for each node $v \in N_u$ do
48:                 if $v \in T$ then
49:                     continue
50:                 end if
51:                 if toActive($u, v, M, C_I$) then
52:                     $T \leftarrow T \cup \{v\}$
53:                     if toProgressive($v, M, C_I, R$) then
54:                         $L \leftarrow L \cup \{v\}$
55:                         $R \leftarrow R \cup \{v\}$
56:                     end if
57:                 end if
58:             end for
59:             $L \leftarrow L \setminus u$
60:         end while
61:         $influence \leftarrow influence + \frac{|T|}{i}$
62:     end for
63:     return $influence$
64: end function
```

The function CalInf is shown in Algorithm 1 also (line 27–64). In the CalInf procedure, the inputs include the information context set C_I, the seed node set S, and the instance M of IM-IDC model, the task is to compute the expected influence obtained by diffusing information from S in M under possible world semantics. As discussed before, the problem of computing $\delta(\cdot)$ is at least $\sharp P$-hard; therefore, we give a randomized algorithm to estimate the value of $\delta(\cdot)$. It is easy to verify that the procedure CalInf can give an estimation of $\delta(\cdot)$ satisfying the requirements by allowing multiple runs of the randomized algorithm. Thus, our idea is to compute the influence by simulating the information diffusion procedures enough times. First, the variable *influence* for storing the final result is initialized to be zero (line 28). Then, the randomized estimation method will be ran for n times (line 29–62) (the value of n can be determined according to the method in [2]) and the averaged value of all result influences will be returned (line 63). During each iteration, we use three variables T, L, and R to represent the set which is composed of the active nodes, the progressive nodes, and progressive nodes whose influences have not been calculated, respectively (line 30). Then, S is used to initialize all temporary variables used (line 32–43). Nodes in V are divided into two parts, S and $V \setminus S$, and for each node v the variable $v.state$ is used to record the node v is inactive, active, or progressive, and the variable $v.con[c]$ is used to indicate whether v has been influenced by the information over a specified context c. Then, the inactive nodes will be processed one by one, and the node may be influenced during this procedure (line 44–60). The function ToActive is invoked to determine whether node v will be influenced by u (line 51). Because of the privacy considerations, even if v becomes active, it may stop reposting the information, therefore, the function ToProgressive is invoked to determine whether v will be *progressive* (line 54). During the iterations, L is used to maintain the nodes which should be considered in the computation of influence. New nodes becoming progressive in the last iteration will be added into L (line 55), and once one node has been considered, it will be removed from L(line 59). The two functions ToProgressive and ToActive are shown in Algorithm 2, which are used to change the state of some node.

4.3 Analysis of algorithm ApproIM-IDC

In this part, we will show that algorithm ApproIM-IDC has performance guarantee on time complexity and the approximation ratio. The main idea is to show the influence function δ satisfies the properties of monotone and submodular.

First, based on the observation that the main procedure of algorithm ApproIM-IDC is to iterate among all nodes, and the CalInf procedure only enumerates every edge of G, it is easy to verify that algorithm CalInf can be finished in polynomial time. In the function ToProgressive, the time costs can be bounded by $O(|C| + |V|)$, where C represents the context set usually with constant size and V is caused by the iteration of $N(v)$. The time cost of function ToActive can be bounded by $|C_I| < |C|$. In CalInf, the main time cost comes from the procedure of processing each progressive node (line 44–60), it is not hard to verify that the cost can be bounded by $O\left(|E|^2\right) \cdot |C|$; therefore, the time cost of CalInf can be bounded by $O\left(n \cdot |E|^2 \cdot |C| + n \cdot |V|\right)$. Combining CalInf with ApproIM-IDC and ignoring the value $|C|$ by treating it to be a constant, the total time cost of ApproIM-IDC can be bounded by $O(k \cdot n \cdot |V| \cdot |E|^2)$.

Algorithm 2 ApproIM-IDC(con.)

```
Input: $M = \langle G, P, C, U, \Theta \rangle$, the information context set
       $C_I$ and a positive integer $k$
Output: The seed node set $S$

function ToProgressive($v, M, C_I, R$)
    $sizeCu \leftarrow |U(v)|$
    $p_1 \leftarrow 0$
    for each context $c \in U(v)$ do
        if $c \in C_I$ then
            $p_1 += \frac{1}{sizeCu}$
        end if
    end for
    $N_v \leftarrow$ neighbor nodes of $v$ in $G$
    $\hat{N}_v \leftarrow \emptyset$
    for each node $u \in N(v)$ do
        if $u \in R$ then
            $\hat{N}_v = \hat{N}_v \cup \{u\}$
        end if
    end for
    $f_1 \leftarrow \frac{|\hat{N}_v|+1}{|N_v|+1}$
    if TossCoin($p_1$) then
        if $f_1 \geq \theta_v$ then
            return true
        end if
    end if
    return false
end function

function ToActive($u, v, M, C_I$)
    $flag \leftarrow$ true
    for each context $c \in C_I$ do
        if TossCoin$\left(p^c_{uv}\right)$ then
            $v.con[c] \leftarrow$ true
        end if
        $flag \leftarrow flag \wedge v.con[c]$
    end for
    return $flag$
end function
```

Then, we will give an equivalent form of δ, represented by $\hat{\delta}$, which is based on the general threshold model defined by [11]. In that model, the influence functions are defined over node sets but not over nodes. In detail, for a special node u, an influence function f_v is defined by mapping the elements in $2^{\{V\}}$ to the range $[0, 1]$, a necessary condition is that $f_v(\emptyset) = 0$. A threshold γ_v is given to control whether v will be activated, where v will become *active* when we have $f_v(S) \geq \gamma_v$ for which S is the neighbors of v

active. It is not hard to see that this model can be extended to the setting of context considerations. We can replace f_v with a set of influence functions $\{f_v^{\varphi_c}\}$ where φ_c is a corresponding dimension to a special context in C, and let the condition that v becomes active be v is influenced by all contexts, that is we have $f_v^{\varphi_c}(S) \geq \gamma_v^{\varphi_c}$ for all contexts.

For the convenience in following this part, we will introduce several useful notations. For an arbitrary active node set $\hat{S}$, where $\hat{S}$ may be different from the input parameter S of the problem since $\hat{S}$ will represent the active set in the network during the whole procedure of information diffusion, let $H_u(S)$ be the probability that u satisfies the constraints defined by $p_1(u, I)$. Obviously, the condition of p_1 depends on the context sets of information and users, it is independent from which users have become active. To be consistent, we still use the parameter S for H_u. For a special information I, we have $H_u(S) = |C_u \cap C_I|/|C_u| = \lambda_u$.

For an arbitrary active node set $\hat{S}$ and a node u which has passed the verification of p_1 condition, let $H'_u(S)$ be the probability that u satisfies the constraints defined by $f_1(u)$. Actually, the probability can be represented by an indicator function $I_{f_1(u,S)\geq\theta_u}(u)$, which is defined to be 1 if $f_1(u, S) \geq \theta_u$ or 0 if $f_1(u, S) < \theta_u$. Here, $f_1(u)$ can be rewritten with respect to S as $f_1(u, S) = \frac{|N_u \cap S|+1}{|N_u|+1}$, where the value $f_1(u, S)$ only depends on the network structure and the active set S.

Here, the definition of multi-dimensional threshold model is introduced first, which is used in [11] to give general definitions of influence maximization problem for extending linear threshold and independent cascade models. Here, for a special dimension φ, each node is associated with a monotone threshold function f_v^{φ} which maps subsets of neighbors of v to a probability value in $[0, 1]$, satisfying $f_v^{\varphi}(\emptyset) = 0$. Initially, each node v selects a random threshold value θ_v^{φ} uniformly. At each step k, a node v becomes active if and only if for all dimensions the node v has $f_v^{\varphi}(S) \geq \theta_v^{\varphi}$ where S is the set of neighbors of v active in the step $k - 1$.

Theorem 2 *For any instance $I = \langle G, C, P, U, \Theta\rangle$ of IDC model and an information context set C_I, there exists an equivalent model $I' = \langle G, F, \Phi\rangle$ of multi-dimensional threshold model. That is, for any seed set A, we have $\delta_I(A) = \delta_{I'}(A)$.*

Proof The proof will consist of two parts, constructing I' for given I and showing $\delta_I(A) = \delta_{I'}(A)$.

First, we will introduce the method of construction I' by assuming that for each node u we have $\theta_u > \frac{1}{1+|N_u|}$ which will be eliminated later. Given an instance $I = \langle G, C, P, U, \Theta\rangle$ of IDC model and an information context set C_I, we can build the corresponding multi-dimensional threshold model $I' = \langle G, F, \Phi\rangle$ as follows:

- First, let the network G used in I' be the same one in I.
- Second, for each context $c \in C$, construct a corresponding dimension $\varphi_c \in \Phi$, and let the set $\{\varphi_c | c \in C\}$ be Φ_C. Additionally, build two dimensions φ_1 and φ_2 in Φ. For each node $v \in G$ and each dimension $\varphi \in \Phi$, there will be a definition of function f_v^{φ}.
- Then, for each context $c \in C_I$ and its active neighbor set S, let $f_v^{\varphi_c}(S) = 1 - \prod_{u\in S}\left(1 - p_{uv}^c\right)$, and for context $c \notin C_I$ let $f_v^{\varphi_c}(S) = 1$. If $S = \emptyset$, let $f_v^{\varphi_c}(S) = 0$.
- For the dimension φ_1, let $f_v^{\varphi_1}(S) = H_v(S) = \lambda_v$.
- For the dimension φ_2, its construction is a little complex. Essentially, we will build a dimension set for nodes in G, that is there exist a corresponding dimension φ_v for each node $v \in G$. The final dimension φ_2 is indeed the composition of all φ_v. Let $f_v^{\varphi_v}(S) = f_1(v, S)$ and $f_v^{\varphi_u}(S) = 1$ for all $u \neq v$, and let $f_v^{\varphi_2}(S) = f_v^{\varphi_v}(S)$. During the information diffusion procedure, $f_v^{\varphi_2}(S)$ will be compared with the random selected value θ_{φ_v}.

Next, we will show that the two models are equivalent, that is given any seed set A, we have $\delta_I(A) = \delta_{I'}(A)$. The proof is by induction. The whole procedure of information diffusion is divided step by step, where the initial seed set is step$_0$ and the following steps can be represented by step$_1$, step$_2$, and so on. Let A_i be the active node set in step$_i$. Obviously, $A_0 = A$. Intuitively, since the influence function δ is defined by the exception value of influenced range, it is sufficient to show that the probability $\mathbf{Pr}[v \text{ is active}]$ that each node v will become active during the information diffusion procedure. Moreover, because the diffusion steps are relatively independent according to the definition of IDC model, it will be sufficient to show that the values of $\mathbf{Pr}[v \in A_k]$ are same for the two models.

(1) For the basic, consider A_0, it is obvious that $\mathbf{Pr}_I[v \in A_0] = \mathbf{Pr}_{I'}[v \in A_0]$.

(2) Inductively, assume that for all $i \leq k$, we have $\mathbf{Pr}_I[v \in A_i] = \mathbf{Pr}_{I'}[v \in A_i]$. Then, we will try to show that for some node $v \notin A_k$, the probabilities that v become active in the next step for the two models are the same. According to the definition of IDC model, we have

$$\begin{aligned} &\mathbf{Pr}_I[v \in A_{k+1}] \\ =&H_v(A_k)\cdot \mathbf{Pr}[H'_v(A_k)=1]\cdot \prod_{c\in C_I} H_v^c(A_k)\\ =&H_v(A_k)\cdot \mathbf{Pr}[f_1(v,A_k)\geq\theta_v]\cdot \prod_{c\in C_I} H_v^c(A_k)\end{aligned} \tag{4}$$

Here, $H_v^c(A_k)$ is the probability that v is influenced by some factor c and some node in A_k. Since the influence effects of all nodes in A_k are independent, we have

$$H_v^c(A_k) = 1 - \prod_{u \in A_k \cap N_v} \left(1 - p_{uv}^c\right) \tag{5}$$

According to the definition of I', since for each dimension φ the threshold value θ_φ is randomly selected within 0 and 1 in uniform probability, we have

$$\begin{aligned}
&\mathbf{Pr}_{I'}[\, v \in A_{k+1}] \\
&= \prod_{\varphi \in \Phi} \mathbf{Pr}\left(f_v^{\varphi}(A_k) \geq \theta_\varphi\right) \\
&= f_v^{\varphi_1}(A_k) \cdot \mathbf{Pr}\left[f_v^{\varphi_2}(A_k) \geq \theta_{\varphi_v}\right] \prod_{\varphi_c \in \Phi_C} f_v^{\varphi_c}(A_k)
\end{aligned} \tag{6}$$

Then, according to the construction of I', we have $f_v^{\varphi_1}(A_k) = H_v(A_k)$ and

$$\begin{aligned}
&\mathbf{Pr}\left[f_v^{\varphi_2}(A_k) \geq \theta_{\varphi_v}\right] \\
&= \prod_{u \in G} \mathbf{Pr}\left[f_v^{\varphi_u}(A_k) \geq \theta_{\varphi_u}\right] \\
&= \prod_{u \in G \wedge u \neq v} \mathbf{Pr}\left[f_v^{\varphi_u}(A_k) \geq \theta_{\varphi_u}\right] \cdot \\
&\quad \mathbf{Pr}\left[f_v^{\varphi_v}(A_k) \geq \theta_{\varphi_v}\right] \\
&= \mathbf{Pr}\left[f_v^{\varphi_v}(A_k) \geq \theta_{\varphi_v}\right] \\
&= \mathbf{Pr}[f_1(v, A_k) \geq \theta_{\varphi_v}] \\
&= f_1(v, A_k)
\end{aligned} \tag{7}$$

Because all θ_v in I and all θ_{φ_v} in I' are randomly selected in uniform way, we have

$$\begin{aligned}
\mathbf{Pr}\left[f_v^{\varphi_2}(A_k) \geq \theta_{\varphi_v}\right] &= f_1(v, A_k) \\
&= \mathbf{Pr}[f_1(v, A_k) \geq \theta_v]
\end{aligned} \tag{8}$$

Moreover, we have

$$\begin{aligned}
&\prod_{\varphi_c \in \Phi_C} f_v^{\varphi_c}(A_k) \\
&= \prod_{\varphi_c \in \Phi_C \wedge c \notin C_I} f_v^{\varphi_c}(A_k) \\
&\quad \cdot \prod_{\varphi_c \in \Phi_C \wedge c \in C_I} f_v^{\varphi_c}(A_k) \\
&= \prod_{\varphi_c \in \Phi_C \wedge c \in C_I} f_v^{\varphi_c}(A_k) \\
&= \prod_{c \in C_I} f_v^{\varphi_c}(A_k) \\
&= \prod_{c \in C_I} \Big(1 - \prod_{u \in A_k \cap N_u} \left(1 - p_{uv}^c\right)\Big) \\
&= \prod_{c \in C_I} H_v^c(A_k).
\end{aligned} \tag{9}$$

Therefore, we have $\mathbf{Pr}_I[\, v \in A_{k+1}] = \mathbf{Pr}_{I'}[\, v \in A_{k+1}]$.

Totally, we will have $\delta_I(A) = \delta_{I'}(A)$. □

Actually, the proof of Theorem 2 still needs the following three theorems to be complete, since we made an assumption that $\theta_u > \frac{1}{|N_u|+1}$ and did not verify the wellness of definition of I'.

Theorem 3 *For all threshold function f_v^{φ} built for I', they are monotone.*

Proof Let $S \subseteq S'$ be two subsets of V_G, it can be shown by verifying the functions. For the dimension $\varphi_c \in \Phi_C$, we have

$$\begin{aligned}
f_v^{\varphi_c}(S') &= 1 - \prod_{u \in S'} \left(1 - p_{uv}^c\right) \\
&= 1 - \prod_{u \in S} \left(1 - p_{uv}^c\right) \cdot \prod_{u \in S' \setminus S} \left(1 - p_{uv}^c\right) \\
&\geq 1 - \prod_{u \in S} \left(1 - p_{uv}^c\right) \cdot \prod_{u \in S' \setminus S} 1 \\
&= f_v^{\varphi_c}(S).
\end{aligned} \tag{10}$$

For the dimension φ_1, we have

$$f_v^{\varphi_1}(S') = H_v(S') = \frac{|C_v \cap C_I|}{|C_v|} = f_v^{\varphi_1}(S).$$

For the dimension φ_2, we have

$$f_1(v, S') = \frac{|N_v \cap S'| + 1}{|N_v| + 1} \geq \frac{|N_v \cap S| + 1}{|N_v| + 1} = f_1(v, S)$$

and

$$f_v^{\varphi_2}(S') = f_1(v, S') \geq f_1(v, S) = f_v^{\varphi_2}(S).$$

Therefore, threshold functions in I' are monotone. □

Theorem 4 *For all threshold function f_v^{φ} built for I', they can be extended to a special case satisfying that $f_v^{\varphi}(\emptyset) = 0$.*

Proof It can be finished by $f_v^{\varphi_1}(\emptyset) = 0$. Then, it is easy to verify that for all threshold functions, we have $f_v^{\varphi}(\emptyset) = 0$. The correctness can be verified easily also. Consider the proof shown in Theorem 2, if $S = \emptyset$, it means that $A = A_0 = S = \emptyset$. Trivially, we will have $\delta_I(A) = \delta_{I'}(A) = \emptyset$. □

Theorem 5 *Without the assumption that $\theta_u > \frac{1}{|N_u|+1}$, we still have Theorem 2.*

Proof Obviously, if the assumption that $\theta_u > \frac{1}{|N_u|+1}$ is false, we will have the condition defined by f_1 is trivially satisfied by any non-empty active neighbor set S. Therefore, in that case, we can let $f_v^{\varphi_2}(S) = 1$ for all $S \neq \emptyset$ and $f_v^{\varphi_2}(\emptyset) = 0$. Since the diffusing procedure is only meaningful for non-empty neighbor set, the definition will not affect the correctness of Theorem 2. □

Lemma 1 [11] *For the multi-dimensional threshold model with monotone non-decreasing submodular threshold functions, the expected influence range is a monotone non-decreasing submodular function with respect to the seed set.*

Theorem 6 *Algorithm* Approim-IDC *can solve the IM-IDC problem with* $(1 - 1/e)$ *approximation ratio.*

Proof According to the result in [2], a possible way to prove the theorem above is to show that δ is *monotone* and *submodular*. Then, the greedy-based approximation algorithm will induce a $(1 - 1/e)$ approximation ratio.

To show that the influence measure function δ is *monotone*, it can be obtained by analyzing the following factors in the information diffusion procedure.

- The first selection is sourced from the privacy consideration about that users only repost those information related to their public information. Intuitively, all users are relatively independent in this selection; if more users are added into the seed set, there will be more users in the next selection step. Therefore, this factor will not decrease the influence.
- The second selection is about f_1, which is sourced from the privacy consideration that the users tend to hidden themselves in other neighbor nodes. Intuitively, if more seed nodes are added, for each special node v, when calculating f_1, it is only possible that more neighbors become *progressive*, therefore, v will be more likely to be *progressive* in the following steps. Thus, this factor will not decrease the influence neither.
- The last selection in the IM-IDC problem is the *activate* operations. Intuitively, if more seeds are added, for each special node v, the only possible change is that more nodes will be used to activate v, which will also increase the probability that v become active. Therefore, in the view of this factor, the influence measure function is monotone also.

To show the submodularity, we can use the result in [11] and give a representation of δ in the general threshold model. According to Theorem 2, for a given instance $I = \langle G, C, P, U, \Theta \rangle$ of IDC model and an information context set C_I, there exists an equivalent model $I' = \langle G, F, \Phi \rangle$ of multi-dimensional threshold model, such that we have $\delta_I(A) = \delta_{I'}(A)$ for any seed set A. Since our goal is to prove $\delta = \delta_I$ is submodular with respect to A, it is sufficient to show that $\delta_{I'}$ is submodular. According to Lemma 1, if we can show for the multi-dimensional threshold model I' all threshold functions are monotone non-decreasing submodular, the expected value $\delta_{I'}$ will be a monotone non-decreasing submodular function with respect to the seed set. Therefore, it is sufficient to show that all f_v^{φ} for $\varphi \in \Phi$ are monotone non-decreasing and submodular.

Obviously, because of Theorem 3, we only need to show all f_v^{φ} functions are submodular. Recall that, assuming a function f is defined on all subsets of a set S, f is submodular if and only if $f(A) + f(B) \geq f(A \cup B) + f(A \cap B)$ for any $A, B \subseteq S$.

- First, $f_v^{\varphi_1}$ is submodular. If neither of A and B is empty, we have $f_v^{\varphi_1}(A) + f_v^{\varphi_1}(B) = 2\lambda_v = f_v^{\varphi_1}(A \cup B) + f_v^{\varphi_1}(A \cap B)$. If one of them is empty, we have $f_v^{\varphi_1}(A) + f_v^{\varphi_1}(B) = \lambda_v = f_v^{\varphi_1}(A \cup B) + f_v^{\varphi_1}(A \cap B)$. If $A = B = \emptyset$, we have $f_v^{\varphi_1}(A) + f_v^{\varphi_1}(B) = 0 = f_v^{\varphi_1}(A \cup B) + f_v^{\varphi_1}(A \cap B)$.
- Second, $f_v^{\varphi_u}$ is submodular. If $u \neq v$, we have $f_v^{\varphi_u}(A) + f_v^{\varphi_u}(B) = 2 = f_v^{\varphi_u}(A \cup B) + f_v^{\varphi_u}(A \cap B)$; otherwise, we have $f_v^{\varphi_v}(A) + f_v^{\varphi_v}(B) = f_1(v, A) + f_1(v, B) = \frac{2+|N_v \cap A| + |N_v \cap B|}{|N_v|+1}$ and $f_v^{\varphi_v}(A \cup B) + f_v^{\varphi_v}(A \cap B) = \frac{2+|N_v \cap (A \cap B)| + |N_v \cap (A \cup B)|}{|N_v|+1}$. Since for arbitrary set S, we have $|N_v \cap S| = \sum_{u \in N_v} |u \cap S|$, consider the nodes in N_v by three disjoint parts, we can easily show the submodularity. Let u be a node in N_v. If $u \notin A \cup B$, we have $|u \cap A| = |u \cap B| = |u \cap (A \cap B)| = |u \cap (A \cup B)| = |\emptyset| = 0$; if $u \in A \cap B$, we have $|u \cap A| = |u \cap B| = |u \cap (A \cap B)| = |u \cap (A \cup B)| = |\{u\}| = 1$; if $u \in A \cup B$ and $u \notin A \cap B$, we have $|u \cap A| + |u \cap B| = |u \cap (A \cap B)| + |u \cap (A \cup B)| = |\{u\}| = 1$. Totally, we have $f_v^{\varphi_u}(A) + f_v^{\varphi_u}(B) \geq f_v^{\varphi_u}(A \cup B) + f_v^{\varphi_u}(A \cap B)$.
- Finally, $f_v^{\varphi_c}$ is submodular. According to the definition of $f_v^{\varphi_c}$, we have

  $$f_v^{\varphi_c}(A) + f_v^{\varphi_c}(B)$$

  $$= 2 - \prod_{u \in A} \left(1 - p_{uv}^{c}\right) - \prod_{u \in B} \left(1 - p_{uv}^{c}\right)$$

  and

  $$f_v^{\varphi_c}(A \cap B) + f_v^{\varphi_c}(A \cup B)$$

  $$= 2 - \prod_{u \in A \cap B} \left(1 - p_{uv}^{c}\right) - \prod_{u \in A \cup B} \left(1 - p_{uv}^{c}\right).$$

  Since

  $$\left(1 - \prod_{u \in A \backslash B} \left(1 - p_{uv}^{c}\right)\right) \left(1 - \prod_{u \in B \backslash A} \left(1 - p_{uv}^{c}\right)\right) \geq 0,$$

  we have

  $$1 + \prod_{u \in (A \backslash B) \cup (B \backslash A)} \left(1 - p_{uv}^{c}\right) \geq \prod_{u \in A \backslash B} \left(1 - p_{uv}^{c}\right) + \prod_{u \in B \backslash A} \left(1 - p_{uv}^{c}\right). \tag{11}$$

Then, we have

$$\begin{aligned} &\prod_{u\in A\cap B}\left(1-p^{c}_{uv}\right)+\prod_{u\in A\cup B}\left(1-p^{c}_{uv}\right) \\ \geq &\prod_{u\in A}\left(1-p^{c}_{uv}\right)+\prod_{u\in B}\left(1-p^{c}_{uv}\right). \end{aligned} \tag{12}$$

In total, we have
$f^{\varphi_c}_v(A)+f^{\varphi_c}_v(B)\geq f^{\varphi_c}_v(A\cap B)+f^{\varphi_c}_v(A\cup B)$.

Finally, since the influence measure function δ is monotone and submodular, the greedy algorithm APPROIM-IDC has a $(1-1/e)$ approximation ratio. Here, it should be noted that the approximation ratio result obtained by submodularity property is independent from the classical influence maximization problem, although we change the way of defining threshold functions in IDC model, we can still obtain the ratio $1-1/e$.

□

5 Extensions and optimizations

5.1 Extension for general influence maximization

The results obtained in previous part are based on careful analysis about the process of diffusing special information denoted by C_I over IDC model. Theorem 2 only works for the case that a special information has been given and the context set C_I can be obtained in advance. Essentially, given a network G and settings for IDC model, the APPROIM-IDC algorithm shown in Fig. 1 can only output a seed set A which can maximize the expected influence of given information with context set C_I. A more general case is that the context set of some information is randomly selected within a given domain, and it is expected that the APPROIM-IDC algorithm can be simply extended to solve the related influence maximization problem.

First, assume that the information context set C_I is taken from the global context set C, where each context in C is selected by probability α.

Theorem 7 *For any instance $I=\langle G,C,P,U,\Theta\rangle$ of IDC model, there exists an equivalent model $I'=\langle G,F,\Phi\rangle$ of multi-dimensional threshold model. That is, for any seed set A, suppose the information context set C_I is taken from the global context set C satisfying that each context is selected uniformly in probability α, we have $\hat{\delta}_I(A)=\mathbf{Exp}_{C_I}\delta_I(A)=\delta_{I'}(A)$.*

Proof According to the proof of Theorem 2, we know that the influence function $\delta_I(A)$ can be explained as follows. The whole procedure of information diffusion is divided step by step, where the initial seed set is step_0 and the following steps can be represented by step_1, step_2, and so on. The construct of I' only needs a small fix, where for the dimension φ_1, let $f^{\varphi_1}_v(S)=\alpha$. Obviously, the instance I' is still well defined, and we only need to show the equivalence. It is sufficient to show that the probability $\mathbf{Pr}[v \text{ is active}]$ that each node v will become active during the information diffusion procedure. Moreover, for each random selected information, because the diffusion steps are still independent according to the definition of IDC model, it will be sufficient to show that the values of $\mathbf{Pr}[v\in A_k]$ are the same for the two models. Considering the formula shown in Theorem 2, we have

$$\begin{aligned} &\mathbf{Pr}_I[\,v\in A_{k+1}] \\ =&H_v(A_k)\cdot\mathbf{Pr}[f_1(v,A_k)\geq\theta_v]\cdot\prod_{c\in C_I}H^c_v(A_k). \end{aligned} \tag{13}$$

Obviously, for each special information C_I and each node v, the value $H_v(A_k)$ is independent from A_k. Let us consider another instance I'' which is a variant of I such that the verification of p_1 is ignored, let $\delta_{I''}(A)$ be the expected influence value of A. It can be found that $\delta_{I''}(A)$ is independent from C_I since I'' does not consider the constraint of p_1. Then, we have $\hat{\delta}_I(A)=\mathbf{Exp}_{C_I}\delta_I(A)=\mathbf{Exp}H_v(S)\cdot\delta_{I''}(A)$. According to the definition of p_1, we have $\mathbf{Exp}H_v(S)=\mathbf{Exp}|C_v\cap C_I|/|C_v|=\alpha$. Thus, we have $\hat{\delta}_I(A)=\alpha\cdot\delta_{I''}(A)$. According to the definition of I' and I'', it is easy to check that $\delta_{I'}(A)=\alpha\cdot\delta_{I''}(A)=\hat{\delta}_I(A)$. □

It is not hard to check that $\hat{\delta}_I(A)$ also satisfies the submodular property, then, by replacing the procedure of verifying p_1 with checking the value of random variable generated by probability α, the APPROIM-IDC algorithm can be extended to solve the influence maximization problem for general information in IDC model, which also guarantees that the approximation ratio is $1-1/e$.

For more, assume that the random information has only context set with fixed size b. Utilizing similar techniques, we can construct the corresponding instance I' by letting $f^{\varphi_1}_v(S)=\frac{b}{|C|}$. Since $\mathbf{E}(H_u(S))=\frac{C^{b-1}_{|C|-1}}{C^{b}_{|C|}}=\frac{b}{|C|}$, we can obtain similar result.

Theorem 8 *For any instance $I=\langle G,C,P,U,\Theta\rangle$ of IDC model, suppose the information context set C_I is taken from the global context set C satisfying that each context is selected uniformly and the context size is b, there exists an equivalent model $I'=\langle G,F,\Phi\rangle$ of multi-dimensional threshold model, satisfying that $\mathbf{Exp}_{C_I}\delta_I(A)=\delta_{I'}(A)$.*

Still, the corresponding influence maximization problem can be solved approximately with ratio $1-1/e$.

5.2 Optimizations by preprocessing

Compared with classical information diffusion model, the IDC model is more complex, which is mainly caused by the verification of information relativeness and privacy. It appears to have two drawbacks, the first one is we need more computation cost to simulate the procedure of diffusing information in IDC model, the second one is that most of the previous work on optimizing the influence maximization algorithms do not work again since the distributions of active nodes become very different. In general case, it seems that nothing can be done to fix this, since IDC model is more expressive and the drawbacks are the side effects of expressibility. In this part, some special cases are considered, and optimization methods are introduced.

Suppose we know that many information with the same context set C_I is often used to generate the seed set with maximum expected influence. Given an instance $I = \langle G, C, P, U, \Theta \rangle$ of IDC model, we can simplify G and the procedure of diffusing information by preprocessing I to generate another equivalent instance I'. First, given C_I, obviously, we can eliminate the function set P to P_{C_I} which only includes the functions related to contexts appearing in C_I. Then, for each node v satisfying that $C_v \cap C_I = \emptyset$, the out edges of v can be removed from G. Intuitively, even if v becomes active, it will not try to trigger other nodes further, according to the definition of IDC model. For more, the seed set A can only be considered to be selected within $V_G \setminus v$, except for some trivial cases like $|V_G| \leq k$. When the values in Θ have been given, if some node v satisfies that $\frac{1}{|N_v|+1} \geq \theta_v$, the verification of the second constraint defined by f_1 can be removed.

Algorithm 3 REDUNCONTEXT

Input: The context set C, the similarity matrix D, $0 \leq \epsilon_1 < \epsilon_2 \leq 1$, and $0 < \alpha \leq 1$

Output: A partition P of context C

```
1:  function REDUNCONTEXT($C, D, \epsilon_1, \epsilon_2$)
2:      $P \leftarrow \{\{c\} | c \in C\}$
3:      for each value $D[\,i]\,[\,j]$ in $D$ do
4:          if $D[\,i]\,[\,j] < \epsilon_1$ then
5:              $D[\,i]\,[\,j] \leftarrow 0$
6:          end if
7:          if $D[\,i]\,[\,j] > \epsilon_2$ then
8:              Let $c_i(c_j)$ be the set in $P$ containing $i(j)$
9:              $P \leftarrow P \setminus \{c_i, c_j\}$
10:             $P \leftarrow P \cup c_i \cup c_j$
11:         end if
12:     end for
13:     while true do
14:         $flag \leftarrow$ false
15:         for each pair $p$ and $p'$ in $P$ do
16:             $factor \leftarrow \frac{\left(\sum_{c_i, c_j \in p \cup p'(D[i][j])}\right)^{\frac{1}{2}}}{|p|+|p'|}$
17:             if $factor \geq \alpha$ then
18:                 $P \leftarrow P \setminus \{p, p'\}$
19:                 $P \leftarrow P \cup p \cup p'$
20:                 $flag \leftarrow$ true
21:                 break;
22:             end if
23:         end for
24:         if !$flag$ then
25:             break;
26:         end if
27:     end while
28:     return $P$
29: end function
```

5.3 Optimization by removing redundant contexts

In the definition of IDC model, we did not discuss the problem of redundant contexts which are usually viewed in real applications. Obviously, the computation cost of simulating information diffusion procedure will increase much, even if only one extra context is added. Therefore, less contexts will produce efficient algorithms for influence maximization problem.

Redundant contexts need to be identified, which is helpful to improve the efficiency and fill the missing context information using similar ones. To be simple, it is assumed that there is a matrix which contains the information about similarity values between each two contexts. In practical applications, such a matrix can be obtained by learning methods such as embedding algorithms [12] or ranking algorithms such as simrank [13]. Intuitively, since the matrix contains similarity values of every pair of contexts, while the ideal structure we need is actually a clique-based matrix, efficient algorithms are needed to resolve this problem. As shown in algorithm 3, the main idea is a bottom-up method. Two parameters ϵ_1 and ϵ_2, are used to preprocess the similarity matrix to eliminate the noises as much as possible. Then, P is initialized to be the most refined partition of all contexts. For each two partitions in P, every time REDUNCONTEXT considers to merge the partitions to build a bigger partition by calculating a value which can be used to measure the closeness of the two partitions (line 15–23). If the two partitions are close enough, they will be merged into one. If there are no partitions, it can be merged further, REDUNCONTEXT stops, and outputs P as the final result. Intuitively, if two contexts are put into the same partition in P, they will be treated as redundant contexts latter.

5.4 Obtaining the influence probabilities by learning

In this part, one possible solution for obtaining the influence probabilities by learning methods is introduced, which is based on word2vec, a powerful method for learning compressed representations. Word2vec is a group of learning models that are used to produce word embeddings which are usually used to reconstruct linguistic contexts of words, whose detail can be found in [12]. The main idea of word2vec is to take the inputs of a large corpus of text and compute a vector space, where each unique word in the corpus is represented by a corresponding vector in the space. There are two models often utilized, CBOW (continuous Bag-of-Words) and Skip-gram (continuous Skip Gram). A common framework of CBOW is shown in Fig. 2, it is widely used. There are usually three layers: input layer, hidden layer, and output layer. Intuitively, the word2vec method can be used to construct condensed representations for a set of keys. The *co-occurrence* relationships of keys are considered, which means that those keys appearing at the same time when we try to search the keys within information we take. If we treat the keys as input of the word2vec method, the hidden layer will maintain a matrix which can be used to explain the relations between different keys. The matrix can help us to know which keys are *similar*, which keys have *co-occurrence* relations, and so on. Usually, the output will be a key which has highest possibility to appear with all given keys together.

Therefore, one possible solution for determining the influence probabilities for the contexts can be summarized as follows. For each information sequence, suppose that the same techniques are utilized to extract contexts from both users and messages and a bag-of-word model Q is built to learn the probabilities, the sequence will be transformed into a context set sequence. Here, each context set is obtained from the corresponding users in the information sequence. Then, by analyzing the message, another special context set for the message can be produced, and the context set sequence will be associated with each item in the message context set. After that, for each context c, the associated context set sequence will be sent to the model Q, where each context set can be reduced to one unique item, since we do not consider the similarity relation between contexts here. When estimating the influence probability between two nodes u and v on special context c, a certain number of prefix sequences can be generated and concatenated with u. Using the whole sequence as the input of Q, the appearing probability of v after this sequence can be obtained in the output layer. Finally, the influence probability can be estimated efficiently by running the model multiple times and taking the average probability values.

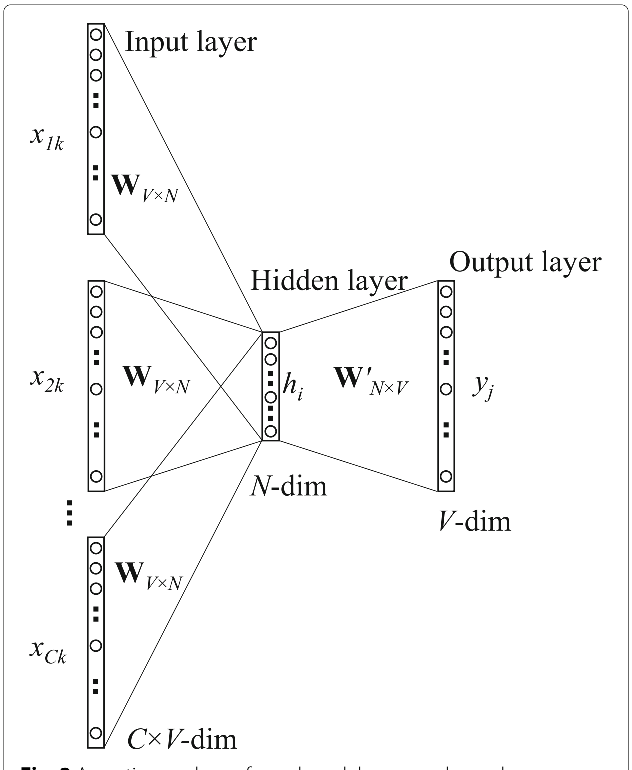

Fig. 2 A continuous bag-of-word model commonly used

6 Experiments

In this part, experiments on real datasets are conducted to evaluate the efficiency and performance of the approximation influence maximization algorithm on IM-IDC problem. The aims of our experiments are to illustrate the effects of different parameters of the algorithm on the influence obtained by the algorithm.

6.1 Experiment setting

We ran our experiments on two real datasets, DBLP and LiveJournal, which are collected from the SNAP project of Stanford University[1]. The DBLP dataset is a large network of research collaboration maintained by Michael Ley. In the network of DBLP, the nodes represent the authors of academic papers, and there exist one edge between two nodes if and only if the two corresponding authors have collaborations. For DBLP, we use the coauthor relationships to compute the influence probability between two authors. The LiveJournal dataset is a free online community with almost 10 million members, a significant fraction of these members are highly active. (For example, according to the statistics of the SNAP project, roughly 300,000 update their content in any given 24-h period.) The members use LiveJournal to maintain information about journals, individual, and group blogs. Also, LiveJournal allows people to declare which other members are their friends they belong, which provides us the social relation between users.

The algorithm APPROIM-IDC is implemented and executed on PCs with 3.40 GHz Intel Core i7 CPU and 32 GB of DDR3 RAM, running Ubuntu 16.04. All experiments about running times were ran five times, and the average values are reported.

6.2 Experimental results and discussions

In the experiments, the effects of contexts considerations and the performances of the algorithms are evaluated. The algorithm proposed in the paper is ran over different datasets, by choosing different parameters, we focus on the influence effects, the running time costs, and so on.

One key parameter of the ApproIM-IDC algorithm is the context set C, which actually can be treated to be a constant set in the experiments, since in real applications (e.g., DBLP) the number of contexts is limited. Although the two datasets we used have different properties, the contexts are similar, since both of them are about publications and authors. Therefore, We generated the context sets by first extract contexts from DBLP and LiveJournal by calculating the frequencies of different items, and then combing the two contexts together.

For the two other important parameters U and Θ, the following methods are adopted to generate the parameters. Since not all contexts are equivalent in the aspect of appearing frequencies, we did not obtain the information-related data; therefore, when generating the context sets, we consider the frequencies of contexts in current datasets and try to generate data by similar distributions. For the nodes in the network, besides the context information collected by analyzing the item frequencies, we also generate the contexts based on the context distribution obtained from the data in special probability (0.1 in the experiments), and mix the two sets of contexts together. We do not use the real message in the experiments, since the information diffusion procedure only focuses the context information indeed, random context sets with low, medium, and high cardinalities are generated to represent the real message. For the parameter Θ, in general experiments, it is randomly selected for each node, while it is randomly generated according to the Poisson distribution when the effects of Θ are evaluated. According to the parameter $\lambda = 0.1, 0.3, 0.5$, the corresponding cases can be identified to be IM-IDC-A, IM-IDC-B, and IM-IDC-C, respectively.

Effects of seed set size. The effects of given seed set can be evaluated by the influenced nodes size, that is to verify how many nodes are expected to be influenced in average setting. On the two datasets, we compare the ApproIM-IDC algorithm on different parameters settings of U to distinguish different kinds of information. For each special setting, the seed node set size is changed from low to high, the influence sizes are recorded. The results are shown in Figs. 3 and 4. It can be observed that as the size of seed nodes set increases, the size of influenced nodes increases almost in linear speed. The result is expected since in a network all enough large nodes tend to perform uniformly during the information diffusion. Especially, as discussed in the previous parts, the IDC model has similar representations with the general one which can be obtained by treating each set as the basic considerations of influence sets.

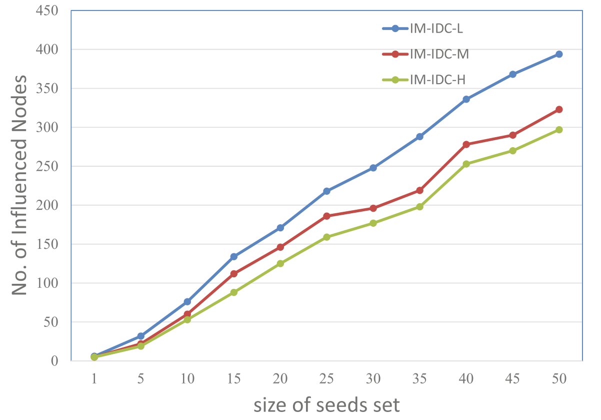

Fig. 3 The No. of influence set size on DBLP data while increasing the size of seed nodes, where IM-IDC-L, IM-IDC-M, and IM-IDC-H are ApproIM-IDC with *low*, *medium*, and *high* context cardinalities

Also, it can be found that when increasing the value of U, that is when we change the context set of users from low to high, the size of influenced nodes set gets smaller. Essentially, when we increase the size of values of $U(\cdot)$, there are mainly two factors which will affect the influence set size. The first is that the probability that each node become progressive from active gets higher; however, in practical, it depends on the tossing coin results not just the probability. Also, the second step about computing f_1 lets the effects of this factor not so obvious. The second factor is that the probability that one user gets active becomes lower. Actually, in real applications, this factor shows the main effect since it causes a main decrease of the influence probabilities, especially when the size of $U(\cdot)$ is relatively small. As shown by the results, when

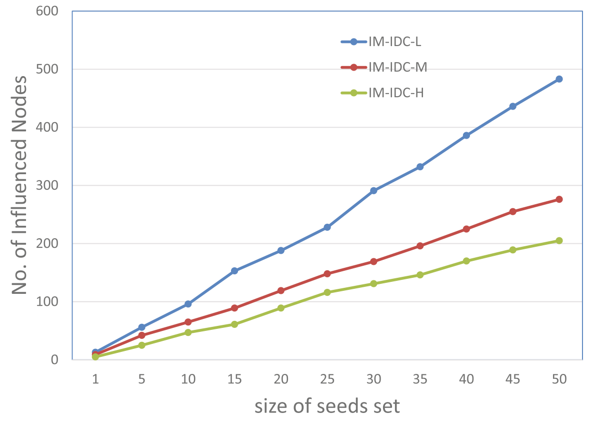

Fig. 4 The No. of influence set size on LiveJournal data while increasing the size of seed nodes, where IM-IDC-L, IM-IDC-M, and IM-IDC-H are ApproIM-IDC with *low*, *medium*, and *high* context cardinalities

increasing the size of value of $U(\cdot)$, the entire trend of the influence set size shows to get lower and lower. Also, the difference between IM-IDC-H and IM-IDC-M is smaller than the one between IM-IDC-L and IM-IDC-M.

Effects of Θ. As discussed above, the parameter set Θ is a key factor affecting the procedure of information diffusion. In this part, the effect of Θ is evaluated by comparing the influence set size for seed node sets with different sizes on the two datasets. When increasing the seed node size from 1 to 50, we ran APPROIM-IDC on the two datasets for three different λ values for the Poisson distribution generator, the influence sizes are reported. The result is shown in Figs. 5 and 6. It can be observed that as the value of seed node size increases, the size of influenced nodes increases, which are expected and observed in the experiments above also. Comparing the results for IM-IDC-A, IM-IDC-B, and IM-IDC-C, it can be observed that as the increase of the parameter λ, the influence set size decreases when the sizes of seed nodes are the same. As shown in the APPROIM-IDC method, it can be known that the value of λ will determine the values of θ of most nodes. The higher the value is, the larger the corresponding θ value is. Then, during the procedure of information diffusion, it will be harder for the node to become progressive, since it needs more progressive neighbor nodes in that case.

Running time. For the real applications, one key challenge of solving the influence maximization problems is the time cost. That is just the reason that we need approximation algorithms whose performance on running time is better than deterministic algorithms. For the two datasets, we ran the APPROIM-IDC algorithm proposed by this paper on different parameters of seed node size. For the parameters of APPROIM-IDC, we picked six combinations which may represent most settings of APPROIM-IDC and give us an illustration about the performances of APPROIM-IDC under different application settings. The running time results are shown in Fig. 7. It can be found that as the size of seeds set increases the running time cost also increases, when seed node size becomes larger the increase speed of running time cost becomes slow. Also, we can find that the parameters such as context cardinalities and Θ have important affections on the performance of APPROIM-IDC on running time costs. Generally speaking, the higher the context cardinalities are, the more effcient the running time performance is, and the higher the values in Θ are, the running time performances of APPROIM-IDC become much better.

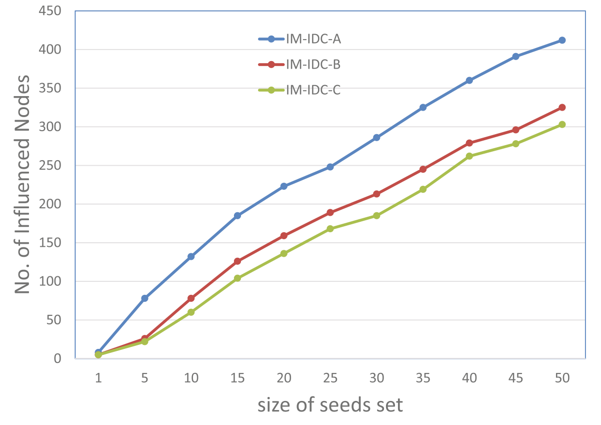

Fig. 5 The No. of influence set size on DBLP data while increasing the size of seed nodes, where IM-IDC-A, IM-IDC-B, and IM-IDC-C are APPROIM-IDC with *low*, *medium*, and *high* Poisson distribution parameters

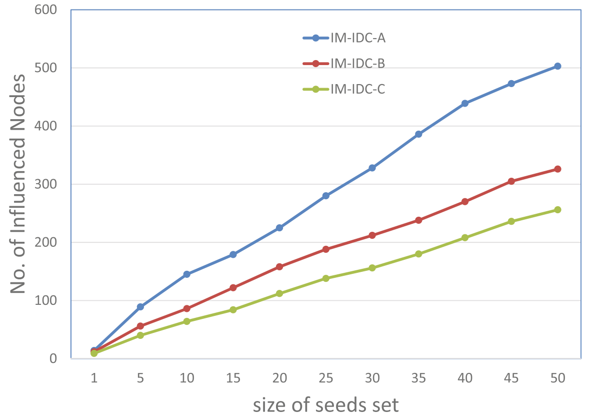

Fig. 6 The No. of influence set size on LiveJournal data while increasing the size of seed nodes, where IM-IDC-A, IM-IDC-B, and IM-IDC-C are APPROIM-IDC with *low*, *medium*, and *high* Poisson distribution parameters

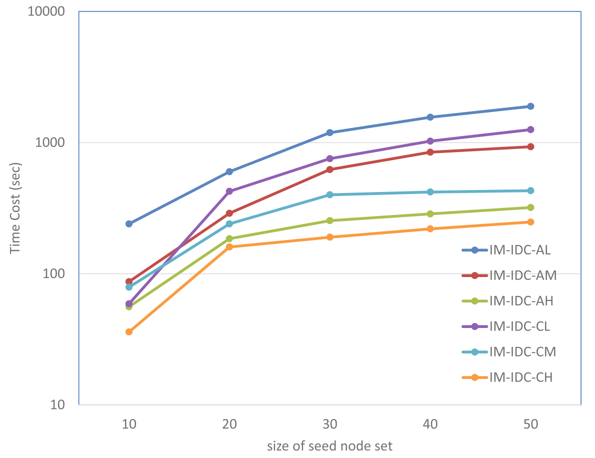

Fig. 7 The running time costs of APPROIM-IDC on LiveJournal data while increasing the size of seed nodes

7 Methods

An approximation algorithm based on greedy strategies is proposed for the influence maximization problem under privacy consideration in social networks. The proposed algorithm utilizes the monotone and submodularity properties of the influence maximization problem. A method for building information diffusion model under privacy protection is proposed. Extension and optimizing methods based on preprocessing and learning are introduced to improve the performance of given algorithms.

8 Related work

The influence maximization is an important and classical problem in the research area of online social networking, which has many applications such as viral marketing and computational advertising. It is firstly studied by Domingo and Richardson [14, 15], and the formalized definitions and comprehensive theoretical analysis are given in [2]. The standard formal definition of influence maximization can be explained as follows, given the constraint that at most k nodes can be selected, the input is a graph which represents the "influence" relationships between nodes, the problem is to compute a set of k nodes such that the number of nodes influenced by the k nodes is maximum. Essentially, the key question is what the network looks like and how the information is diffused. Different models have been formally defined to simulate the information propagation processes with different characteristics; the two most popular models are the *independent cascade* (IC for short) and *linear threshold* (LT for short) models. In [2], the influence maximization problems under both IC and LT models are shown to be NP-hard problems, and the problem of computing the exact influence of given nodes set is shown to be ♯P-hard problem in [1].

After the problem is proposed, many research efforts have been made to find the node set with maximum influence. Kempe et al. [2] proposed an algorithm for influence maximization based on greedy ideas which has constant approximation ratio $(1 - 1/e)$. The time complexity of the greedy approximation algorithm of influence maximization is $O(n^2(m + n))$, which is based on the assumption that influence can be simulated efficiently, but the time cost is still too high in large-scale social networks. To overcome the shortcomings of greedy-based algorithms, [16] proposed CELF (cost-effective lazy-forward) algorithm. CELF can improve the performance of greedy-based algorithms for influence maximization by reducing the times of evaluations of influence set of given seed set; however, its performance on large-scale data is still not satisfying. Using the similar ideas, CELF++ is proposed to improve the performance of algorithms for solving influence maximization by [17]. In [3], degree-discount algorithm is proposed to improve the performance of greedy-based influence maximization algorithms. By assuming all influence probabilities are the same in IC models, [3] reduces the complexities of influence maximization problems and gives better algorithms based on the new models. Utilizing the structural properties of communities in social networks, [18] proposed new algorithms by merging similar nodes and reduce the cost of computing influence set. Goyal et al. [19] proposed SIMPATH algorithm in LT model which improves the performance of greedy-based influence maximization algorithm in LT model. Jiang et al. [20] proposed simulated annealing-based influence maximization algorithms.

There are also many research efforts focusing on the influence maximization over new information diffusion models. For example, [21] proposed new information diffusion models utilizing the idea of finding shortest paths, which assume that although the network structure is complex, the diffusion of information is always processed along the shortest paths. The paper also designs heuristic algorithms for the corresponding influence maximization problem. Using this model, [1] proposed heuristic algorithms based on maximum broadcast paths, which assumes that the information propagated on the network is not transfered by shortest path but maximum broadcast paths. Intuitively, the broadcast paths can be used to estimate the influence range, since the information flow is similar. Therefore, based on the influence probabilities between nodes, a tree structure which reflects the maximum broadcast information transferring way can be built efficiently. Then, by assigning threshold for each node, the tree structure can be used to control which nodes will be important during the information diffusion and by focusing the important nodes, the size of nodes related to the computation of expected influence is reduced. Also, [1] proved the submodularity of influence functions defined based on maximum broadcast paths and designed approximation algorithms with $1 - 1/e$ approximation ratio. In [22], timeliness networks with opportunistic selection are investigated and the information maximization model is extended to those applications. In [23], maximal time bound is considered to limit the abilities of diffusing information in social networks and efficient algorithms for influence maximization problem for computing maximal time bounded positive influence set is proposed. In [18], similarities of nodes of communities in social networks are utilized to reduce the number of nodes involved in the influence computation, and [24] proposed efficient influence maximization algorithms in parallel-computing environments. Cai et al. [25] tries to extend the information maximization models to the applications of crowd-sourced data-based social networks. Han et al. [26] considers the communities in social networks and studies the influence maximization problem over such networks.

There are also many works which try to extend the classic influence maximization methods to other application settings. There has been some work focusing on the problem of influence maximization under location-based social networks. When locations are considered, the influence action will be processed by judging whether two nodes are near enough, and [27] and [28] focus on the problem of finding k users which can influence maximum users in the location-based social network. Topics are important for the information diffusion, [29] and [30] study the problem of influence maximization based on topic-aware considerations. For sensitive information protection considertion, [31] uses the idea of information diffusion to prevent sensitive information in social networks. More related work of applications in social networks can be found in [32, 33].

9 Conclusion

In this paper, based on the privacy considerations of information diffusion process on social networks, the IDC model for diffusing information under privacy protection mechanisms is proposed. By theoretical analysis, we determine the complexities of solving influence maximization on the new model and design efficient algorithms with approximation performance guarantee. By experiments over real dataset, the performances of IDC model and the algorithms proposed are verified. One possible further question is how to provide sophisticated and accurate influence probability predicting algorithms for social networks. Another important question is what is the lower bound of influence maximization problem under privacy consideration? Essentially, it focuses on the optimal approximation ratio for heuristic algorithms of influence maximization problem, which is one direction of our further study.

Endnote

[1] http://snap.stanford.edu/

Abbreviations

IDC: Context-based information diffusion model; IM-IDC: Influence maximization problem on IDC; IC: Independent cascade; LT: Linear threshold

Acknowledgements

Not applicable.

Funding

This work was supported by the National Social Science Foundation of China via grant 14CXW045.

Availability of data and materials

All datasets used for experiments in this paper are collected from the SNAP project of Stanford University, which can be found at http://snap.stanford.edu/.

Authors' contributions

DJ completes the main work and updates the manuscript of this paper. TL gave the main idea of the key method, designed the study, and helped to draft the manuscript. All authors read and approved the final manuscript.

Authors' information

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 21 September 2018 Accepted: 19 March 2019
Published online: 31 May 2019

References

1. W. Chen, C. Wang, Y. Wang, in *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '10*. Scalable influence maximization for prevalent viral marketing in large-scale social networks (ACM, New York, 2010), pp. 1029–1038
2. D. Kempe, J. Kleinberg, E. Tardos, in *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '03*. Maximizing the spread of influence through a social network (ACM, New York, 2003), pp. 137–146
3. W. Chen, Y. Wang, S. Yang, in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '09*. Efficient influence maximization in social networks (ACM, New York, 2009), pp. 199–208
4. L. Liu, J. Tang, J. Han, M. Jiang, S. Yang, in *Proceedings of the 19th ACM International Conference on Information and Knowledge Management, CIKM '10*. Mining topic-level influence in heterogeneous networks (ACM, New York, 2010), pp. 199–208
5. J. Tang, J. Sun, C. Wang, Z. Yang, in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '09*. Social influence analysis in large-scale networks (ACM, New York, 2009), pp. 807–816
6. N. R. Adam, J. C. Worthmann, Security-control methods for statistical databases: a comparative study. ACM Comput. Surv. **21**(4), 515–556 (1989)
7. R. Agrawal, R. Srikant, in *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, SIGMOD '00*. Privacy-preserving data mining (ACM, New York, 2000), pp. 439–450
8. L. Sweeney, K-anonymity: a model for protecting privacy. Int. J. Uncertain. Fuzziness Knowl.-Based Syst. **10**(5), 557–570 (2002)
9. C. Dwork, in *Proceedings of the 33rd International Conference on Automata, Languages and Programming - Volume Part II, ICALP'06*. Differential privacy (Springer, Berlin, Heidelberg, 2006), pp. 1–12
10. M. L. Fisher, G. L. Nemhauser, L. A. Wolsey, *An analysis of approximations for maximizing submodular set functions—II*. (M. L. Balinski, A. J. Hoffman, eds.) (Springer, Berlin, Heidelberg, 1978), pp. 73–87
11. D. Du, in *Proceedings of the 11th International Conference on Combinatorial Optimization and Applications, COCOA'17*. Maximization of multi-factor influence (Springer, Berlin, 2017)
12. T. Mikolov, I. Sutskever, K. Chen, G. Corrado, J. Dean, in *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2, NIPS'13*. Distributed representations of words and phrases and their compositionality (Curran Associates Inc., USA, 2013), pp. 3111–3119
13. G. Jeh, J. Widom, in *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '02*. Simrank: A measure of structural-context similarity (ACM, New York, 2002), pp. 538–543. https://doi.org/10.1145/775047.775126
14. P. Domingos, M. Richardson, in *Proceedings of the Seventh ACM SIGKDD International conference on Knowledge Discovery and Data Mining, KDD '01*. Mining the network value of customers (ACM, New York, 2001), pp. 57–66
15. M. Richardson, P. Domingos, in *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '02*. Mining knowledge-sharing sites for viral marketing (ACM, New York, 2002), pp. 61–70
16. J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, N. Glance, in *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '07*. Cost-effective outbreak detection in networks (ACM, New York, 2007), pp. 420–429
17. A. Goyal, W. Lu, L. V. S. Lakshmanan, in *Proceedings of the 20th International Conference Companion on World Wide Web, WWW '11*. Celf++: Optimizing

the greedy algorithm for influence maximization in social networks (ACM, New York, 2011), pp. 47–48

18. Y.-C. Chen, W.-C. Peng, S.-Y. Lee, Efficient algorithms for influence maximization in social networks. Knowl. Inf. Syst. **33**(3), 577–601 (2012)
19. A. Goyal, W. Lu, L. V. S. Lakshmanan, in *Proceedings of the 2011 IEEE 11th International Conference on Data Mining, ICDM '11*. Simpath: An efficient algorithm for influence maximization under the linear threshold model (IEEE Computer Society, Washington, 2011), pp. 211–220
20. Q. Jiang, G. Song, G. Cong, Y. Wang, W. Si, K. Xie, in *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence, AAAI'11*. Simulated annealing based influence maximization in social networks (AAAI Press, Palo Alto, 2011), pp. 127–132
21. M. Kimura, K. Saito, in *Knowledge-Based Intelligent Information and Engineering Systems: 10th International Conference, KES 2006, Bournemouth, UK, October 9-11, 2006. Proceedings, Part II*, ed. by B. Gabrys, R. J. Howlett, and L. C. Jain. Approximate Solutions for the Influence Maximization Problem in a Social Network (Springer, Berlin, Heidelberg, 2006), pp. 937–944
22. M. Han, M. Yan, Z. Cai, Y. Li, An exploration of broader influence maximization in timeliness networks with opportunistic selection. J. Netw. Comput. Appl. **63**(C), 39–49 (2016). https://doi.org/10.1016/j.jnca.2016.01.004
23. T. Shi, S. Cheng, Z. Cai, Y. Li, J. Li, Retrieving the maximal time-bounded positive influence set from social networks. Pers. Ubiquit. Comput. **20**(5), 717–730 (2016). https://doi.org/10.1007/s00779-016-0943-7
24. J. Kim, S. K. Kim, H. Yu, in *Data Engineering (ICDE), 2013 IEEE 29th International Conference On*. Scalable and parallelizable processing of influence maximization for large-scale social networks? (IEEE Computer Society, Washington, DC, 2013), pp. 266–277
25. J. L. Z. Cai, M. Yan, Y. Li, in *Computer Communications, IEEE INFOCOM 2016-The 35th Annual IEEE International Conference On*. Using crowdsourced data in location-based social networks to explore influence maximization (IEEE, Washington, DC, 2016), pp. 1–9
26. M. Han, M. Yan, Z. Cai, Y. Li, X. Cai, J. Yu, Influence maximization by probing partial communities in dynamic online social networks. Trans. Emerg. Telecommun. Technol. **28**(4), 1–15 (2016)
27. L. Guo, D. Zhang, G. Cong, W. Wu, K.-L. Tan, Influence maximization in trajectory databases. IEEE Trans. Knowl. Data Eng. **29**(3), 627–641 (2017). https://doi.org/10.1109/TKDE.2016.2621038
28. X. Li, X. Cheng, S. Su, C. Sun, Community-based seeds selection algorithm for location aware influence maximization. Neurocomputing. **275**, 1601–1613 (2018). https://doi.org/10.1016/j.neucom.2017.10.007
29. X. Xiong, R. Li, Y. Li, X. Gu, T. Liang, in *Web Information Systems Engineering – WISE 2018*, ed. by H. Hacid, W. Cellary, H. Wang, H.-Y. Paik, and R. Zhou. Topical authority-sensitive influence maximization (Springer, Cham, 2018), pp. 262–277
30. B. Manaskasemsak, R. Phuangpanya, A. Rungsawang, in *Proceedings of the 3rd International Conference on Communication and Information Processing, ICCIP '17*. Topic-constrained influence maximization in social networks (ACM, New York, 2017), pp. 405–410. https://doi.org/10.1145/3162957.3162997
31. Z. Cai, Z. He, X. Guan, Y. Li, Collective data-sanitization for preventing sensitive information inference attacks in social networks. IEEE Trans. Dependable Secure Comput. **PP**(99), 1–1 (2016)
32. W. Han, X. Zhu, Z. Zhu, W. Chen, W. Zheng, J. Lu, A comparative analysis on weibo and twitter. Tsinghua Sci. Technol. **21**(1), 1–16 (2016). https://doi.org/10.1109/TST.2016.7399279
33. R. Bi, Y. Li, X. Zheng, An optimal content caching framework for utility maximization. Tsinghua Sci. Technol. **21**(4), 374–384 (2016). https://doi.org/10.1109/TST.2016.7536715