Information Systems 46 (2014) 1–23

Contents lists available at ScienceDirect

Information Systems

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

# Viral marketing for dedicated customers

Cheng Long [a,*], Raymond Chi-Wing Wong [b]

[a] Rm 4212, The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong
[b] Rm 3542, The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong

ARTICLE INFO

*Article history:*
Received 17 August 2013
Accepted 6 May 2014
Recommended by: F. Carino
Available online 21 May 2014

*Keywords:*
Viral marketing
Interest-Specified
Dedicated customers

ABSTRACT

Viral marketing has attracted considerable concerns in recent years due to its novel idea of leveraging the social network to propagate the awareness of products. Specifically, viral marketing first targets a limited number of users (seeds) in the social network by providing incentives, and these targeted users would then initiate the process of awareness spread by propagating the information to their friends via their social relationships. Extensive studies have been conducted for maximizing the awareness spread given the number of seeds (the *Influence Maximization* problem). However, all of them fail to consider the common scenario of viral marketing where companies hope to use as few seeds as possible yet influencing at least a certain number of users. In this paper, we propose a new problem, called *J-MIN-Seed*, whose objective is to minimize the number of seeds while at least $J$ users are influenced. $J$-MIN-Seed, unfortunately, is NP-hard. Therefore, we develop an approximate algorithm which can provide error guarantees for $J$-MIN-Seed. We also observe that all existing studies on viral marketing assume that *all* users in the social network are *of interest* for the product being promoted (i.e., all users are potential consumers of the product), which, however, is not always true. Motivated by this phenomenon, we propose a new paradigm of viral marketing where the company can specify which types of users in the social network are of interest when promoting a specific product. Under this new paradigm, we re-define our $J$-MIN-Seed problem as well as the *Influence Maximization* problem and design some algorithms with provable error guarantees for the new problems. We conducted extensive experiments on real social networks which verified the effectiveness of our algorithms.



## 1. Introduction

Viral marketing is an advertising strategy that takes the advantage of the effect of "word-of-mouth" among the relationships of individuals to promote a product. Instead of broadcasting to a *massive* number of users directly as existing advertising methods [1] do, viral marketing targets a *limited* number of *initial* users (by providing *incentives*) and utilizes their social relationships, such as friends, families and co-workers, to further spread the awareness of the product among individuals. Each individual who gets the awareness of the product is said to be *influenced*. The number of all influenced individuals corresponds to the *influence* incurred by the initial users. According to some recent research studies [2], people tend to trust the information from their friends, relatives or families more than that from general advertising media like TVs. Hence, it is believed that viral marketing is one of the most effective marketing strategies [3]. In fact, extensive commercial instances of viral marketing succeed in real life. For example, *Nike Inc.* used social networking websites such as *orkut.com* and *facebook.com* to market products successfully [4].

The propagation process of viral marketing within a social network can be described in the following way.

* Corresponding author. Tel.: +852 9512 9172; fax: +852 2358 1477.
*E-mail address:* clong@cse.ust.hk (C. Long).

http://dx.doi.org/10.1016/j.is.2014.05.003
0306-4379/

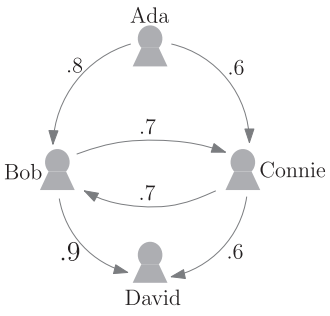



**Fig. 1.** Social network for *J*-MIN-Seed.

At the beginning, the advertiser selects a set of initial users and provides these users incentives so that they are willing to initiate the awareness spread of the product in the social network. We call these initial users *seeds*. Once the propagation is initiated, the information of the product *diffuses* or *spreads* via the relationships among users in the social network. A lot of models about how the above diffusion process works have been proposed [5–10]. Among them, the *Independent Cascade Model* (*IC model*) [5,6] and the *Linear Threshold Model* (*LT model*) [7,8] are the two that are widely used in the literature. In the social network, the IC model simulates the situation where for each influenced user $u$, each of its neighbors has a probability to be influenced by $u$, while the LT model captures the phenomenon where each user's tendency to become influenced increases when more of its neighbors become influenced.

### 1.1. Minimizing seed set

Consider the following scenario of viral marketing. A company wants to advertise a new product via viral marketing within a social network. Specifically, it hopes that at least a certain number of users, says $J$, in the social network must be influenced yet the number of seeds for viral marketing should be as small as possible. Clearly, the above problem can be formalized as follows. *Given a social network $G(V,E)$, we want to find a set of seeds such that the size of the seed set is minimized and at least $J$ users are influenced at the end of viral marketing.* We call this problem *J-MIN-Seed*.

We use Fig. 1 to illustrate the main idea of the *J*-MIN-Seed problem. The four nodes shown in Fig. 1 represent four members in a family, namely Ada, Bob, Connie and David. In the following, we use the terms “nodes” and “users” interchangeably since they correspond to the same concept. The directed edge $(u,v)$ with the weight of $w_{u,v}$ indicates that node $u$ has the probability of $w_{u,v}$ to influence node $v$ for the awareness of the product. Now, we want to find the *smallest* seed set such that *at least* 3 nodes can be influenced by this seed set. It is easy to verify that the expected influence incurred by seed set {Ada} is about 3.57[2] under the IC model and no smaller seed set can incur at least three influenced nodes. Hence, seed set {Ada} is our solution.

*J*-MIN-Seed can be applied to most (if not all) applications of viral marketing. Intuitively, *J*-MIN-Seed asks for the minimum cost (seeds) while satisfying an explicit requirement of revenue (influenced nodes). Clearly, in the mechanism of viral marketing, a seed and an influenced node correspond to cost and potential revenue of a company, respectively, because the company has to *pay* the seeds for incentives, while an influenced node might bring revenue to the company. In many cases, companies face the situation where the goal of revenue has been set up explicitly and the cost should be minimized. Thus, *J*-MIN-Seed meets these companies' demands.

No existing studies have been conducted for *J*-MIN-Seed on the IC model and the LT model. even though it plays an essential role in the viral marketing field. First, most existing studies related to viral marketing focus on maximizing the influence incurred by a certain number of seeds, says $k$ [11–16]. Specifically, they aim at maximizing the number of influenced nodes when only $k$ seeds are available. We denote this problem by *k-MAX-Influence*. Clearly, *J*-MIN-Seed and *k*-MAX-Influence have different *goals* with different given *resources*. Second, though a few studies [17,18] have been done for minimizing the number of seeds while influencing a certain number of users, which is called the *Target Set Selection* (TSS) problem, they adopt the *Deterministic Linear Threshold* (DLT) model as the underlying diffusion model. In contrast, we consider the *Independent Cascade* (IC) model and the *Linear Threshold* (LT) model as the underlying diffusion models for our *J*-MIN-Seed problem. As will be shown later, both the IC model and the LT model enjoy a nice property (*submodularity*), which, however, is not owned by the DLT model, and based on this property, we design an approximate algorithm for *J*-MIN-Seed with good error guarantees. Mainly, [17,18] provide some results about the hardness of approximating the TSS problem, which, do not apply to the *J*-MIN-Seed problem.

Naïvely, we can solve the *J*-MIN-Seed problem [19] by adapting an existing algorithm for *k*-MAX-Influence. Let $k$ be the number of seeds. We set $k=1$ at the beginning and increment $k$ by 1 at the end of each iteration. For each iteration, we use an existing algorithm for *k*-MAX-Influence to calculate the maximum number of nodes, denoted by $I$, that can be influenced by a seed set with the size equal to $k$. If $I \geq J$, we stop our process and return the current number $k$. Otherwise, we increment $k$ by 1 and perform the next iteration. However, this naïve method is very time-consuming since it issues the existing algorithm for *k*-MAX-Influence many times for solving *J*-MIN-Seed. Note that *k*-MAX-Influence is NP-hard [12]. Any existing algorithm for *k*-MAX-Influence is computationally expensive, which results in this naïve method with a high

[2] The expected influence incurred by seed set {Ada} on Bob is $1-(1-0.8)\cdot(1-0.6\cdot 0.7)=0.884$ (note that Ada can influence Bob either (*footnote continued*) directly with the probability equal to 0.8 or via Connie with the probability equal to 0.6∗0.7). Similarly, we can compute the expected influence incurred by {Ada} on other users. Overall, the influence incurred by {Ada} is equal to 3.57.

computation cost. Hence, we should resort to other more efficient solutions.

In this paper, $J$-MIN-Seed is, unfortunately, proved to be NP-hard. Motivated by this, we design an approximate algorithm called *MS-Greedy* for $J$-MIN-Seed. Specifically, MS-Greedy iteratively adds into a seed set one node that generates the greatest influence gain until the influence incurred by the seed set is at least $J$. Besides, we work out an additive error bound and a multiplicative error bound for MS-Greedy.

### 1.2. Interest-Specified viral marketing

Existing studies on viral marketing assume implicitly that *all* users in the social network are *of interest* for a specific product being promoted via viral marketing, which, however, is not true in some cases. For instance, a *young* student might not be of interest for the company when the product being promoted is a product designed for the *old*. In these cases, it is a necessity to provide the company an option to specify which users in the social network are of interest in order to influence the *truly* potential customers effectively. Motivated by this phenomenon, we propose a new paradigm of viral marketing called *Interest-Specified Viral Marketing*, where the company can specify which users are of interest when promoting a specific product. To this purpose, we assume that each user in the social network is associated with a set of attribute values and the company can specify the users to be of interest by providing a set $A_I$ of attribute values. Consequently, all users that contain some attribute values from $A_I$ correspond to the users that are of interest. Note that a user is of interest to a company means that the company has interest in this user (the product of the company is designed for the group of users which includes this user), which further implies that this user would probably be interested in the product. Thus, "user interest" and "company interest" co-exists in our Interest-Specified Viral Marketing paradigm.

Interest-Specified viral marketing is more general than the existing one. Since when all users in the social network are specified to be of interest, the Interest-Specified viral marketing becomes the existing one trivially. Besides, Interest-Specified viral marketing renders a more effective marketing strategy because it provides the option to focus on only the truly potential customers.

Under the paradigm of Interest-Specified Viral Marketing, we propose two problems *Interest-Specified J-MIN-Seed* (IS-$J$-MIN-Seed) and *Interest-Specified k-MAX-Influence* (IS-$k$-MAX-Influence), which are the counterparts of $J$-MIN-Seed and $k$-MAX-Influence, respectively. For both IS-$J$-MIN-Seed and IS-$k$-MAX-Influence, we develop some approximate algorithms which can provide a certain degree of error guarantee.

### 1.3. Contributions

We summarize our contributions as follows:

- To the best of our knowledge, we are the first to propose the $J$-MIN-Seed problem [19] under the IC model and the LT model, which is a fundamental problem in viral marketing.
- Since $J$-MIN-Seed is NP-hard, we develop an approximate algorithm for $J$-MIN-Seed which is proved to provide both additive and multiplicative error guarantees.
- We propose a new paradigm of viral marketing, i.e., *Interest-Specified viral marketing*, is more general and flexible than the existing one. Under this new paradigm, the company can specify which users in the social network are of interest when promoting a specific product.
- We propose two problems IS-$J$-MIN-Seed and IS-$k$-MAX-Influence under the Interest-Specified viral marketing paradigm and develop some approximate algorithms that can provide error guarantees for each of these problems.
- We conducted extensive experiments on real social networks which verified our algorithms and the related theoretical results.

The rest of the paper is organized as follows. We review the related work in Section 2. In Section 3, we first introduce the existing viral marketing paradigm and then propose our Interest-Specified viral marketing paradigm. We study the $J$-MIN-Seed problem, the IS-$J$-MIN-Seed problem and the IS-$k$-MAX-Influence problem in Sections 4, 5 and 6, respectively. We conducted our empirical studies in Section 7 and conclude our paper in Section 8.

## 2. Related work

In Section 2.1, we discuss two widely used *diffusion models* in a social network, and in Section 2.2, we give the related work about the *influence maximization* problem. We briefly review the Target Set Selection (TSS) problem in Section 2.3 and introduce some other relevant works in Section 2.4.

### 2.1. Diffusion models

Given a social network represented in a directed graph $G$, we denote $V$ to be the set containing all the nodes in $G$ each of which corresponds to a user and $E$ to be the set containing all the directed edges in $G$. Each edge $e \in E$ in the form of $(u, v)$ is associated with a weight $w_{u,v} \in [0, 1]$. Different diffusion models have different meanings on weights. In the following, we discuss the meanings for two popular diffusion models, namely the Independent Cascade (IC) model and the Linear Threshold (LT) model.

*Independent Cascade (IC) model* [7,8]. The first model is the Independent Cascade (IC) model. In this model, the influence is based on how a single node influences each of its *single* neighbor. The weight $w_{u,v}$ of an edge $(u, v)$ corresponds to the probability that node $u$ influences node $v$. Let $S_0$ be the initial set of influenced nodes (seeds in our problem). The diffusion process involves a number of *steps* where each step corresponds to the influence spread from some influenced nodes to other non-influenced nodes. At step $t$, all influenced nodes at step $t-1$ remain influenced, and each node that becomes influenced at step $t-1$ for the

*first* time has one chance to influence its non-influenced neighbors. Specifically, when an influenced node $u$ attempts to influence its non-influenced neighbor $v$, the probability that $v$ becomes influenced is equal to $w_{u,v}$. The propagation process halts at step $t$ if no nodes become influenced at step $t-1$. The running example in Fig. 1 is based on the IC model.

For a graph under the IC model, we say that the graph is *deterministic* if all its edges have the probabilities equal to 1. Otherwise, we say that it is *probabilistic*.

*Linear Threshold* (*LT*) *model* [5,6]. The second model is the Linear Threshold (LT) model. In this model, the influence is based on how a single node is influenced by its *multiple* neighbors together. The weight $w_{u,v}$ of an edge $(u,v)$ corresponds to the relative strength that node $v$ is influenced by its neighbor $u$ (among all of $v$'s neighbors). Besides, for each $v \in V$, it holds that $\sum_{(u,v) \in E} w_{u,v} \leq 1$. The dynamics of the process proceeds as follows. Each node $v$ selects a threshold value $\theta_v$ from range $[0,1]$ randomly. Same as the IC model, let $S_0$ be the set of initial influenced nodes. At step $t$, the non-influenced node $v$, for which the total weight of the edges from its *influenced* neighbors exceeds its threshold ($\sum_{(u,v) \in E}$ and $u$ is influenced $w_{u,v} \geq \theta_v$), becomes influenced. The spread process terminates when no more influence spread is possible.

For a graph under the LT model, we say that the graph is *deterministic* if the thresholds of all its nodes have been set before the process of influence spread. Otherwise, we say that it is *probabilistic*.

### 2.2. Influence maximization

There is a long history of study on the information diffusion process among individuals. The first effort devoted to diffusion study is due to Ryan and Gross [20] who discovered that the diffusion is a social process which has strong effects on the farmers' decision making of whether or not to adopt the hybrid corn seeds. More recently, motivated by the fact that the social network plays a fundamental role in spreading ideas, innovations and information, Domingoes and Richardson proposed to use social networks for marketing purpose, which is called viral marketing [11,21]. By viral marketing, they aimed at selecting a limited number of seeds such that the influence incurred by these seeds is maximized. We call this fundamental problem as the *influence maximization* problem.

In [12], Kempe et al. formalized the above influence maximization problem as a discrete optimization problem which corresponds to *k-MAX-Influence. Given a social network $G(V,E)$ and an integer k, find k seeds such that the incurred influence is maximized.* Kempe et al. proved that *k*-MAX-Influence is NP-hard for both the IC model and the LT model. To achieve better efficiency, they provided a $(1-1/e)$-approximation algorithm for *k*-MAX-Influence.

Recently, several studies have been conducted to solve *k*-MAX-Influence in a more efficient and/or scalable way than the aforementioned approximate algorithm in [12]. Specifically, in [13], Leskovec et al. employed a "lazy-forward" strategy to select seeds, which has been shown to be effective for reducing the cost of the approximate algorithm in [12]. In [14], Kimura et al. proposed a new shortest-path cascade model, based on which, they developed efficient algorithms for *k*-MAX-Influence. Motivated by the drawback of non-scalability of all aforementioned solutions for *k*-MAX-Influence, Chen et al. [22] proposed to *re-use* the Monte-Carlo simulation results for estimating the influence spread incurred by different sets of seeds and also proposed a new heuristic called "Degree Discount" for estimating the influence spread efficiently. Chen et al. [15] proved that the problem of computing the influence spread incurred by a set of seeds under the IC model is #P-hard and proposed a heuristic called "PMIA" for estimating the influence spread calculation under the IC model. Chen et al. [23] proved that the problem of computing the influence spread incurred by a set of seeds under the LT model is #P-hard and proposed a heuristic called "LDAG" for estimating the influence spread calculation under the LT model. Wang et al. [24] proposed a community-based method for finding top-k influential users. Narayanam and Narahari [25] proposed a method called "SPIN" which is based on Shapley value computation. Goyal et al. [26] developed an algorithm based on the concept of "simple paths", which provides a new trade-off between the quality and the efficiency for the *k*-MAX-Influence problem. Jiang et al. [27] proposed an approach based on *Simulated Annealing* (SA) for the influence maximization problem under the IC model. Goyal et al. [28] defined a new propagation probability model called "call distribution" model that reveals how influence flows in the networks based on historical data and studied the influence maximized problem based on the proposed model. Jung et al. [29] proposed a new heuristic based on *influence ranking* (IR) and *influence estimation* (IE) for estimating the influence spread calculation under the IC model, which achieves up to two orders of magnitude faster than PMIA. Li et al. [30] studied the influence maximization problem on social networks with not only the friend relationship but also the foe relationship. More recently, Christian et al. [31] proposed a probabilistic algorithm for the *k*-MAX-Influence problem, which gives $a(1-1/e-\epsilon)$-approximation (for any $\epsilon > 0$) and runs in $O((m+n)\epsilon^{-3} \log n)$ time where $n$ ($m$) is the number of users (links) in the social network.

The influence maximization problem has been extended into the setting with multiple products instead of a single product. Bharathi et al. solved the influence maximization problem for multiple *competitive* products using game-theoretical methods [32–40], while Datta et al. proposed the influence maximization problem for multiple *non-competitive* products [16]. Apart from these studies aiming at maximizing the influence, considerable efforts have been devoted to the diffusion models in social networks [9,10].

### 2.3. Target Set Selection

The *Target Set Selection* (TSS) problem was first proposed in [17]. The Target Set Selection problem is to select a set $S$ of seeds such that the *whole* social network is influenced and the size of $S$ is minimized. In the TSS problem, the underling diffusion model is the *Deterministic Linear Threshold* (DLT) model, which is identical to the LT

model except that the thresholds of the nodes in the social network are *fixed* in the DLT model. In contrast, the thresholds of the nodes in the social network are *randomly* determined under the LT model.

A few studies have been devoted to the TSS problem [17,18,41]. In [17], Ning Chen derived some inapproximability results of the TSS problem. Specifically, it was stated in [17] that the TSS problem cannot be approximated within the ratio of $O(2^{\log^{1-\epsilon} n})$ for any fixed $\epsilon > 0$, unless $NP \subseteq DTIME(n^{poly\log(n)})$. In [18,41], Ben-Zwi explored the hardness of the TSS problem by considering the *treewidth* parameter of the graph. They developed an exact algorithm for TSS which runs in $n^{O(w)}$ time, where $w$ is the treewidth of the underlying social network. Recently, several pruning heuristics have been developed for the TSS problem [42,43].

As could be noticed, the *J*-MIN-Seed problem, which will be discussed later, is exactly the TSS problem except that the diffusion models considered in *J*-MIN-Seed are the IC model and the LT model instead of the DLT model. Due to the different underlying diffusion models, the TSS problem and the *J*-MIN-Seed problem have different results. For example, as will be introduced later, a simple greedy algorithm can give a good approximation error that guarantees for the *J*-MIN-Seed problem, which, however, is not the case for the TSS problem [17]. In a recent work [44], Goyal et al. considered the *J*-MIN-Seed problem independently and provided similar results to those in our previous study [19].

### 2.4. Other relevant works

In [45], the authors studied the problem of identifying the most efficient "spreaders" from which, the information will be propagated to a large portion of the social network. In [46], Centola empirically studied the effect of homophilic structures (similarity of social contacts) on the information diffusion process, according to which, homophily significantly increased the overall information diffusion. In [47], the authors presented a method which is used to identify the influence/susceptibility degree of the users in the social networks. In [48], motivated by the phenomenon that the diffusion processes of different products might compete with each other, the authors proposed a scalable framework for incorporating multiple competitive diffusion models in social networks. Some other works studied the problem of determining the optimal size of the seed set. For example, [49] identifies the optimal size of the seed set based on parameters such as the coefficients of innovation and imitation, market potential, discount rate, and gross margin. Stonedahl et al. [50] defined a strategy space for this task by weighting a combination of network characteristics such as average path length, clustering coefficient, and degree. We note here that we adopt a simple yet natural *seeding strategy* for viral marketing, i.e., the seed size is either pre-set (influence maximization), in which case, the above branch of studies could be used for setting the seed size, or regarded as an objective to optimize (seed minimization), in which case, the strategy is intuitive since using fewer seeds saves money for the company. More recently, Saharara et al. [51] studied the viral marketing with the following two settings: (1) the network is evolving and (2) the weights (or strengths) of the edges are product-dependent.

## 3. Viral marketing paradigms

We discuss the existing viral marketing paradigm in Section 3.1 and propose the new paradigm, Interest-Specified viral marketing, in Section 3.2. We illustrate the method for estimating the influence spread corresponding to a seed set in Section 3.3.

### 3.1. Existing viral marketing paradigm

#### 3.1.1. Paradigm

As mentioned in Section 1, in the existing viral marketing paradigm, the company first targets a limited number of seeds and then these seeds would initiate the diffusion process of the product information in the social network automatically. Thus, the most critical problem in viral marketing is to decide which users should be targeted as seeds.

To answer this question, most existing studies [12,15,26] on viral marketing assume such a scenario, where the budget of how many seeds could be targeted is given, e.g., $k$, and the goal is to maximize the influence resulted from the diffusion process initiated by the seeds. The problem corresponding to this scenario is *k-MAX-Influence*.

In this work, we assume another scenario, where the influence requirement has been specified, e.g., at least $J$ users should be influenced, and the goal is to minimize the number of seeds. The problem corresponding to this scenario is called *J-MIN-Seed* and will be formally defined in Section 4.

Given a set $S$ of seeds, we define the *influence* incurred by the seed set $S$ (or simply the influence of $S$), denoted by $\sigma(S)$, to be the expected number of nodes influenced during the diffusion process initiated by $S$.

#### 3.1.2. Properties

Since the analysis of the error bounds of our approximate algorithm for *J*-MIN-Seed, which will be in Section 4, is based on the property that function $\sigma(\cdot)$ is *submodular*, we first briefly introduce the concept of submodular function, denoted by $f(\cdot)$. After that, we provide several properties related to the influence diffusion process in a social network.

**Definition 1** (*Submodularity*). Let $U$ be a universe set of elements and $S$ be a subset of $U$. Function $f(\cdot)$ which maps $S$ to a non-negative value is said to be *submodular* if given any $S \subseteq U$ and any $T \subseteq U$ where $S \subseteq T$, it holds for any element $x \in U - T$ that $f(S \cup \{x\}) - f(S) \geq f(T \cup \{x\}) - f(T)$. □

In other words, we say $f(\cdot)$ is *submodular* if it satisfies the "diminishing marginal gain" property: the marginal gain of inserting a new element into a set $T$ is at most the marginal gain of inserting the same element into a subset of $T$.

According to [12], function $\sigma(\cdot)$ is submodular for both the IC model and the LT model. The main idea is as follows. When we add a new node $x$ into a seed set $S$, the influence incurred by the node $x$ (without considering the nodes in $S$) might overlap with that incurred by $S$. The larger $S$ is, the more overlap might happen. Hence, the marginal gain is smaller on a (larger) set compared to that on any of its subsets. We formalize this statement with the following Property 1.

**Property 1.** *Function $\sigma(\cdot)$ is submodular for both the IC model and the LT model.* □

To illustrate the concept of submodular functions, consider Fig. 1. Assume that a seed set $T$ is $\{Ada\}$. Let a subset $S$ of $T$ be $\emptyset$. We insert into seed sets $T$ and $S$ the same node *Bob*. In fact, it is easy to calculate $\sigma(\emptyset)=0$, $\sigma(\{Ada\})=3.57$, $\sigma(\{Bob\})=2.64$ and $\sigma(\{Ada,Bob\})=3.83$. Consequently, we know that the *marginal gain* of adding a new node *Bob* into set $T$, i.e., $\sigma(\{Ada,Bob\})-\sigma(\{Ada\})=0.26$, is smaller than that of adding *Bob* into one of its subsets $S$, i.e., $\sigma(\{Bob\})-\sigma(\emptyset)=2.64$.

In the $k$-MAX-Influence problem, we have a submodular function $\sigma(\cdot)$ which takes a set of seeds as an input and returns the expected number of influenced nodes incurred by the seed set as an output. Similarly, in the $J$-MIN-Seed problem, we define a function $\alpha(\cdot)$ which takes a set of influenced nodes as an input and returns the smallest number of seeds needed to influence these nodes as an output. One may ask: *Is function $\alpha(\cdot)$ also submodular?* Unfortunately, the answer is "no" which is formalized with the following Property 2.

**Property 2.** *Function $\alpha(\cdot)$ is not submodular for both the IC model and the LT model.* □

Property 2 suggests that we cannot directly adapt existing techniques for the $k$-MAX-Influence problem (which involves a *submodular* function as an objective function) to our $J$-MIN-Seed problem (which involves a *non-submodular* function as an objective function).

### 3.2. Interest-Specified viral marketing paradigm

#### 3.2.1. Paradigm

As could be noticed, in the existing viral marketing paradigm discussed in Section 3.1, it is implicitly assumed that *all* users in the social network are *of interest* for the product being promoted. Specifically, the users in the social network are not differentiated from each other by considering the specific product being promoted, which, however, could cause some problem in those cases, where the product being promoted is not designed for the general people, but for a specialized group of people, says the young. Motivated by this phenomenon, we propose a new paradigm of viral marketing, where people can specify which users in the social network are of interest when promoting products. We call this paradigm *Interest-Specified Viral Marketing*.

The main procedure of viral marketing in this new paradigm is the same as that in the existing paradigm, except that the company should also specify which users in the social network are of interest at the beginning, which will be discussed next.

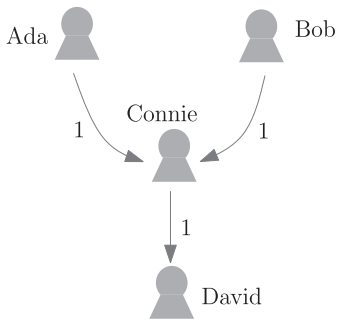



**Fig. 2.** Counterexample ($\alpha(\cdot)$).

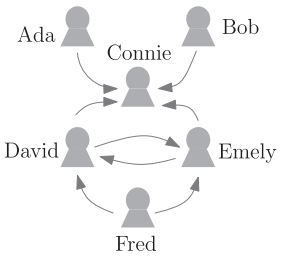
In real-life applications, the company can specify the users of interest in an arbitrary way. For ease of illustration, in this paper, we consider the following way for specifying which users are of interest. We assume that each user in the social network is associated with a set of *attribute values*, e.g., ages, professions and genders. Usually, these attribute values are good indicators of whether a user is of interest for a specific product. For example, when we promote a product aimed at the young, all users that are young can be specified to be of interest. The attribute values could come from different attribute domains. Examples of attributes include gender, age and profession, and their corresponding domains are {male, female}, {1, 2, …, 120} and {salesman, teacher, …}, respectively. For example, the set of attribute values of Ada in Fig. 3 is {female, young, student}, meaning that *she* is a young student.

We formally present the above way of specifying the users of interest as follows. Let $A=\{a_1,a_2,\ldots,a_n\}$ be the attribute domain and $A_I=\{a_{i_1},a_{i_2},\ldots,a_{i_m}\}$ $(1\le i_1<i_2<\cdots<i_m\le n)$, a subset of $A$, be the set of specified attribute values of *interest*. Then, all nodes that contain some attribute values from $A_I$ are regarded to be of interest wrt $A_I$ and we denote the set of users that are of interest wrt $A_I$ by $V_I$. For example, let $A$ be {young, adult, old} and $A_I$ be {young}. Then, all the users that contain the attribute value "young" are of interest. Let $S$ be a set of seeds. At the end of the process of viral marketing with its seed set of $S$, a corresponding set of users in the social network would be influenced. We define $\sigma(S,A_I)$ to be the expected number of influenced users incurred by $S$ that are of interest, where $A_I$ is the set of attribute values of interest. To illustrate, consider the social network presented in Fig. 3(a), where there are six people. The attribute values of each user are shown in Fig. 3(b). We assume that each edge has its weight equal to 1, which indicates that an influenced node $u$ will influence a non-influenced node $v$ immediately if there exists an edge from $u$ to $v$. Let $A_I$ be {young} and $S$ be {Ada}. Then we have $\sigma(S,A_I)=2$. Note that $\sigma(S)$ is equal to $\sigma(S,A)$.

#### 3.2.2. Properties

Same as $\sigma(S)$, $\sigma(S,A_I)$ is also submodular. We present this property in Property 3.

a




b

| Users | Attribute values |
|---|---|
| Ada | female, young, student |
| Bob | male, young, student |
| Connie | female, young, student |
| David | male, middle-aged, teacher |
| Emely | female, middle-aged, lawyer |
| Fred | male, old, retired |

**Fig. 3.** An example for Interest-Specified viral marketing. (a) Social network and (b) attribute values.

**Property 3.** *$\sigma(S, A_I)$ is a submodular function for both the IC model and the LT model.* □

To illustrate, consider the example in Fig. 3. Assume that $A_I$ is {young}. Let $T$ be {Ada} and $S$ be a subset of $T$, $\emptyset$. We have $\sigma(T, A_I) = 2$ since two users that of interest (i.e., Ada and Connie) would be influenced due to the seed set $T$ and $\sigma(S, A_I) = 0$ since no users would be influenced due to an empty seed set. Consider adding a new seed Connie. The marginal gain of adding Connie into $T$, i.e., $\sigma(T \cup \{\text{Connie}\}, A_I) - \sigma(T, A_I)$, is 0, while the marginal gain of adding Connie into $S$, i.e., $\sigma(S \cup \{\text{ Connie }\}, A_I) - \sigma(S, A_I)$, is 1. Thus, we have $\sigma(T \cup \{\text{ Connie }\}, A_I) - \sigma(T, A_I) \leq \sigma(S \cup \{\text{ Connie }\}, A_I) - \sigma(S, A_I)$.

### 3.3. Influence estimation

We discuss the methods for estimating $\sigma(S)$ and $\sigma(S, A_I)$ Sections 3.3.1 and 3.3.2, respectively.

#### 3.3.1. $\sigma(S)$

It is proved in [15,23] that the problem of calculating $\sigma(S)$ given a seed set $S$ is #P-hard. Thus, it is prohibitively expensive to compute $\sigma(S)$ exactly. Usually, in the literature of viral marketing, people adopt *Monte-Carlo simulation* to estimate $\sigma(S)$. The main idea of *Monte-Carlo* simulation is as follows. Let $S$ be the seed set. It simulates the diffusion process initiated by $S$ many times and for each time, it counts the number of influenced users at the end of that diffusion process. Then, it averages the counts to be the estimated $\sigma(S)$. It is shown in [12] that on a social networks with about $10k$ nodes and $50k$ vertices, the accuracy of this method is good enough when the diffusion process is simulated 10,000 times and the gain on the accuracy by performing more simulations, e.g., 20,000 times, is not significant. However, it remains unclear whether the above Monte-Carlo simulation method provides a certain degree of accuracy guarantee.

In this paper, we prove a theoretical result on the accuracy achieved by the Monte-Carlo simulation method and present it in the following lemma.

**Lemma 1.** *Let $c$ be a real number between 0 and 1. Given a social network $G(V, E)$ and a seed set $S$, Monte-Carlo simulation method achieves a $(1 \pm \epsilon)$-approximation of $\sigma(S)$ with the confidence at least $c$ by performing the simulation process at least $(|V|-1)^2 \ln(2/(1-c))/2\epsilon^2|S|^2$ times.* □

Note that our work in this paper is orthogonal to the methods of estimating $\sigma(S)$. Therefore, all existing efficient heuristic-based methods [23,15,26] for estimating $\sigma(S)$ can be adopted for estimating $\sigma(S)$ in this work.

#### 3.3.2. $\sigma(S, A_I)$

Since $\sigma(S, A_I)$ is more general than $\sigma(S)$, we know that $\sigma(S, A_I)$ is at least #P-hard.

Again, we utilize the same idea of Monte-Carlo simulation for estimating $\sigma(S, A_I)$. The only difference is that for each simulation process, instead of counting *all* influenced users, we *only* count the influenced users that are of interest.

## 4. J-MIN-Seed

We define the *J*-MIN-Seed problem in Section 4.1 and introduce an approximate algorithm called *Greedy* for *J*-MIN-Seed in Section 4.2. We provide the theoretical analysis of Greedy in Section 4.3 and discuss two different implementations of Greedy in Section 4.4.

### 4.1. Problem definition

We define the *J*-MIN-Seed problem as follows.

**Problem 1** (*J-MIN-Seed*). Given a social network $G(V, E)$ and an integer $J$, it is to find a set $S$ of seeds such that $|S|$ is minimized and $\sigma(S) \geq J$. □

We say that node $u$ is *covered* by seed set $S$ if $u$ is influenced during the influence diffusion process initiated by $S$. It is easy to see that *J*-MIN-Seed aims at minimizing the number of seeds while satisfying the requirement of covering at least $J$ nodes. Given a node $x$ in $V$ and a subset $S$ of $V$, the *marginal gain* of inserting $x$ into $S$, denoted by $G_x(S)$, is defined to be $\sigma(S \cup \{x\}) - \sigma(S)$.

We show the hardness of *J*-MIN-Seed with the following theorem.

**Theorem 1.** *The J-MIN-Seed problem is NP-hard for both the IC model and the LT model.* □

### 4.2. Approximate algorithm

As proved in Section 3, *J*-MIN-Seed is NP-hard. It is expected that there is no efficient exact algorithm for *J*-MIN-Seed. As discussed in Section 1, if we want to solve *J*-MIN-Seed, a naïve adaption of any existing algorithm

originally designed for $k$-MAX-Influence is time-consuming. The major reason is that it executes an existing algorithm many times and the execution of this existing algorithm for an iteration is *independent* of the execution of the same algorithm for the next iteration. Motivated by this observation, we propose Greedy which solves $J$-MIN-Seed efficiently by executing an iteration based on the results from its previous iteration.

Specifically, Greedy first initializes a seed set $S$ to be an empty set. Then, it selects a non-seed node $u$ such that the marginal gain of inserting $u$ into $S$ is the greatest and then it inserts $u$ into $S$. It repeats the above steps until at least $J$ nodes are influenced. We present Greedy in Algorithm 1.

**Algorithm 1.** Greedy.

**Input**: $G(V,E)$: a social network.
$J$: the required number of nodes to be influenced
**Output**: $S$: a seed set.
1: $S \leftarrow \emptyset$
2: **while** $\sigma(S) < J$ **do**
3: $u \leftarrow \arg\max_{x \in V-S}(\sigma(S \cup \{x\}) - \sigma(S))$
4: $S \leftarrow S \cup \{u\}$
5: **return** $S$

Greedy is similar to the algorithm from [12] for $k$-MAX-Influence except its stopping criterion. The stopping criterion in Greedy is $\sigma(S) \geq J$ while the stopping criterion in the algorithm from [12] is $|S| \geq k$ where $k$ is a user parameter of $k$-MAX-Influence. Besides, they have different theoretical results. Greedy for $J$-MIN-Seed has theoretical results which guarantee the number of *seeds* used while the algorithm for $k$-MAX-Influence has theoretical results which guarantee the number of *influenced nodes*.

### 4.3. Theoretical analysis

In this part, we show that Greedy in Algorithm 1 can return the seed set with both an additive error guarantee and a multiplicative error guarantee.

Greedy gives the following additive error bound.

**Lemma 2** (*Additive error guarantee*). *Let $h$ be the size of the seed set returned by Greedy and $t$ be the size of the optimal seed set for $J$-MIN-Seed. Greedy gives an additive error bound equal to $1/e \cdot J + 1$. That is, $h - t \leq 1/e \cdot J + 1$. Here, $e$ is the natural logarithmic base.* □

Before we give the multiplicative error bound of Greedy, we first give some notations. Suppose that Greedy terminates after $h$ iterations. We denote $S_i$ to be the seed set maintained by Greedy at the end of iteration $i$ where $i = 1, 2, \ldots, h$. Let $S_0$ denote the seed set maintained before Greedy starts (i.e., an empty set). Note that $\sigma(S_i) < J$ for $i = 1, 2, \ldots, h-1$ and $\sigma(S_h) \geq J$.

In the following, we give the multiplicative error bound of Greedy.

**Lemma 3** (*Multiplicative Error Guarantee*). *Let $\sigma'(S) = \min\{\sigma(S), J\}$. Greedy is a $(1 + \min\{k_1, k_2, k_3\})$-approximation of $J$-MIN-Seed, where $k_1 = \ln J/(J - \sigma'(S_{h-1}))$, $k_2 = \ln \sigma'(S_1)/(\sigma'(S_h) - \sigma'(S_{h-1}))$, and $k_3 = \ln(\max\{\sigma'(\{x\})/(\sigma'(S_i \cup \{x\}) - \sigma'(S_i)) | x \in V, 0 \leq i \leq h, \sigma'(S_i \cup \{x\}) - \sigma'(S_i) > 0\})$.* □

According to Lemma 3, the multiplicative error bound of Greedy depends on the execution process of the algorithm. As will be shown in our empirical studies, the theoretical multiplicative error bound of Greedy is usually smaller than 4 and the practical multiplicative error is around 2 in most cases.

### 4.4. Implementations

As can be seen, the efficiency of Greedy in Algorithm 1 relies on the calculation of the influence of a given seed set (operator $\sigma(\cdot)$). However, the influence calculation process for the IC model is #P-hard [15]. Under such a circumstance, we adopt the Monte-Carlo simulation method discussed in Section 3.3 when using operator $\sigma(\cdot)$. We denote this implementation by *SM-Greedy1*.

In fact, we have an alternative implementation of Greedy as follows. Instead of sampling the social network to be deterministic *when* calculating the influence incurred by a given seed set, we can sample the social network to generate a certain number of deterministic graphs *only* at the beginning. Then, we solve the $J$-MIN-Seed problem on each such deterministic graph using Greedy, where the cost of operator $\sigma(\cdot)$ simply becomes the time to traverse the graph.

At the end, we return the average of the sizes of the seed sets returned by Greedy based on all samples (deterministic graphs). We denote this alternative implementation by *SM-Greedy2*. Note that with the SM-Greedy2 implementation, we can only obtain the average size of the seed sets, but not a real seed set.

## 5. Interest-Specified J-MIN-Seed

We provide the formal definition of Interest-Specified $J$-MIN-Seed in Section 5.1 and design some approximate algorithms for it in Section 5.2.

### 5.1. Problem definition

Recall that given a positive number $J$, the $J$-MIN-Seed problem is to select a set $S$ of seeds such that the number of influenced nodes, i.e., $\sigma(S)$, is at least $J$ and $|S|$ is minimized. We denote its counterpart in the paradigm of Interest-Specified viral marketing by *Interest-Specified J-MIN-Seed* (*IS-J-MIN-Seed*). The goal part of IS-$J$-MIN-Seed is the same as $J$-MIN-Seed, i.e., selecting as few seeds as possible, while the constraint part of IS-$J$-MIN-Seed is different from that of $J$-MIN-Seed. Instead of imposing only one *overall* requirement of influencing at least a certain number $J$ users as $J$-MIN-Seed does, IS-$J$-MIN-Seed enforces multiple requirements on the number of influenced users, one for each attribute value of interest. Specifically, for each attribute value of interest, $a_{i_l}$ $(1 \leq l \leq m)$, it is required to influence at least a certain number $j_l$ of users containing the attribute value $a_{i_l}$. We provide the formal definition of the IS-$J$-MIN-Seed problem in the following problem.

**Problem 2** (*Interest-Specified J-MIN-Seed*). Given a set of $m$ positive integers $\mathcal{J} = \{j_1, j_2, \ldots, j_m\}$, it is to find a set

$S$ of seeds such that $\sigma(S,\{a_{i_l}\}) \geq j_l$ for $1 \leq l \leq m$ and $|S|$ is minimized. □

To illustrate, consider our running example in Fig. 3. Suppose that a company has a stock of three products designed for *young* people. In order to sell out these products, it wants to influence at least three users that are young while the cost (the number of seeds) for viral marketing is minimized. This problem is essentially an instance of IS-*J*-MIN-Seed where $\mathcal{J}=\{3\}$ and $a_{i_1}$ is set to be "young". It can be verified that the solution of this problem instance is {Ada, Bob}, since {Ada, Bob} is the smallest seed set that can incur (at least) three influenced users that are young (i.e., Ada, Bob and Connie). Note that *J*-MIN-Seed cannot be adopted in this scenario since it provides no option for the company to specify which kinds of users in the social network are of interest.

IS-*J*-MIN-Seed is more general than *J*-MIN-Seed in the sense that when all the users in the social network are assumed to have a common attribute value of interest, say $a_{i_1}$, and $\mathcal{J}$ is set to be $\{J\}$, IS-*J*-MIN-Seed becomes *J*-MIN-Seed exactly.

### 5.2. Approximate algorithms

The IS-*J*-MIN-Seed problem can be regarded as an optimization problem with $m$ constraints, where the constraints are $\sigma(S,\{a_{i_l}\}) \geq j_l$ for $1 \leq l \leq m$ and the objective is to minimize $|S|$. Considering IS-*J*-MIN-Seed is more general than *J*-MIN-Seed and *J*-MIN-Seed is NP-hard, we know that IS-*J*-MIN-Seed is NP-hard as well. In order to solve the IS-*J*-MIN-Seed problem efficiently, in the following, we explore three approximate algorithms for IS-*J*-MIN-Seed, namely *MS-Independent* (Section 5.2.1), *MS-Incremental* (Section 5.2.2) and *MS-Greedy* (Section 5.2.3).

#### 5.2.1. Algorithm MS-Independent

The main idea of algorithm MS-Independent is to satisfy the $m$ constraints of IS-*J*-MIN-Seed *independently*. Specifically, MS-Independent finds a seed set $S_1$ such that $\sigma(S_1,\{a_{i_1}\}) \geq j_1$. Then, it finds a seed set $S_2$ such that $\sigma(S_2,\{a_{i_2}\}) \geq j_2$. Similarly, it finds a seed set $S_l$ such that $\sigma(S_l,\{a_{i_l}\}) \geq j_l$ for $3 \leq l \leq m$. Finally, it constructs a seed set $S$ as the union of $S_l$ $(1 \leq l \leq m)$, i.e., $S = \cup_{1 \leq l \leq m} S_l$, and returns $S$ as the (approximate) solution. It is easy to verify that $S$ satisfies all the constraints of the IS-*J*-MIN-Seed problem, i.e., $\sigma(S,\{a_{i_l}\}) \geq j_l$ for $1 \leq l \leq m$, using the monotonicity of $\sigma(S,\{a_{i_l}\})$. We provide the framework of MS-Independent in Algorithm 2.

**Algorithm 2.** MS-Independent.

1: $S \leftarrow \emptyset$
2: **for** $l=1$ to $m$ **do**
3: find a seed set $S_l$ such that $\sigma(S,\{a_{i_l}\}) \geq j_l$
4: $S \leftarrow \cup_{1 \leq l \leq m} S_l$
5: **return** seed set $S$

One remaining issue of MS-Independent is how to find a seed set $S_l$ such that $\sigma(S,\{a_{i_l}\}) \geq j_l$ (line 3 in Algorithm 2). A trivial solution is to include all users containing attribute value $a_{i_l}$ into $S_l$. However, this would result in the seed set $S$ returned by MS-Independent to be the set of *all* users of interest, which is a trivial solution of the IS-*J*-MIN-Seed problem. Roughly, the smaller $S_l$ it finds, the more *likely* the seed set $S$ returned by MS-Independent is smaller since $S$ is the union of $S_l$ $(1 \leq l \leq m)$.

In this paper, we use a simple greedy algorithm for this purpose. Specifically, it first sets $S_l$ to be $\emptyset$. Then, it proceeds with iterations and at each iteration, it picks the node $v_m$ that incurs the largest gain of influencing users with the attribute value $a_{i_l}$ among all nodes, i.e., $v_m = \arg\max_{x \in V-S}\{\sigma(S \cup \{x\},\{a_{i_l}\}) - \sigma(S,\{a_{i_l}\})\}$ and inserts $v_m$ into $S_l$. It stops when the number of influenced users with attribute $a_{i_l}$ is at least $j_l$, i.e., $\sigma(S,\{a_{i_l}\}) \geq j_l$.

Before providing the approximation factor of MS-Independent, we introduce some notations first. Consider the above greedy procedure for finding $S_l$. Assume that it runs with $r_l$ iterations in total. We use $S_l^h$ to represent the seed set at the end of iteration $h$ $(1 \leq h \leq r_l)$. Besides, we define $\sigma'_a(S,\{a_{i_l}\})$ to be $\min\{\sigma(S,\{a_{i_l}\}), j_l\}$ for $1 \leq l \leq m$. We provide the approximation factor of MS-Independent in Lemma 4.

**Lemma 4.** *For a problem instance of IS-J-MIN-Seed, let $S^*$ be the optimal solution and $S$ be the solution returned by MS-Independent. Let $S_x$ be the largest seed set among $S_l$ $(1 \leq l \leq m)$. We have $|S|/|S^*| \leq m \cdot (1+\min\{t_x^1, t_x^2, t_x^3\})$, where*

$$t_x^1 = \ln \frac{j_x}{j_x - \sigma'_a(S_x^{r_x-1},\{a_{i_x}\})},$$

$$t_x^2 = \ln \frac{\sigma'_a(S_x^1,\{a_{i_x}\})}{\sigma'_a(S_x^{r_x},\{a_{i_x}\}) - \sigma'_a(S_x^{r_x-1},\{a_{i_x}\})}$$

*and*

$$t_x^3 = \ln \max\left\{ \frac{\sigma'_a(\{v\},\{a_{i_x}\})}{\sigma'_a(S_x^h \cup \{v\},\{a_{i_x}\}) - \sigma'_a(S_x^h,\{a_{i_x}\})} \middle| 1 \leq h \leq r_x, \right.$$
$$\left. v \in V,\ \sigma'_a\left(S_x^h \cup \{v\},\{a_{i_x}\}\right) - \sigma'_a\left(S_x^h,\{a_{i_x}\}\right) > 0 \right\}. \quad \square$$

As will be shown in our experiments, the approximation error bound of MS-Independent is usually small, say around 3.

#### 5.2.2. Algorithm MS-Incremental

MS-Independent tries to satisfy each constraint of the IS-*J*-MIN-Seed problem *independently*. In other words, when finding $S_{l_2}$ to influence at least $j_{l_2}$ users with the attribute value $a_{i_{l_2}}$, it does not take the previously found seed sets $S_{l_1}$ $(l_1 < l_2)$ into consideration, which might also incur some influenced users with attribute $a_{i_{l_2}}$. Thus, the constraint of influencing at least $j_{l_2}$ users containing attribute value $a_{i_{l_2}}$ might be satisfied *excessively*.

Motivated by this, we propose the second approximate solution, MS-Incremental, for IS-*J*-MIN-Seed as follows. As its name implies, MS-Incremental tries to satisfy the constraints of IS-*J*-MIN-Seed one after one *incrementally*. Specifically, it maintains a set $S$ to store the selected seeds. Initially, $S$ is set to $\emptyset$. Then, it proceeds with $m$ stages (Stage $1, 2, \ldots, m$). At Stage 1, it selects some seeds and inserts them into $S$ such that the constraint of influencing at least $j_1$ users containing attribute value $a_{i_1}$ is satisfied (i.e., $\sigma(S,\{a_{i_1}\}) \geq j_1$). At Stage 2, it selects some other seeds

and inserts them into $S$ such that the constraint of influencing at least $j_2$ users containing attribute $a_{i_2}$ is satisfied (i.e., $\sigma(S,\{a_{i_2}\})\geq j_2$). It continues in the same manner for the remaining stages. As can be noted, when satisfying the constraint of influencing at least $j_l$ users containing attribute value $a_{i_l}$ at Stage $l$, it takes into consideration those seeds that have been selected for satisfying the previously satisfied constraints.

One remaining issue of MS-Incremental is, at Stage $l$, how to select some seeds together with the previously selected seeds such that the constraint of influencing at least $j_l$ users containing attribute value $a_{i_l}$ is satisfied. In this paper, we consider a similar greedy procedure as that for MS-Independent as follows. It repeatedly picks the user that incurs the largest marginal gain of satisfying the constraint of influencing at least $j_l$ users containing attribute value $a_{i_l}$ and inserts it into $S$ until this constraint is satisfied (i.e., $\sigma(S,\{a_{i_l}\})\geq j_l$). We present MS-Incremental in Algorithm 3.

**Algorithm 3.** MS-Incremental.

```
1: S←∅; l←1
2: while l ≤ m do
3:   v_m ← arg max_{v∈V−S}{σ(S∪{v},{a_{i_l}}) − σ(S,{a_{i_l}})}
4:   S ← S∪{v_m}
5:   if σ(S,{a_{i_l}}) ≥ j_l then
6:     //Change to satisfy the next constraint
7:     l ← l+1
8: return seed set S
```

### 5.2.3. Algorithm MS-Greedy

In this part, we introduce the third approximate solution, MS-Greedy, for IS-$J$-MIN-Seed. Different from MS-Independent and MS-Incremental which try to satisfy the constraints *individually*, MS-Greedy attempts to satisfy the constraints *collectively*.

Let $S$ be a seed set. MS-Greedy sets $S$ to $\emptyset$ initially. Then, it proceeds with iterations and at each iteration, it picks the node $v_m$ that incurs the largest "gain" and inserts $v_m$ into $S$. Here, the "gain" of a node $v$ based on a seed set $S$ is defined to be the *new contribution* of satisfying the remaining un-satisfied constraints made by $v$. We define the *valid* contribution of satisfying the constraint of influencing at least $j_l$ users containing attribute value $a_{i_l}$ made by $v$ to be $\min\{\sigma(S\cup\{v\},\{a_{i_l}\}),j_l\}-\min\{\sigma(S,\{a_{i_l}\}),j_l\}$. Note that if $\sigma(S,\{a_{i_l}\})\geq j_l$, that is, the constraint of influencing at least $j_l$ users that contain attribute $a_{i_l}$ is satisfied, then the valid contribution of including any additional seed is simply 0. As a result, the new contribution of satisfying *all* un-satisfied constraints made by $v$ is equal to $\sum_{1\leq l\leq m}(\min\{\sigma(S\cup\{v\},\{a_{i_l}\}),j_l\}-\min\{\sigma(S,\{a_{i_l}\}),j_l\})$, i.e., $\sum_{1\leq l\leq m}(\sigma'_a(S\cup\{v\},\{a_{i_l}\})-\sigma'_a(S,\{a_{i_l}\}))$. In the following, we define $\sigma'_a(S)$ as $\sum_{1\leq l\leq m}\sigma'_a(S,\{a_{i_l}\})$. Then, the gain of including a node $v$ into a seed set $S$ becomes $\sigma'_a(S\cup\{v\})-\sigma'_a(S)$. MS-Greedy terminates when all constraints are satisfied. We present MS-Greedy in Algorithm 4.

**Algorithm 4.** MS-Greedy.

```
1: S ← ∅
2: while there exist an unsatisfied constraint do
3:   v_m ← arg max_{v∈V−S}{σ'_a(S∪{v}) − σ'_a(S)}
4:   S ← S∪{v_m}
5: return seed set S
```

Before introducing the approximation factor of MS-Greedy, we define some notations first. Assume that MS-Greedy runs with $r$ iterations. Let $S_h$ denote the seed set at the end of iteration $h$ ($1\leq h\leq r$). We provide the approximation factor of MS-Greedy in Lemma 5.

**Lemma 5.** *For a problem instance of IS-J-MIN-Seed, let $S^*$ be the optimal solution and $S$ be the approximate solution returned by MS-Greedy. We have $|S|/|S^*|\leq 1+\min\{t^1,t^2,t^3\}$, where*

$$t^1=\ln\frac{J_{sum}}{J_{sum}-\sigma'_a(S_{r-1})},\quad t^2=\ln\frac{\sigma'_a(S_1)}{\sigma'_a(S_r)-\sigma'_a(S_{r-1})},$$

$$t^3=\ln\max\left\{\frac{\sigma'_a(\{v\})}{\sigma'_a(S_h\cup\{v\})-\sigma'_a(S_h)}\middle|1\leq h\leq r,\ v\in V,\ \sigma'_a(S_h\cup\{v\})-\sigma'_a(S_h)>0\right\}$$

*and*

$$J_{sum}=\textstyle\sum_{1\leq l\leq m}j_l.\qquad\square$$

As will be shown in our experiments, the approximation error bound of MS-Greedy is usually small, say around 2.5.

## 6. Interest-Specified k-MAX-Influence

We provide the formal definition of Interest-Specified $k$-MAX-Influence in Section 6.1 and design an approximate algorithm for it in Section 6.2

### 6.1. Problem definition

Recall that given a positive integer $k$, the $k$-MAX-Influence problem is to select a set $S$ of at most $k$ seeds such that the number of influenced nodes incurred by $S$ is maximized. We denote its counterpart in the paradigm of Interest-Specified viral marketing by *Interest-Specified k-MAX-Influence* (*IS-k-MAX-Influence*). The constraint part of IS-$k$-MAX-Influence is the same as that of $k$-MAX-Influence, i.e., at most $k$ seeds can be selected, while the goal of IS-$k$-MAX-Influence is more concise than that of $k$-MAX-Influence. Specifically, instead of maximizing the *overall* number of influenced users, i.e., $\sigma(S)$, as $k$-MAX-Influence does, IS-$k$-MAX-Influence maximizes *only* the number of the influenced users that are of interest, i.e., $\sigma(S,A_I)$. The formal definition of IS-$k$-MAX-Influence is provided in the following problem.

**Problem 3** (*Interest-Specified k-MAX-Influence*). Given a positive integer $k$, it is to find a seed set $S$ such that $|S|\leq k$ and $\sigma(S,A_I)$ is maximized. □

To illustrate, consider the following scenario. Suppose that a company is promoting a product which is designed only for *young* people. Now, the company wants to launch a viral marketing procedure based on the social network in Fig. 3. Due to the limited budget, the company can select at most two seeds. Since the product is aimed at the young, only the users who are young are of *interest* to this company. This problem is essentially an instance of IS-$k$-MAX-Influence, where $A_I=\{\text{young}\}$ and $k=2$. As can be verified, the solution of this IS-$k$-MAX-Influence is

{Ada, Bob} since the number of influenced users that are of interest is 3 (i.e., Ada, Bob and Connie) which is maximized. Note that $k$-MAX-Influence cannot be adopted in this scenario. This is because if $k$-MAX-Influence is adopted and $k$ is set to be 2, the corresponding solution is {Ada, Fred} incurring five users (Ada, Connie, David, Emely, Fred), among which, however, only two (i.e., Ada and Connie) are of interest to the company.

IS-$k$-MAX-Influence is more general than $k$-MAX-Influence in the sense that when $A_I$ is set to be $A$, i.e., all the users in the social network are specified to be of interest, IS-$k$-MAX-Influence becomes $k$-MAX-Influence exactly.

### 6.2. Approximate algorithm

As mentioned previously, IS-$k$-MAX-Influence is more general than $k$-MAX-Influence. It is proved that the $k$-MAX-Influence problem is NP-hard [12]. Therefore, the Interest-Specified $k$-MAX-Influence problem is NP-hard as well.

To solve IS-$k$-MAX-Influence efficiently, we provide an approximate solution, a greedy algorithm called *MI-Greedy*, which provides an approximation factor equal to $(1-1/e)$. Initially, MI-Greedy sets the seed set $S$ to be $\varnothing$. Then, it proceeds with $k$ iterations. At each iteration, the user that incurs the largest marginal gain among all non-seeds in $V$ is selected and inserted into $S$. We present the MI-Greedy algorithm in Algorithm 5.

**Algorithm 5.** MI-Greedy.

1: $S \leftarrow \varnothing$
2: **for** $i=1$ to $k$ **do**
3: $v_m \leftarrow \arg\max_{x \in V-S}\{\sigma(S \cup \{x\}, A_I) - \sigma(S, A_I)\}$
4: $S \leftarrow S \cup \{v_m\}$
5: **return** seed set $S$

Given a positive number $k$, the Interest-Specified $k$-MAX-Influence problem is to select a set $S$ of $k$ seeds such that the number of influenced users of interest, $\sigma(S, A_I)$, is maximized. That is, Interest-Specified $k$-MAX-Influence can be regarded as the problem of maximizing the submodular function $\sigma(S, A_I)$ where $S \subset V$ and $|S| \leq k$. By using Lemma 7 (provided in the appendix), we know that the MI-Greedy provides an approximation factor equal to $(1-1/e)$ for IS-$k$-MAX-Influence. We formalize this result in the following lemma.

**Lemma 6.** *Algorithm MI-Greedy provides the approximation factor of* $(1-1/e)$ *for the Interest-Specified k-MAX-Influence problem, where e is the base of the natural logarithm.* □

## 7. Empirical study

We set up our experiments in Section 7.1 and give the corresponding experimental results in Section 7.2.

### 7.1. Experimental setup

We conducted our experiments on a 2.26 GHz machine with 4 GB memory under a Linux platform. All algorithms were implemented in C/C++.

#### 7.1.1. Datasets

We used five real datasets for our empirical study, namely *HEP-T*, *Epinions*, *Amazon*, *DBLP* and *Twitter*.

The first four datasets are used for *J*-MIN-Seed. HEP-T is a collaboration network generated from "High Energy Physics-Theory" section of the e-print arXiv (http://www.arXiv.org). In this collaboration network, each node represents one specific author and each edge indicates a co-author relationship between the two authors corresponding to the nodes incident to the edge. The second one, Epinions, is a *who-trust-whom* network at Epinions.com, where each node represents a member of the site and the link from member $u$ to member $v$ means that $u$ trusts $v$ (i.e., $v$ has a certain influence on $u$). The third real dataset, Amazon, is a product co-purchasing network extracted from Amazon.com with nodes and edges representing products and co-purchasing relationships, respectively. We believe that product $u$ has an influence on product $v$ if $v$ is purchased often with $u$. Both Epinions and Amazon are maintained by Jure Leskovec. Our fourth real dataset, DBLP, is another collaboration network of computer science bibliography database maintained by Michael Ley.

For Interest-Specified Viral Marketing, we use two datasets, Twitter and HEP-T, where each node is associated with a set of attribute values. Twitter corresponds to a social network crawled from website twitter.com. In this social network, each node represents a user and each edge $(u, v)$ indicates that user $v$ is a follower of user $u$. Besides, each user contains a set of attribute values, such as his/her profession, gender and location information. HEP-T is exactly the collaboration network from "High Energy Physics-Theory" mentioned above except that for each node $v$ in the social network, we randomly pick an attribute from set {young, middle-aged, senior, old} and assign it as $v$'s attribute. For simplicity, we keep the name of "HEP-T" for this dataset. We summarize the features of the above real datasets in Table 1.

For efficiency, we ran our algorithms on the samples of the aforementioned real datasets with the sampling ratio equal to one percent. The sampling process is done as follows. We randomly choose a node as the root and then perform a breadth-first traversal (BFT) from this root. If the BFT from one root cannot cover our targeted number of nodes, we continue to pick more new roots randomly and perform BFTs from them until we obtain our expected number of nodes. Next, we construct the edges by keeping the original edges between the nodes traversed. However, we note here that by equipping some efficient heuristic-based methods [15,26] for influence estimation ($\sigma(S)$ and $\sigma(S, A_I)$), the techniques developed in this paper could be scalable to large social networks.

#### 7.1.2. Configurations

- *Weight generation for the IC model*. We use the QUADRIVALENCY model to generate the weights. Specifically, for each edge, we uniformly choose a value from set $\{0.1, 0.25, 0.5, 0.75\}$, each of which represents minor, low, medium and high influence, respectively.
- *Weight generation for the LT model*. For each node $u$, let $d_u$ denote its in-degree, we assign the weight of each

**Table 1**
Statistics of real datasets.

| Dataset | HEP-T | Epinions | Amazon | DBLP | Twitter |
|---|---|---|---|---|---|
| No. of nodes | 15,233 | 75,888 | 262,111 | 654,628 | 1977 |
| No. of edges | 58,891 | 508,837 | 1,234,877 | 1,990,259 | 8,846,476 |
| Average degree | 4.12 | 6.1 | 13.4 | 9.4 | 884 |
| Maximal degree | 64 | 588 | 3079 | 425 | 2513 |
| No. of connected component | 1781 | 73 K | 11 | 1 | 165 |
| Largest component size | 6794 | 517 K | 76 K | 262 K | 1824 |
| Average component size | 8.6 | 9.0 | 6.9 K | 262 K | 12 |

edge to $u$ as $1/d_u$. In this case, each node obtains the equivalent influence from each of its neighbors.

- *No. of times for Monte-Carlo simulation*. For each influence calculation under both the IC model and the LT model, we perform the simulation process 10,000 times by default.
- *Parameter J*. In the following, we denote parameter $J$ as a *relative* real number between 0 and 1 denoting the fraction of the influenced nodes among all nodes in the social network (instead of an *absolute* positive integer denoting the total number of influenced nodes) because a relative measure is more meaningful than an absolute measure in the experiments. We set $J$ to be 0.5 by default. Alternative configurations considered are $\{0.1, 0.25, 0.5, 0.75, 1\}$.
- *Attribute domain A*. For Twitter, we use set {male, female}, while for the HEP-T dataset, we use set {young, middle-aged, senior, old}.
- *Set of attributes of interest* $A_I$. We set $A_I$ to {male} and {middle-aged, senior} on default for Twitter and HEP-T, respectively.
- *Parameter k for Interest-Specified k-MAX-Influence*. $k$ is varied from set $\{5, 10, 15, 20, 25\}$.
- *Parameter* $\mathcal{J} = \{j_1, j_2, \ldots, j_m\}$ *for Interest-Specified J-MIN-Seed*. We first define a parameter $\gamma$, which follows normal distribution $\mathcal{N}[\mu, \delta]$. Then, we set $j_k$ to $\gamma \cdot N_k$, where $N_k$ is the number of users containing attribute value $a_{i_k}$. In our experiments, $\mu$ varies from set $\{0.1, 0.25, 0.5, 0.75, 1\}$ and $\delta$ is set to 0.1 on default.

#### 7.1.3. Algorithms

*J-MIN-Seed*: The following shows the algorithms for $J$-MIN-Seed.

- *Greedy1* corresponds to the first implementation of Greedy as introduced in Section 4.4.
- *Greedy2* corresponds to the alternative implementation of Greedy as introduced in Section 4.4.
- *Random* corresponds to the method which repeatedly selects the seeds from the un-covered nodes at random until $J$ users have been influenced. Correspondingly, we denote it by *Random*.
- *Degree-heuristic* corresponds to the method which repeatedly picks the node with the largest out-degree yet un-covered and adds it into the seed set until the incurred influence exceeds the threshold.
- *Centrality-heuristic* is another heuristic method which uses *distance centrality* as the heuristic. In sociology, distance centrality is a common measurement of nodes' importance in a social network based on the assumption that a node with short distances to other nodes would probably have a higher chance to influence them. Centrality-heuristic repeatedly selects the seeds in a decreasing order of nodes' distance centralities until the requirement of influencing at least $J$ nodes is met.

In the experiment, we do not compare our algorithms with the naïve adaption of an existing algorithm for $k$-MAX-Influence described in Section 1 because this naïve adaption is time-consuming as discussed in Section 4.

*Interest-Specified J-MIN-Seed*: The following shows the algorithms for Interest-Specified $J$-MIN-Seed.

- *MS-Independent*, *MS-Incremental* and *MS-Greedy* are our proposed algorithms.
- *MS-Random* is the first baseline which repeatedly selects seeds at random until all the constraints are satisfied.
- *MS-Degree* is the second baseline which adopts the heuristic of *out-degree* for selecting seeds.
- *MS-Centrality* is the third baseline which uses the heuristic of *distance centrality* for selecting seeds.

*Interest-Specified k-MAX-Influence*: The following shows the algorithms for Interest-Specified $k$-MAX-Influence.

- *MI-Greedy* is our proposed algorithm MI-Greedy.
- *MI-Random* is the first baseline which selects $k$ seeds randomly.
- *MI-Degree* represents the second baseline which selects as seeds the $k$ nodes whose *out-degrees* are among the top-$k$.
- *MI-Centrality* is our third baseline which selects as seeds the $k$ nodes whose *centralities* are among the top-$k$.

### 7.2. Experimental results

#### 7.2.1. Experiments for J-MIN-Seed

We use three measurements, namely the *no. of seeds*, *running time* and *memory*. The no. of seeds measurement is used for measuring the effectiveness of the algorithms for

$J$-MIN-Seed. Specifically, the fewer seeds an algorithm returns, the better the algorithm is. The running time and memory correspond to two common measurements on algorithms. The main memory usage of all algorithms in our experiments is to store the underlying social network. We also conducted the experiments on the errors incurred by our algorithms. All these experiments are presented as follows.

*No. of seeds*. We vary parameter $J$ from 0.1 to 1. The experimental results for the IC model are shown in Fig. 4. Consider the results on HEP-T (Fig. 4(a)) as an example. We find that algorithms Greedy1 and Greedy2 are comparable in terms of quality. Both of them outperform other baselines significantly. Similar results can be found in other real datasets.

For the LT model, we conducted the similar experiments, whose results are shown in Fig. 5. Same as the IC model, Greedy1 and Greedy2 beat other algorithms by an order of almost one magnitude in terms of the no. of seeds returned by the algorithms.

*Running time*. Again, we vary $J$. For the IC model, according to the results shown in Fig. 6, we find that Greedy1 is the slowest algorithm. This is reasonable since Greedy1 has to calculate the marginal gain of each non-seed at each iteration while the heuristic-based algorithms simply choose the non-seed with the best heuristic value (e.g., out-degree and centrality). We also find that the alternative implementation of Greedy, i.e., Greedy2, shows its advantage in terms of efficiency. Greedy2 is faster than Greedy1 because the total cost of sampling in Greedy2 is much smaller than that in Greedy1. Surprisingly, we find that Greedy2 is even faster than Random though the cost of choosing a seed in Random is $O(1)$. This is possible since Random usually has to select more seeds than Greedy2 in order to incur at least the same amount of influence and for each iteration, Random also needs to calculate the influence incurred by the current seed set which performs the Monte-Carlo simulation many times.

For the LT model, we show the experimental results in Fig. 7. Again, Greedy1 requires the most running time. However, different from the results for the IC model, Greedy2's efficiency is similar to that of other heuristic algorithms.

*Memory*. Same as the experiments for effectiveness and efficiency testing, we vary $J$ and the experimental results are shown in Fig. 8 for the IC model and Fig. 9 for the LT model. According to these results, our greedy algorithms share the nice feature of low space complexity with other heuristic algorithms (less than 2 MB for all experiments in this paper).

*Error analysis*. To verify the error bounds derived in this paper, we compare the number of seeds returned by our algorithms with the optimal one on small datasets (0.5% of the HEP-T dataset). We performed Brute-Force search to obtain the optimal solution.

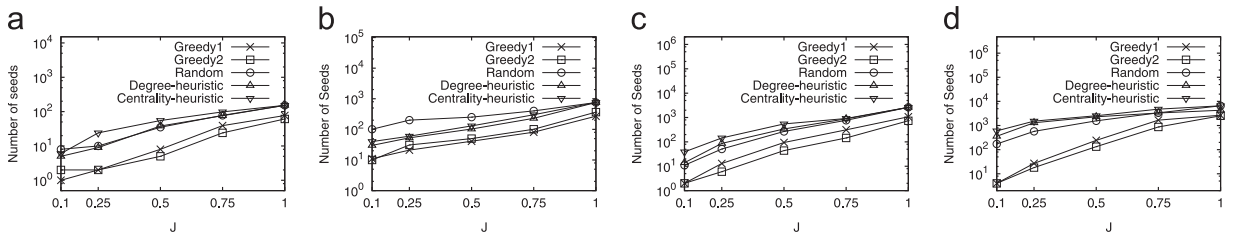



**Fig. 4.** No. of seeds ($J$-MIN-Seed, IC Model). (a) HEP-T, (b) Epinions, (c) Amazon and (d) DBLP.

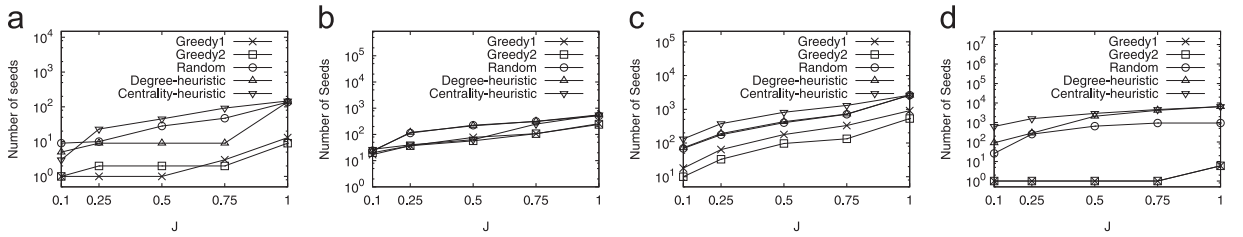



**Fig. 5.** No. of seeds ($J$-MIN-Seed, LT Model). (a) HEP-T, (b) Epinions, (c) Amazon and (d) DBLP.

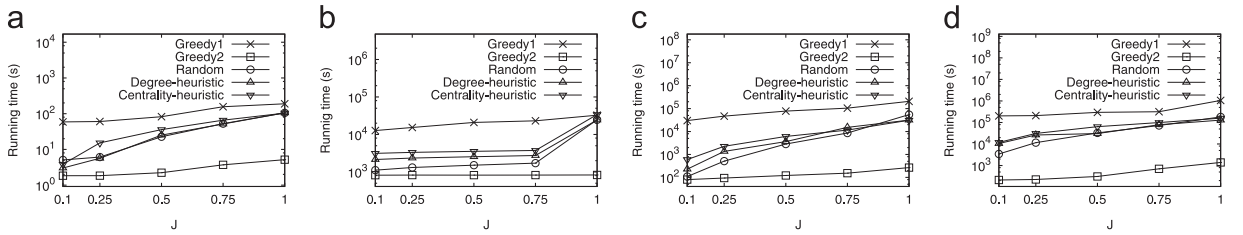



**Fig. 6.** Run time Time ($J$-MIN-Seed, IC Model). (a) HEP-T, (b) Epinions, (c) Amazon and (d) DBLP.

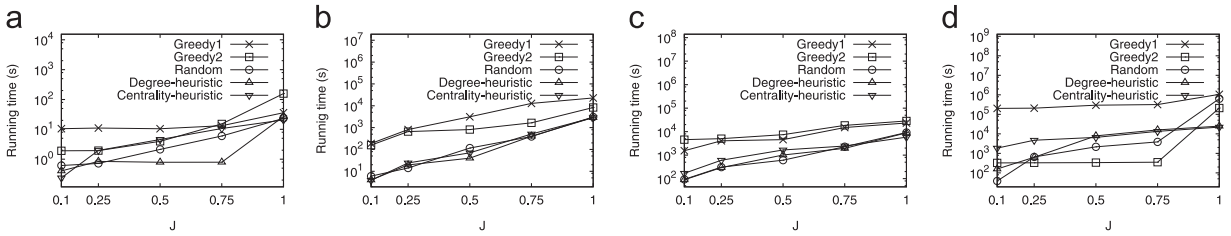



**Fig. 7.** Run time Time (*J*-MIN-Seed, LT Model). (a) HEP-T, (b) Epinions, (c) Amazon and (d) DBLP.

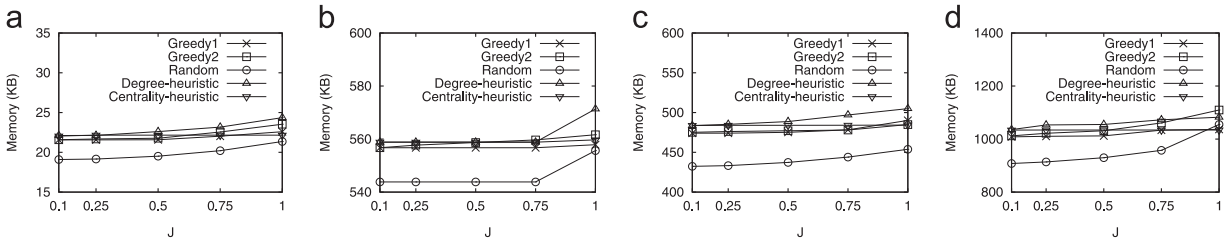



**Fig. 8.** Memory (*J*-MIN-Seed, IC Model). (a) HEP-T, (b) Epinions, (c) Amazon and (d) DBLP.

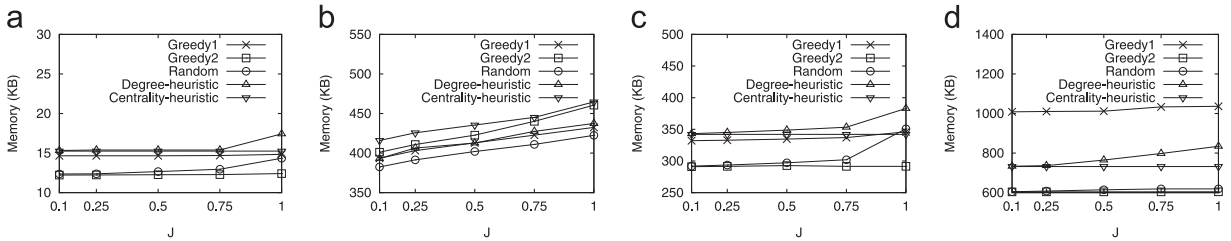



**Fig. 9.** Memory (*J*-MIN-Seed, LT Model). (a) HEP-T, (b) Epinions, (c) Amazon and (d) DBLP.

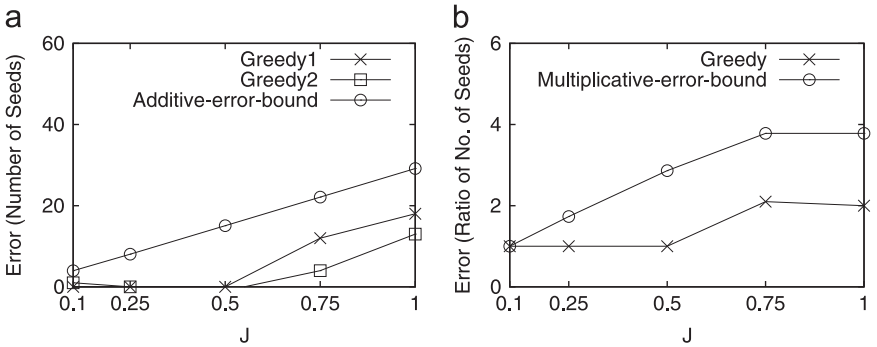



**Fig. 10.** Error analysis (*J*-MIN-Seed, IC Model). (a) Additive and (b) multiplicative.

For the IC model, the experimental results are shown in Fig. 10. According to these results, the additive errors incurred by our algorithms are generally much smaller than the theoretical error bounds on the real dataset. In Fig. 10(b), we find that the multiplicative error of our greedy algorithm grows slowly when $J$ increases. Besides, we discover that $k_2$ is the smallest among $k_1$, $k_2$ and $k_3$ in most cases of our experiments. That is, the multiplicative bound becomes $(1+k_2)$ (i.e., $(1+\ln \sigma'(S_1)/(\sigma'(S_h)-\sigma'(S_{h-1}))$) in these cases. Based on this, we can explain the phenomenon in Fig. 10(b) that the theoretical multiplicative error bound does not change too much when we increase $J$ from 0.75 to 1.

For the LT model, the results are shown in Fig. 11. According to these results, the additive errors of our greedy algorithms are much smaller than the theoretical error bounds. For the multiplicative errors shown in Fig. 11 (b), we find that the theoretical bounds are usually 1, i.e., the approximate solution should be exactly the optimal one, which is verified by our results.

### 7.2.2. *Interest-Specified J-MIN-Seed*

We used the same measurements for Interest-Specified *J*-MIN-Seed as those for *J*-MIN-Seed. The experimental experiments are shown as follows.

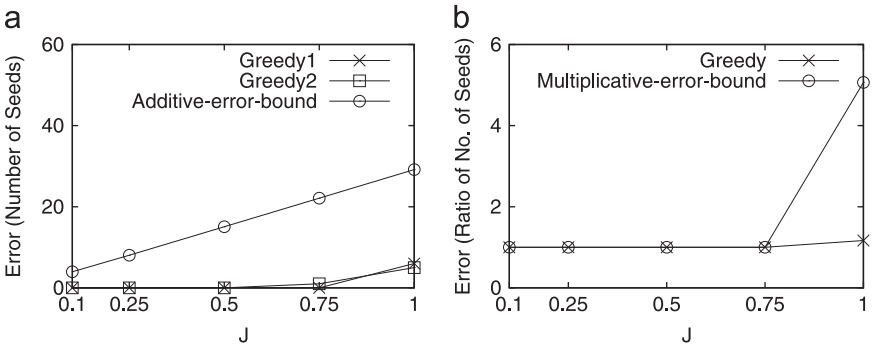



**Fig. 11.** Error analysis (*J*-MIN-Seed, LT Model). (a) Additive and (b) multiplicative.

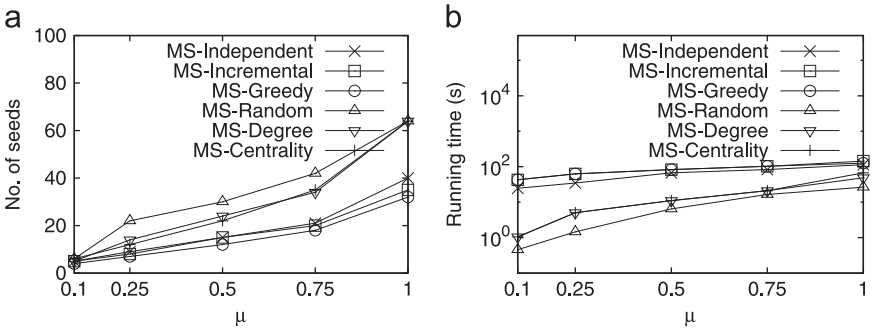



**Fig. 12.** IS-*J*-MIN-Seed (on Twitter, IC model). (a) No. of seeds and (b) Run time.

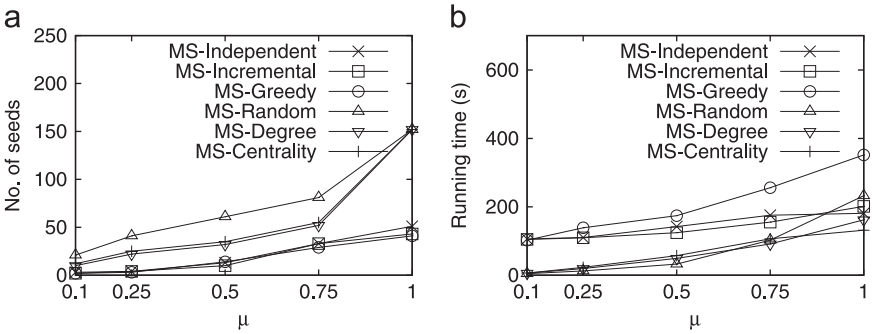



**Fig. 13.** IS-*J*-MIN-Seed (on HEP-T, IC model). (a) No. of seeds and (b) Run time.

*No. of seeds*. We vary the parameter $\mu$ and record the number of seeds returned for each setting. For the IC model, the results are presented in Fig. 12(a) (on Twitter) and in Fig. 13(a) (on HEP-T). According to these results, we find that the three algorithms proposed in this paper, namely MS-Independent, MS-Incremental and MS-Greedy, perform better than the baselines. Besides, among these three algorithms, MS-Greedy outperforms slightly better than the other two. For the LT model, the results are shown in Fig. 14(a) (on Twitter) and in Fig. 15(a) (on HEP-T). Similar results to those for the IC model can be found from these results for the LT model.

*Running time*. For the IC model, the results are presented in Fig. 12(b) (on Twitter) and in Fig. 13(b) (on HEP-T). According to these results, we find that MS-Independent, MS-Incremental and MS-Greedy usually run slower than the baselines. This is reasonable since our proposed algorithms need more computation for selecting the seeds than the baselines. For the LT model, the results are similar and are presented in Fig. 14(b) (on Twitter) and in Fig. 15(b) (on HEP-T).

*Memory*. Similar to the case of Interest-Specified *k*-MAX-Influence, all algorithms considered in our experiments have high space efficiency.

*Approximation error*. As discussed in Section 5, our MS-Independent and MS-Greedy algorithms can provide a certain degree of approximation error guarantees. Thus, in this part, we conducted the experiments on the approximation errors incurred by MS-Independent and MS-Greedy. We used a Brute-Force method to compute the

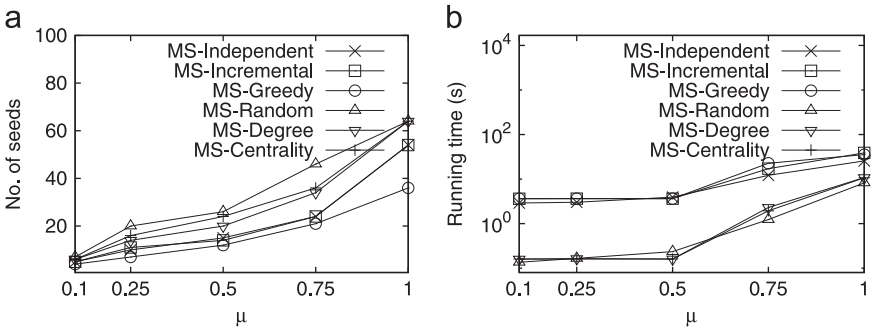



**Fig. 14.** IS-*J*-MIN-Seed (on Twitter, LT model). (a) No. of seeds and (b) Run time.

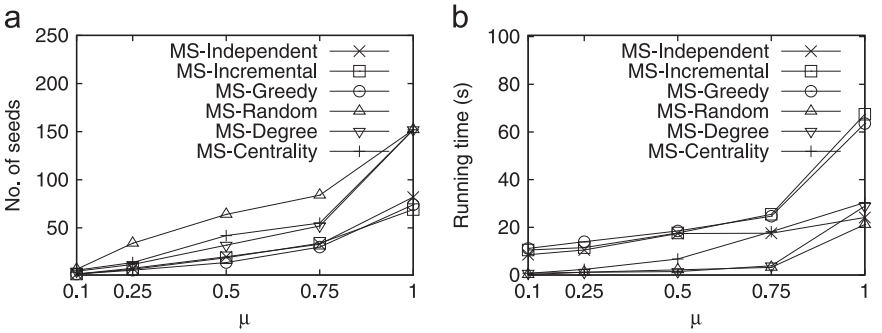



**Fig. 15.** IS-*J*-MIN-Seed (on HEP-T, LT model). (a) No. of seeds and (b) Run time.

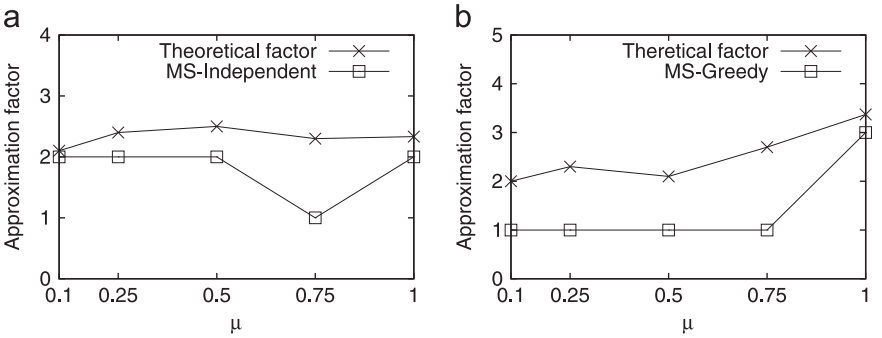



**Fig. 16.** Approximation error (IS-*J*-MIN-Seed, IC model). (a) MS-Independent and (b) MS-Greedy.

optimal solution and then compare this optimal solution with the approximate solutions found by our approximate algorithms.

For the IC model, the experimental results for MS-Independent and MS-Greedy are shown in Fig. 16(a) and in Fig. 16(b), respectively. According to these results, the approximation factors of both MS-Independent and MS-Greedy are around 2–3 (smaller than 4 in all experiments). Note that the approximation factor of MS-Independent would become larger when $m$ increases.

For the LT model, the results for MS-Independent and MS-Greedy are shown in Fig. 17(a) and in Fig. 17(b), respectively. We can observe similar results as those for the IC model.

### 7.2.3. *Interest-Specified k-MAX-Influence*

We used three measurements, namely *influence spread* (the number of influenced users that are of interest), running time and memory. The influence spread measurement measures the effectiveness of the algorithms for IS-*k*-MAX-Influence. The larger the influence spread incurred by the seed set returned by an algorithm is, the better the algorithm is. The experimental results are shown as follows.

*Influence spread*. We vary the parameter $k$ and for each setting of $k$, we record the influence spread incurred by the seed set found by the algorithm. For the IC model, the results are shown in Fig. 18(a) (on Twitter) and in Fig. 19(a) (on HEP-T). According to the results, we find that MI-Greedy performs the best because its constructed seed set incurs the greatest influence spread. For the LT model, the results are shown in Fig. 20(a) (on Twitter) and in Fig. 21 (a) (on HEP-T). Similar patterns as those for the IC model can be found in these results for the LT model.

*Running time*. For the IC model, the results are shown in Fig. 18(b) (on Twitter) and in Fig. 19(b) (on HEP-T).

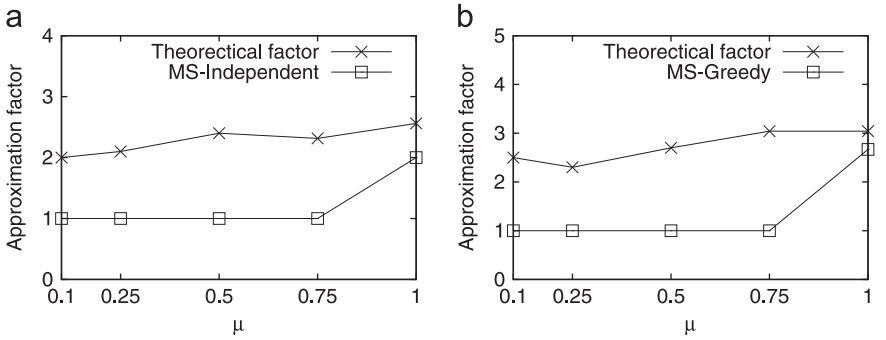



**Fig. 17.** Approximation error (IS-*J*-MIN-Seed, LT model). (a) MS-Independent and (b) MS-Greedy.

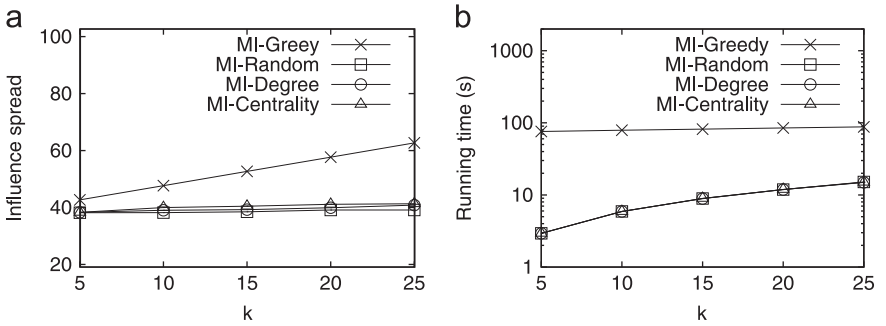



**Fig. 18.** IS-*k*-MAX-Influence (on Twitter, IC model). (a) Influence spread and (b) Run time.

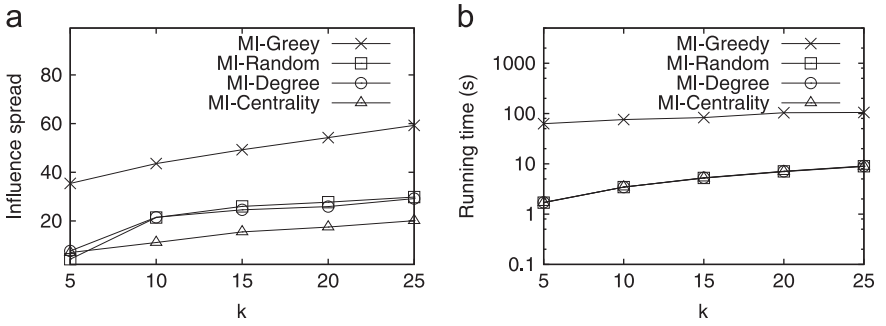



**Fig. 19.** IS-*k*-MAX-Influence (on HEP-T, IC model). (a) Influence spread and (b) Run time.

These results show that MI-Greedy is slower than the baselines. This can be explained by the fact that MI-Greedy needs more computation for selecting a seed than the baselines. For the LT model, the results are similar and are shown in Fig. 20 (b) (on Twitter) and in Fig. 21(b) (on HEP-T).

*Memory*. We vary $k$. We find that all algorithms are space-efficient for both the IC model and the LT model. For example, in all of our experiments, the memory occupied by the algorithms isis no more than $2M$. This is due to the fact that the main memory usage of these algorithms is to store the social network only.

*Approximation error*. To verify the approximation factor (i.e., $(1-1/e)\approx 0.63$) of MI-Greedy, we used a brute-force method to find the seed set that incurs the maximum influence spread of interest on a sampled social network of Twitter (with the sampling rate of 5%). Then, we ran our MI-Greedy algorithm on the same dataset and obtain the approximation solution. After that, we collected the approximation error of MI-Greedy by comparing the optimal influence spread and that incurred by the seed set returned by MI-Greedy.

For the IC model, the results are shown in Fig. 22. According to these results, the influence spread incurred by the seed set returned by MI-Greedy is very close to the optimal one. For the LT model, the results are shown in Fig. 23 and similar results can be found.

### 7.2.4. Experiment conclusion

For the *J*-MIN-Seed problem, our Greedy algorithm beats all the baselines in terms of effectiveness. In addition, the approximation error of our Greedy algorithm is usually much smaller than the theoretical bounds.

For the IS-*J*-MIN-Seed problem, our proposed approximate algorithms, i.e., MS-Independent, MS-Incremental and MS-Greedy, are more effective than the baselines. Besides, the theoretical (multiplicative) approximation

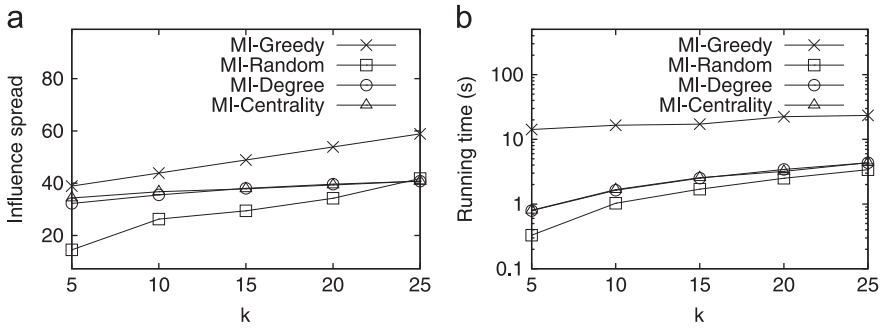



**Fig. 20.** IS-*k*-MAX-Influence (on Twitter, LT model). (a) Influence spread and (b) Run time.

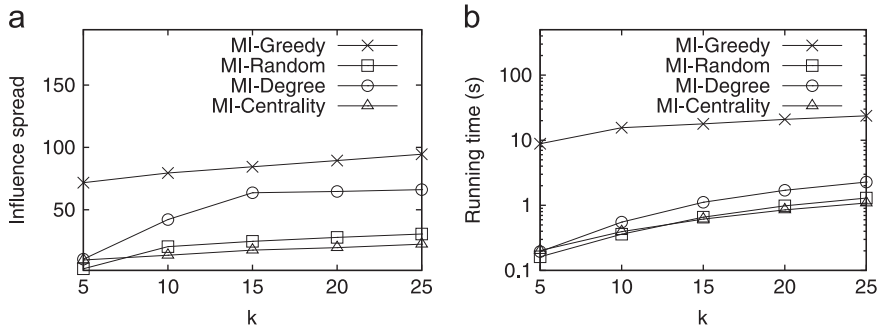



**Fig. 21.** IS-*k*-MAX-Influence (on HEP-T, LT model). (a) Influence spread and (b) Run time.

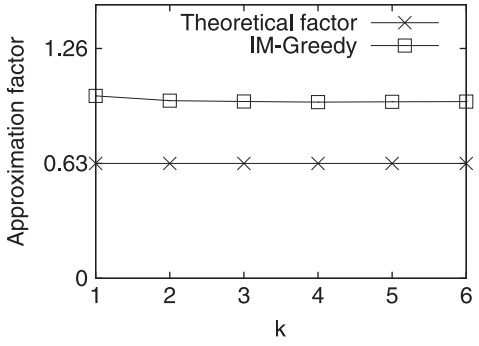



**Fig. 22.** Approximation error (IS-*k*-MAX-Influence, IC model).

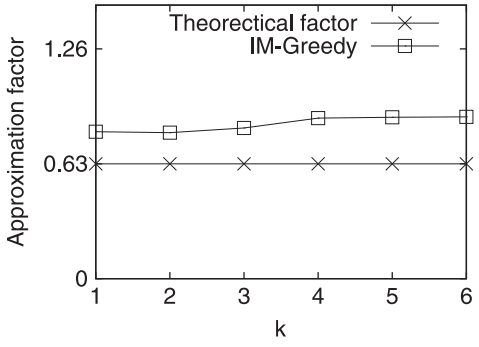



**Fig. 23.** Approximation error (IS-*k*-MAX-Influence, LT model).

error bounds of our algorithms are around 2–3 and in practice, the (multiplicative) approximation errors of our algorithms are smaller than 2 in most cases.

For the IS-*k*-MAX-Influence problem, our MI-Greedy algorithm returns the seed set that incurs the greatest influence spread. Besides, there is usually a gap between the theoretical approximation error bound (i.e., 0.63) and the practical approximation error (e.g., 0.8 on average).

## 8. Conclusion

In this paper, we propose a new viral marketing problem called *J*-MIN-Seed, which has extensive applications in real world. We then prove that *J*-MIN-Seed is NP-hard under two popular diffusion models (i.e., the IC model and the LT model). To solve *J*-MIN-Seed effectively, we develop a greedy algorithm, which can provide approximation guarantees. Besides, we propose a new paradigm of viral marketing called Interest-Specified Viral Marketing, where the companies can specify which kinds of users are of interest. Under this paradigm, we propose two new problems, namely IS-*J*-MIN-Seed and IS-*k*-MAX-Influence, which are the counterparts of *J*-MIN-Seed and *k*-MAX-Influence, respectively. These two problems are more general than their counterparts which are NP-hard and thus they are NP-hard as well. Then, for each of the two proposed problems, we design approximate algorithms that could provide approximation error guarantees. Finally, we conducted extensive experiments on real datasets, which verified the effectiveness of our algorithms.

There are several interesting research directions related to our work. First, in this paper, we assume that we have the complete access to the whole social network which might not be true in some cases (e.g., due to security

problems or economic issues). Thus, how to effectively carry out the viral campaigns in these cases still remain un-solved. Second, it is interesting to capture other factors in addition to the product's target customers, e.g., the users' spatial information and the community structure within the social network, in order to further improve the effectiveness of the viral marketing campaign. Third, it is worth mentioning that the direction of extracting a sub-net from the original social network which involves only those users who are of interest for a specific product has not been unexplored yet. The key of this direction is how to set the weights of the edges in the sub-net (if exists), which turns out to be non-trivial.

## Acknowledgements

We are grateful to the anonymous reviewers for their constructive comments on this paper. The research is supported by grant FSGRF13EG27.

## Appendix A. Proof of lemmas/theorems

**Proof of Property 1.** The proof can be found in [12]. □

**Proof of Property 2.** We prove Property 2 by constructing a problem instance where $\alpha(\cdot)$ does not satisfy the conditions of a submodular function. We first discuss the case for the IC model. Consider the example as shown in Fig. 2. In this figure, there are four nodes, namely Ada, Bob, Connie and David. We assume that each edge is associated with its weight equal to 1, which indicates that an influenced node $u$ will influence a non-influenced node $v$ *definitely* when there is an edge from $u$ to $v$. Let set $T$ be {Ada, Connie, David} and a subset of $T$, says $S$, be {Connie, David}. Obviously, when Ada is influenced, it will further influence Connie and David, i.e., all the nodes in $T$ will be influenced when Ada is selected as a seed. Thus, $\alpha(T)=1$. Similarly, we know that $\alpha(S)=1$. Now, we add Bob into both $T$ and $S$ and then obtain $\alpha(T\cup\{\text{ Bob }\})=2$ (by the seed set {Ada, Bob}) and $\alpha(S\cup\{\text{ Bob }\})=1$ (by the seed set {Bob}). As a result, we know that $\alpha(T\cup\{\text{ Bob }\})-\alpha(T)=1>\alpha(S\cup\{\text{ Bob }\})-\alpha(S)=0$, which, however, violates the conditions of a submodular function.

Next, we discuss the case for the LT model. Consider the special case where each node's threshold is equal to a value slightly greater than 0. Consequently, a node will be influenced whenever one of its neighbors becomes influenced. The resulting diffusion process is actually identical to the special case for the IC model where the weights of all edges are 1 s. That is, the example in Fig. 2 can also be applied for the LT model. Hence, Property 2 also holds for the LT model. □

**Proof of Theorem 1.** First, we give the *J*-MIN-Seed's decision problem as follows. Given a social network $G(V,E)$ and two integers $J$ and $l$, we want to find a set $S$ of seeds such that $|S|\le l$ and $\sigma(S)\ge J$. It could be noted that this decision problem is identical to that of the *k*-MAX-Influence problem. Therefore, the procedure of proving *k*-MAX-Influence's NP-hardness in [12] can carry over to proving the NP-hardness of the *J*-MIN-Seed problem. Interested readers are referred to [12] for details. □

**Proof of Lemma 1.** Assume that we perform the simulation process $n$ times. Let $I_i$ be the influence incurred by the seed set $S$ during the $i$th simulation. Let $E(I)$ be the expected value of $I_i$ and $\bar{I}=\sum_{i=1}^{n} I_i$ be the mean of the $I_i$ values. According to *Hoeffding's Inequality*, for any non-negative real number $t$, we know

$$\Pr\left(|\bar{I}-E(I)|\ge t\right)\le 2\exp\left(-\frac{2t^2n^2}{\sum_{i=1}^{n}(u_i-l_i)^2}\right)$$

where $u_i$ and $l_i$ are the upper bound and the lower bound of $I_i$, respectively.

Considering $I_i\le|V|$ and $I_i\ge 1$, i.e., $u_i=|V|$ and $l_i=1$, for $1\le i\le n$, we have

$$\Pr\left(|1-\frac{\bar{I}}{E(I)}|\ge\frac{t}{E(I)}\right)\le 2\exp\left(-\frac{2t^2n}{(|V|-1)^2}\right)$$

Let $\epsilon=t/E(I)$. We obtain

$$\Pr\left(|1-\frac{\bar{I}}{E(I)}|\ge\epsilon\right)\le 2\exp\left(-\frac{2\epsilon^2E(I)^2n}{(|V|-1)^2}\right)$$

Hence, in order to obtain a $(1\pm\epsilon)$-approximation algorithm with the confidence at least $c$, the following condition should hold:

$$2\exp\left(-\frac{2\epsilon^2E(I)^2n}{(|V|-1)^2}\right)\le 1-c$$

As a result, we obtain the requirement on the number of simulations as follows:

$$n\ge\frac{(|V|-1)^2\ln\left(\frac{2}{1-c}\right)}{2\epsilon^2E(I)^2}$$

Since the seeds themselves would be influenced, we know that $E(I)\ge|S|$. As a result, we obtain the following inequality:

$$n\ge\frac{(|V|-1)^2\ln\left(\frac{2}{1-c}\right)}{2\epsilon^2|S|^2}$$

Thus, we finish our proof. □

**Proof of Lemma 2.** Firstly, we give the theoretical bound on the influence for *k*-MAX-Influence. The problem of determining the *k*-element set $S\subset V$ that maximizes the value of $\sigma(\cdot)$ is NP-hard. Fortunately, according to [52], a simple greedy algorithm can solve this maximization problem with the approximation factor of $(1-1/e)$ by initializing an empty set $S$ and iteratively adding the node such that the marginal gain of inserting this node into the current set $S$ is the greatest one until $k$ nodes have been added. We present this interesting tractability property of maximizing a submodular function in Property 1 as follows.

**Lemma 7** (*Nemhauser et al. [52]*). *For a non-negative, monotone submodular function f, we obtain a set S of size k by initializing set S to be an empty set and then iteratively adding the node u one at a time such that the marginal gain*

*of inserting u into the current set S is the greatest. Assume that S* is the set with k elements that maximizes function f, i.e., the optimal k-element set. Then, $f(S) \geq (1-1/e)\cdot f(S^*)$, where e is the natural logarithmic base.* □

Secondly, we derive the additive error bound on the seed set size for J-MIN-Seed based on the aforementioned bound.

As discussed in Section 3, $\sigma(\cdot)$ is submodular. Clearly, $\sigma(\cdot)$ is also non-negative and monotone. The framework in Algorithm 1 involves a number of iterations (lines 2–4) where the size of the seed set $S$ is incremented by one for each iteration. We say that the framework in Algorithm 1 is at stage $j$ if the seed set $S$ contains $j$ seeds at the end of an iteration. The seed set $S$ at stage $j$ is denoted by $S_j$. Consequently, according to Lemma 7, at each stage $j$, we conclude that

$$\sigma(S_j) \geq (1-1/e)\cdot\sigma(S_j^*) \tag{A.1}$$

where $S_j^*$ is the set that provides the maximum value of $\sigma(\cdot)$ over all possible seed sets of size $j$. Note that the total number of stages for the greedy process is equal to $h$ (i.e., the size of the seed set returned by the algorithm). That is, the greedy process stops at stage $h$. Thus, we know that $\sigma(S_h) \geq J$ and the greedy solution for J-MIN-Seed is $S_h$. Consider the last two stages, namely stage $h-1$ and stage $h$. We know that $\sigma(S_{h-1}) < J$ and $\sigma(S_h) \geq J$. Since $\sigma(S_h^*) \geq \sigma(S_h)$, we have $\sigma(S_h^*) \geq J$.

Now, we want to explore the relationship between $h$ and $t$. Note that the following inequality holds:

$$t \leq h \tag{A.2}$$

Consider two stages, stage $i$ and stage $i+1$, such that $\sigma(S_i) < (1-1/e)\cdot J$ while $\sigma(S_{i+1}) \geq (1-1/e)\cdot J$. According to Inequality (A.1), we know $\sigma(S_i^*) < J$. (This is because if $\sigma(S_i^*) \geq J$, then we have $\sigma(S_i) \geq (1-1/e)\cdot J$ with Inequality (A.1), which contradicts $\sigma(S_i) < (1-1/e)\cdot J$.) As a result, we have the following inequality:

$$t > i \tag{A.3}$$

due to the monotonicity property of $\sigma(\cdot)$.

According to Inequality (A.2) and Inequality (A.3), we obtain $t \in [i+1, h]$. That is, the additive error of our greedy algorithm (i.e., $h-t$) is bounded by the number of stages between stage $i+1$ and stage $h$. Since $\sigma(S_{i+1}) \geq (1-1/e)\cdot J$ and $\sigma(S_{h-1}) < J$, the difference of the influence incurred between stage $i+1$ and stage $h-1$ is bounded by $J-(1-1/e)\cdot J = 1/e\cdot J$. Since each stage increases at least 1 influenced node (seed itself), it is easy to see that the number of stages between stage $i+1$ and stage $h-1$ is at most $1/e\cdot J$. Consequently, the number of stages between stage $i+1$ and stage $h$ is at most $1/e\cdot J+1$. As a result, $h-t \leq 1/e\cdot J+1$. □

**Proof of Lemma 3.** This proof involves four parts. In the first part, we construct a new problem $P'$ based on the submodular function $\sigma'(\cdot)$ (instead of $\sigma(\cdot)$). In the second part, we show the multiplicative error bound of the greedy algorithm in Algorithm 1 (using $\sigma'(\cdot)$ instead of $\sigma(\cdot)$) for this new problem $P'$. We denote this adapted greedy algorithm by $A'$. For simplicity, we denote the original greedy algorithm in Algorithm 1 using $\sigma(\cdot)$ by $A$. In the third part, we show that this new problem is equivalent to the J-MIN-Seed problem. In the fourth part, we show that the multiplicative error bound deduced in the second part can be used as the multiplicative error bound of algorithm $A$ for J-MIN-Seed.

Firstly, we construct a new problem $P'$ as follows. Note that $\sigma'(S) = \min\{\sigma(S), J\}$. Problem $P'$ is formalized as follows:

$$\arg\min\{|S|: \sigma'(S) = \sigma'(V), S \subseteq V\}. \tag{A.4}$$

Secondly, we show the multiplicative error bound of algorithm $A'$ for problem $P'$ by using the following lemma.

**Lemma 8** (*Wolsey [53]*). *Given problem* $\arg\min\{\sum_{x\in S} g(x): f(S) = f(U), S \subseteq U\}$ *where f is a nondecreasing and submodular function defined on subsets of a finite set U, and g is a function defined on U. Consider the greedy algorithm that selects x in $U-S$ such that $(f(S\cup\{x\})-f(S))/g(x)$ is the greatest and adds it into S at each iteration. The process stops when $f(S) = f(U)$. Assume that the greedy algorithm terminates after h iterations and let $S_i$ denote the seed set at iteration i ($S_0 = \emptyset$). The greedy algorithm provides a $(1+\min\{k_1, k_2, k_3\})$-approximation of the above problem, where $k_1 = \ln(f(U)-f(\emptyset))/(f(U)-f(S_{h-1}))$, $k_2 = \ln(f(S_1)-f(\emptyset))/(f(S_h)-f(S_{h-1}))$, and $k_3 = \ln(\max\{(f(\{x\})-f(\emptyset))/(f(S_i\cup\{x\})-f(S_i)) | x \in U, 0 \leq i \leq h, f(S_i\cup\{x\})-f(S_i) > 0\})$.* □

We apply the above lemma for problem $P'$ as follows. It is easy to verify that $\sigma'(\cdot)$ is a non-decreasing and submodular function defined on subsets of a finite set $V$. We set $U$ to be $V$ and set $f(\cdot)$ to be $\sigma'(\cdot)$. We also define $g(x)$ to be 1 for each $x \in V$ (or $U$). Note that $\sum_{x\in S} g(x) = |S|$. We rewrite Problem $P'$ (A.4) as follows:

$$\arg\min\left\{\sum_{x\in S} g(x): \sigma'(S) = \sigma'(V), S \subseteq V\right\}. \tag{A.5}$$

The above form of problem $P'$ is exactly the form of the problem described in Lemma 8. Suppose that we adopt the greedy algorithm in Algorithm 1 for problem $P'$ by using $\sigma'(\cdot)$ instead of $\sigma(\cdot)$, i.e., algorithm $A'$. It is easy to verify that algorithm $A'$ follows the steps of the greedy algorithm described in Lemma 8 (i.e., selecting the node $x$ such that $(\sigma'(S\cup\{x\})-\sigma'(S))/g(x)$ is the greatest where $g(x)$ is exactly equal to 1). By Lemma 8, the greedy algorithm $A'$ for problem $P'$ gives $(1+\min\{k_1, k_2, k_3\})$-approximation of problem $P'$, where

$$k_1 = \ln\frac{\sigma'(V)-\sigma'(\emptyset)}{\sigma'(V)-\sigma'(S_{h-1})} = \ln\frac{J}{J-\sigma'(S_{h-1})},$$

$$k_2 = \ln\frac{\sigma'(S_1)-\sigma'(\emptyset)}{\sigma'(S_h)-\sigma'(S_{h-1})} = \ln\frac{\sigma'(S_1)}{\sigma'(S_h)-\sigma'(S_{h-1})},$$

and

$$k_3 = \ln\left(\max\left\{\frac{\sigma'(\{x\})}{\sigma'(S_i\cup\{x\})-\sigma'(S_i)} | x \in V, \; 0 \leq i \leq h, \sigma'(S_i\cup\{x\})-\sigma'(S_i) > 0\right\}\right).$$

Thirdly, we show that problem $P'$ is equivalent to the J-MIN-Seed problem which can be formalized as follows (since $\sum_{x\in S} g(x) = |S|$):

$$\arg\min\left\{\sum_{x\in S} g(x): \sigma(S) \geq J, S \subseteq V\right\}. \tag{A.6}$$

In the following, we show that the set of all possible solutions for the problem in the form of (A.6) (i.e., the $J$-MIN-Seed problem) is equivalent to the set of all possible solutions for the problem in the form of (A.5) (i.e., problem $P'$). Note that the objective functions in both problems are equal. The remaining issue is to show that the constraints for one problem are the same as those for the other problem.

Suppose that $S$ is a solution for the problem in the form of (A.6). We know that $\sigma(S) \geq J$ and $S \subseteq V$. We derive that $\sigma'(S) = J$. Since $\sigma'(V) = J$, we have $\sigma'(S) = \sigma'(V)$ and $S \subseteq V$ (which are the constraints for the problem in the form of (A.5)).

Suppose that $S$ is a solution for the problem in the form of (A.5). We know that $\sigma'(S) = \sigma'(V)$ and $S \subseteq V$. Since $\sigma'(V) = J$, we have $\sigma'(S) = J$. Considering $\sigma'(S) = \min\{\sigma(S), J\}$, we derive that $\sigma(S) \geq J$. So, we have $\sigma(S) \geq J$ and $S \subseteq V$ (which are the constraints for the problem in the form of (A.6)).

Fourthly, we show that the size of the solution (i.e., $|S|$) returned by algorithm $A'$ for the new problem $P'$ is equal to that returned by algorithm $A$ for $J$-MIN-Seed. Since $\sigma(S_i) < J$ for $1 \leq i \leq h-1$, we know that $\sigma'(S_i) = \sigma(S_i)$ for $1 \leq i \leq h-1$. We also know that the element $x$ in $V - S_{i-1}$ that maximizes $\sigma(S_{i-1} \cup \{x\}) - \sigma(S_{i-1})$ (which is chosen at iteration $i$ by algorithm $A$) would also be the element that maximizes $\sigma'(S_{i-1} \cup \{x\}) - \sigma'(S_{i-1})$ (which is chosen at iteration $i$ by algorithm $A'$) for $i = 1, 2, \ldots, h-1$. That is, algorithm $A'$ would proceed in the same way as algorithm $A$ at iteration $i = 1, 2, \ldots, h-1$. Consider iteration $h$ of algorithm $A$. We denote the element selected by algorithm $A$ by $x_h$. Then, we know $\sigma(S_{h-1} \cup \{x_h\}) \geq J$ since algorithm $A$ stops at iteration $h$. Consider iteration $h$ of algorithm $A'$. This iteration is also the last iteration of $A'$. This is because there exists an element $x$ in $V - S_{h-1}$ such that $\sigma'(S_{h-1} \cup \{x\}) = \sigma'(V) (= J)$ (since $x$ can be equal to $x_h$ where $\sigma'(S_{h-1} \cup \{x_h\}) = J$). Note that this element $x$ maximizes $\sigma'(S_{h-1} \cup \{x\}) - \sigma'(S_{h-1})$ and thus is selected by $A'$. We conclude that both algorithms $A$ and $A'$ terminate at iteration $h$. Since the number of iterations for an algorithm ($A$ or $A'$) corresponds to the size of the solution returned by the algorithm, we deduce that the size of the solution returned by algorithm $A'$ is equal to that returned by algorithm $A$.

In view of the above discussion, we know that problem $P'$ is equivalent to $J$-MIN-Seed and algorithm $A'$ for problem $P'$ would proceed in the same way as algorithm $A$ for $J$-MIN-Seed. As a result, the multiplicative bound of algorithm $A'$ for problem $P'$ in the second part also applies to algorithm $A$ (i.e., the greedy algorithm in Algorithm 1) for $J$-MIN-Seed. □

**Proof of Property 3.** We first prove Property 3 for the IC model. The proof has two steps.

First, we prove that $\sigma(S, A_I)$ is submodular on a *deterministic* social network, where the weights of all edges are 1. Let $T$ and $S$ be any two subsets of $V$ such that $S \subset T$ and $v$ be any node *not* in $T$. The marginal gain of $v$ when inserted into $T$ corresponds to the number of nodes of interest that can be reached from $v$ but not from any node in $T$. We denote by $G(v, T)$ the set including all these nodes. Similarly, we use $G(v, S)$ to represent the set of nodes of interest that can be reached from $v$ but not from any node in $S$. Consider any node $u \in G(v, T)$. We show that $u \in G(v, S)$ by contradiction. Suppose $u \notin G(v, S)$. Since $u$ can be reached from $v$ ($v \in G(v, T)$), we know that $u$ must be reachable from a node in $S$. Considering $S \subset T$, we further conclude that $u$ must be reachable from a node in $T$ and thus it contradicts the condition that $u \in G(v, T)$. Therefore, $u \in G(v, S)$. Thus, any node $u \in G(v, T)$ is also included in $G(v, S)$. Therefore, we know that $G(v, T) \subset G(v, S)$. It follows that $\sigma(T \cup \{v\}, A_I) - \sigma(T, A_I) \leq \sigma(S \cup \{v\}, A_I) - \sigma(S, A_I)$. Thus, $\sigma(S, A_I)$ is submodular on a deterministic social network.

Second, based on the above results, we proceed to show that $\sigma(S, A_I)$ is submodular on a general (probabilistic) social network $G$. Any edge $(u, v)$ with the weight of $w_{u,v}$ is discretized to own the weight of 1 with the probability of $w_{u,v}$ and the weight of 0 with the probability of $1 - w_{u,v}$. As a result, social network $G$ can be discretized into a set of deterministic social networks $G_x$, each with a probability, denoted by $P_x$. Let $\sigma_x(S, A_I)$ be the number of nodes of interest incurred by $S$ on $G_x$ and thus $\sigma_x(S, A_I)$ is submodular according to the results of the first step. Besides, $\sigma(S, A_I)$ is the expectation of $\sigma_x(S, A_I)$, i.e., $\sigma(S, A_I) = \sum_x P_x \cdot \sigma_x(S, A_I)$. According to [52], the combination of submodular functions is also submodular, we conclude that $\sigma(S, A_I)$ is submodular.

We then prove Property 3 for the LT model. According to [12], the diffusion process on a graph $G(V, E)$ under the LT model is *equivalent* to the traversal process on the same graph containing only the so-called *live* edges. The live edges in $G$ are selected randomly as follows. For each node $v$, at most one edge among all edges that go to $v$ is specified to be live and the probability of selecting edge $(u, v)$ is $w_{u,v}$, where $u$ is an in-neighbor of $v$. The probability that no live edges go to $v$ is thus equal to $1 - \sum_{u \text{ is a in-neighbor of } v} w_{u,v}$. According to [12], the diffusion process on $G$ is equivalent to the traversal process on $G_l$, where $G_l$ is $G$ by excluding all edges that are not live. Specifically, all nodes that are reachable from the seeds in $G_l$ would be influenced. As a result, the diffusion process of the LT model could be discretized as the traversal processes on a *finite* number of $G_l$'s and each of them is associated with a specific probability.

The traversal process on a specific $G_l$ is equivalent to the diffusion process on graph $G_l$ under the IC model, where the weights of all edges in $G_l$ are set to 1 s. Thus, according to the above results for the IC model, $\sigma(S, A_I)$ is submodular for the traversal process on $G_l$. Considering the diffusion process under the LT model is a weighted combination of the traversal processes on all possible $G_l$'s and for each traversal process on a specific $G_l$, $\sigma(S, A_I)$ is submodular, we know that $\sigma(S, A_I)$ is submodular under the LT model. □

**Proof of Lemma 4.** We define a new problem $Q'$ as $\min\{|S_x| : \sigma(S_x, \{a_{i_x}\}) \geq j_x\}$ $(1 \leq x \leq m)$. That is, $Q'$ corresponds to the problem of fining the *smallest* seed set $S_x$ that satisfies the constraint of influencing at least $j_x$ users containing attribute value $a_{i_x}$. Let $S_x^*$ be the optimal solution of $Q'$. Recall that $S_x$ corresponds to the solution returned by a greedy procedure. According to Lemma 3,

we have

$$|S_x|/|S_x^*| \le 1+\min\{t_x^1, t_x^2, t_x^3\} \tag{A.7}$$

where $t_x^1 = \ln j_x/(j_x - \sigma'(S_x^{r_x-1}, \{a_{i_x}\}))$ and $t_x^2 = \ln \sigma'(S_x^1, \{a_{i_x}\})/(\sigma'(S_x^{r_x}, \{a_{i_x}\}) - \sigma'(S_x^{r_x-1}, \{a_{i_x}\}))$ and $t_x^3 = \ln \max\{\sigma'(\{v\}, \{a_{i_x}\})/(\sigma'(S_x^h \cup \{v\}, \{a_{i_x}\}) - \sigma'(S_x^h, \{a_{i_x}\}))|1 \le h \le r_x, v \in V, \sigma'(S_x^h \cup \{v\}, \{a_{i_x}\}) - \sigma'(S_x^h, \{a_{i_x}\}) > 0\}$. Let $B_x$ be $1+\min\{t_x^1, t_x^2, t_x^3\}$. That is, $|S_x|/|S_x^*| \le B_x$.

First of all, we have the following inequality, which could be easily verified by contradiction:

$$|S^*| \ge |S_x^*| \tag{A.8}$$

Second, since $|S_x| = \max_{1 \le l \le m}|S_l|$ and $S = \cup_{1 \le l \le m} S_l$. we deduce the following inequality:

$$|S| \le m \cdot |S_x| \tag{A.9}$$

By using Eq. (A.7), Eqs. (A.8) and (A.9), we have the following result:

$$|S| \le m \cdot |S_x| \le m \cdot B_x \cdot |S_x^*| \le m \cdot B_x \cdot |S^*|$$

That is, $|S|/|S^*| \le m \cdot B_x$. □

**Proof of Lemma 5.** We formalize the IS-*J*-MIN-Seed problem as problem $P$: $\min\{|S|: \sigma(S, \{a_{i_l}\}) \ge j_l \text{ for } 1 \le l \le m\}$. We prove Lemma 5 with three steps. First, a new problem, $P'$ is defined. Second, we prove that MS-Greedy provides an approximation factor $B$ for problem $P'$, where $B = 1+\min\{t^1, t^2, t^3\}$. Third, we prove that this approximation factor of $B$ also applies to problem $P$ by showing that problem $P$ is equivalent to problem $P'$.

First, we define problem $P'$ as $\min\{|S|: \sigma'_a(S) \ge J_{sum}\}$ where $J_{sum} = \sum_{1 \le l \le m} j_l$.

Second, according to Property 3, function $\sigma(S, A_I)$ is submodular. Thus, we know that function $\sigma(S, \{a_{i_l}\})$ $(1 \le l \le m)$ is submodular. Besides, it is known that, for a submodular function $f: 2^U \to \mathcal{R}$ and a real number $\theta$, function $\min\{f(S), \theta\}$ is also submodular [53]. Thus, we know that $\sigma'(S, \{a_{i_l}\})$ is a submodular function. Furthermore, it is easy to verify that the summation of several submodular functions is submodular and thus we know that $\sigma'_a(S)$ is a submodular function as well. As can be noted in Algorithm 4, MS-Greedy is exactly a greedy procedure based on $\sigma'_a(S)$. According to Lemma 3, we know MS-Greedy provides the approximation factor of $1+\min\{t^1, t^2, t^3\}$ for problem $P'$, where $t^1 = \ln J_{sum}/(J_{sum} - \sigma'_a(S_{r-1}))$, $t^2 = \ln \sigma'_a(S_1)/(\sigma'_a(S_r) - \sigma'_a(S_{r-1}))$ and $t^3 = \ln \max\{\sigma'_a(v)/(\sigma'_a(S_h \cup \{v\}) - \sigma'_a(S_h))|1 \le h \le r, v \in V, \sigma'_a(S_h \cup \{v\}) - \sigma'_a(S_h) > 0\}$.

Third, we prove that problem $P$ is equivalent to problem $P'$ as follows. Assume that $S$ is a feasible solution of problem $P$, that is, $\sigma(S, \{a_{i_l}\}) \ge j_l$ for $1 \le l \le m$. It follows that $\sigma'_a(S, \{a_{i_l}\}) = j_l$ for $1 \le l \le m$. As a result, we know $\sigma'_a(S) = \sum_{1 \le l \le m} j_l$ and thus $\sigma'_a(S) \ge \sum_{1 \le l \le m} j_l = J_{sum}$ . That is, $S$ is also a feasible solution of problem $P'$.

Similarly, assume that $S$ is a feasible solution of problem $P'$. That is, $\sigma'_a(S) \ge \sum_{1 \le l \le m} j_l$. Since $\sigma'_a(S, \{a_{i_l}\}) \le j_l$ for $1 \le l \le m$, we know $\sigma'_a(S, \{a_{i_l}\})$ must be $j_l$ for $1 \le l \le m$. It follows that $\sigma(S, \{a_{i_l}\})$ is at least $j_l$ for $1 \le l \le m$ according to the definition of $\sigma'_a(S, \{a_{i_l}\})$. That is, $S$ is a feasible solution of problem $P$ as well.

In summary, we know that MS-Greedy provide the approximation factor of $1+\min\{t^1, t^2, t^3\}$ for problem $P$. □

**Proof of Lemma 6.** Since $\sigma(S, A_I)$ is submodular, according to Lemma 7, we know that MI-Greedy provides a $(1-1/e)$-factor approximation for the IS-*k*-MAX-Influence problem.

## Appendix B. Supplementary data

Supplementary data associated with this article can be found in the online version at http://dx.doi.org/10.1016/j.is.2014.05.003.

## References


[1] J. Bryant, D. Miron, Theory and research in mass communication, J. Commun. 54 (4) (2004) 662–704.
[2] J. Nail, The Consumer Advertising Backlash, Forrester Research.
[3] I.R. Misner, The World's Best Known Marketing Secret: Building your Business with Word-of-Mouth Marketing, 2nd ed. Bard Press, 1999.
[4] A. Johnson, nike-tops-list-of-most-viral-brands-on-facebook-twitter. URL ⟨http://www.kikabink.com/news/⟩, 2010.
[5] M. Granovetter, Threshold models of collective behavior, Am. J. Sociol. 83 (6) (1978) 1420–1443.
[6] T.C. Schelling, Micromotives and Macrobehavior, WW Norton and Company, 2006.
[7] J. Goldenberg, B. Libai, E. Muller, Talk of the network: a complex systems look at the underlying process of word-of-mouth, Mark. Lett. 12 (3) (2001) 211–223.
[8] J. Goldenberg, B. Libai, E. Muller, Using complex systems analysis to advance marketing theory development: modeling heterogeneity effects on new product growth through stochastic cellular automata, Acad. Mark. Sci. Rev. 9 (3) (2001) 1–18.
[9] D. Gruhl, R. Guha, D. Liben-Nowell, A. Tomkins, Information diffusion through blogspace, in: WWW, 2004.
[10] H. Ma, H. Yang, M. R. Lyu, I. King, Mining social networks using heat diffusion processes for marketing candidates selection, in: CIKM, 2008.
[11] P. Domingos, M. Richardson, Mining the network value of customers, in: KDD, 2001.
[12] D. Kempe, J. Kleinberg, E. Tardos, Maximizing the spread of influence through a social network, in: SIGKDD, 2003.
[13] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, N. Glance, Cost-effective outbreak detection in networks, in: SIGKDD, 2007, pp. 420–429.
[14] M. Kimura, K. Saito, Tractable models for information diffusion in social networks, in: PKDD, 2006, pp. 259–271.
[15] W. Chen, C. Wang, Y. Wang, Scalable influence maximization for prevalent viral marketing in large-scale social networks, in: SIGKDD, 2010.
[16] S. Datta, A. Majumder, N. Shrivastava, Viral marketing for multiple products, in: ICDM, 2010.
[17] N. Chen, On the approximability of influence in social networks, in: Proceedings of the 19th Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '08, 2008, pp. 1029–1037.
[18] O. Ben-Zwi, D. Hermelin, D. Lokshtanov, I. Newman, An exact almost optimal algorithm for target set selection in social networks, in: Proceedings of the 10th ACM Conference on Electronic Commerce, ACM, 2009, pp. 355–362.
[19] C. Long, R.C.W. Wong, Minimizing seed set for viral marketing, in: 2011 IEEE 11th International Conference on Data Mining (ICDM), IEEE, 2011, pp. 427–436.
[20] B. Ryan, N.C. Gross, The diffusion of hybrid seed corn in two IOWA communities, Rural Sociol. 8 (1) (1943) 15–24.
[21] M. Richardson, P. Domingos, Mining knowledge-sharing sites for viral marketing, in: SIGKDD, 2002.
[22] W. Chen, Y. Wang, S. Yang, Efficient influence maximization in social networks, in: SIGKDD, 2009, pp. 199–208.
[23] W. Chen, Y. Yuan, L. Zhang, Scalable influence maximization in social networks under the linear threshold model, in: ICDM, IEEE, 2010, pp. 88–97.
[24] Y. Wang, G. Cong, G. Song, K. Xie, Community-based greedy algorithm for mining top-k influential nodes in mobile social networks, in: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2010, pp. 1039–1048.

[25] R. Narayanam, Y. Narahari, A Shapley value-based approach to discover influential nodes in social networks, IEEE Trans. Autom. Sci. Eng. (99) (2010) 1–18.
[26] A. Goyal, W. Lu, L.V.S. Lakshmanan, SIMPATH: an efficient algorithm for influence maximization under the linear threshold model, in: 2011 IEEE 11th International Conference on Data Mining (ICDM), IEEE, 2011.
[27] Q. Jiang, G. Song, G. Cong, Y. Wang, W. Si, K. Xie, Simulated annealing based influence maximization in social networks, in: Proceedings of the 25th AAAI International Conference on Artificial Intelligence, 2011, pp. 127–132.
[28] A. Goyal, F. Bonchi, L.V. Lakshmanan, A data-based approach to social influence maximization, in: Proceedings of the VLDB Endowment, vol. 5 (1), 2011, pp. 73–84.
[29] K. Jung, W. Heo, W. Chen, IRIE: scalable and robust influence maximization in social networks, in: 2012 IEEE 12th International Conference on Data Mining (ICDM), IEEE, 2012, pp. 918–923.
[30] Y. Li, W. Chen, Y. Wang, Z.-L. Zhang, Influence diffusion dynamics and influence maximization in social networks with friend and foe relationships, in: Proceedings of the Sixth ACM International Conference on Web Search and Data Mining, ACM, 2013, pp. 657–666.
[31] C. Borgs, M. Brautbar, J. Chayes, B. Lucier, Influence maximization in social networks: towards an optimal algorithmic solution, arXiv preprint arxiv:1212.0884.
[32] S. Bharathi, D. Kempe, M. Salek, Competitive influence maximization in social networks, in: Internet and Network Economics, 2007, pp. 306–311.
[33] T. Carnes, C. Nagarajan, S.M. Wild, A. van Zuylen, Maximizing influence in a competitive social network: a follower's perspective, in: Proceedings of the Ninth International Conference on Electronic Commerce, ACM, 2007, pp. 351–360.
[34] J. Kostka, Y.A. Oswald, R. Wattenhofer, Word of mouth: rumor dissemination in social networks, in: Structural Information and Communication Complexity, Springer, 2008, pp. 185–196.
[35] N. Pathak, A. Banerjee, J. Srivastava, A generalized linear threshold model for multiple cascades, in: 2010 IEEE 10th International Conference on Data Mining (ICDM), IEEE, 2010, pp. 965–970.
[36] D. Trpevski, W.K. Tang, L. Kocarev, Model for rumor spreading over networks, Phys. Rev. E 81 (5) (2010) 056102.
[37] A. Borodin, Y. Filmus, J. Oren, Threshold Models for Competitive Influence in Social Networks, Internet and Network Economics, Springer, 2010, pp. 539–550.
[38] C. Budak, D. Agrawal, A.E. Abbadi, Limiting the spread of misinformation in social networks, in: Proceedings of the 20th International Conference on World Wide Web, ACM, 2011, pp. 665–674.
[39] W. Chen, A. Collins, R. Cummings, T. Ke, Z. Liu, D. Rincon, X. Sun, Y. Wang, W. Wei, Y. Yuan, Influence maximization in social networks when negative opinions may emerge and propagate, in: Proceedings of SIAM International Conference on Data Mining, 2011, pp. 379–390.
[40] X. He, G. Song, W. Chen, Q. Jiang, Influence blocking maximization in social networks under the competitive linear threshold model.
[41] O. Ben-Zwi, D. Hermelin, D. Lokshtanov, I. Newman, Treewidth governs the complexity of target set selection, Discret. Optim. 8 (1) (2011) 87–96.
[42] D. Reichman, New bounds for contagious sets, Discret. Math. 312 (10) (2012) 1812–1814.
[43] P. Shakarian, D. Paulo, Large Social Networks can be Targeted for Viral Marketing with Small Seed Sets, arXiv preprint arxiv:1205.4431.
[44] A. Goyal, F. Bonchi, L.V.S. Lakshmanan, S. Venkatasubramanian, On minimizing budget and time in influence propagation over social networks, in: Social Network Analysis and Mining, 2012, pp. 1–14.
[45] M. Kitsak, L.K. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H.E. Stanley, H.A. Makse, Identification of influential spreaders in complex networks, Nat. Phys. 6 (11) (2010) 888–893.
[46] D. Centola, An experimental study of homophily in the adoption of health behavior, Science 334 (6060) (2011) 1269–1272.
[47] S. Aral, D. Walker, Identifying influential and susceptible members of social networks, Science 337 (6092) (2012) 337–341.
[48] M. Broecheler, P. Shakarian, V. Subrahmanian, A scalable framework for modeling competitive diffusion in social networks, in: 2010 IEEE Second International Conference on Social Computing (SocialCom), vol. 295, 2010.
[49] D. Jain, V. Mahajan, E. Muller, An approach for determining optimal product sampling for the diffusion of a new product, J. Prod. Innov. Manage. 12 (2) (1995) 124–135.
[50] F. Stonedahl, W. Rand, U. Wilensky, Evolving viral marketing strategies, in: GECCO, 2010.
[51] H. Sharara, W. Rand, L. Getoor, Differential adaptive diffusion: understanding diversity and learning whom to trust in viral marketing, in: Proceedings of the AAAI International Conference on Weblogs and Social Media (ICWSM), 2011.
[52] G.L. Nemhauser, L.A. Wolsey, M.L. Fisher, An analysis of approximations for maximizing submodular set functions—I, Math. Progr. 14 (1) (1978) 265–294.
[53] L.A. Wolsey, An analysis of the greedy algorithm for the submodular set covering problem, Combinatoria 2 (4) (1981) 385–393.