Multimedia Tools and Applications (2023) 82:9065–9081
https://doi.org/10.1007/s11042-021-11461-3

1209: RECENT ADVANCES ON SOCIAL MEDIA ANALYTICS AND MULTIMEDIA SYSTEMS: ISSUES AND CHALLENGES

Time-bounded targeted influence spread in online social networks

Lei Yu[1] · Guohui Li[1] · Ling Yuan[1] · Li Zhang[2]

Received: 25 November 2020 / Revised: 5 July 2021 / Accepted: 19 August 2021 /
Published online: 29 June 2022

Abstract
Influence maximization with application to viral marketing aims to find a small set of influencers in a social network to maximize the number of influenced users under a certain propagation model. However, in many actual marketing scenarios, companies are usually concerned about precision marketing before the specified deadline. In this paper, different from most of influence maximization problems, we focus on an issue of time-bounded targeted influence spread, where it asks for finding a seed set to maximize the influence on a specific set of target users within a bounded time in the network. This problem is NP-hard, and its objective function maintains the monotonicity and submodularity. We devise a greedy algorithm with approximate guarantee to effectively solve the problem. To overcome the low calculational efficiency of this algorithm in large networks, we further propose several efficient heuristic algorithms to greatly speed up the seed selection. Extensive experiments over real-world available social networks of different sizes show the effectiveness and efficiency of the proposed algorithms.

Keywords Online social networks · Viral marketing · Influence maximization · Heuristic algorithm

✉ Ling Yuan
cherryyuanling@hust.edu.cn

Lei Yu
LYU91@hust.edu.cn

Guohui Li
guohuili@hust.edu.cn

Li Zhang
lzhang@fiberhome.com

[1] School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China

[2] Wuhan Fiberhome Technical Service Co., Ltd, Wuhan, China

1 Introduction

As an increasingly popular medium, online social networks (e.g., Twitter, Facebook, etc.) play a very important role in the communication between people in reality. Moreover, they have also become one of the most important marketing platforms, which allow information to be widely disseminated around the social relationships during a short period of time [7]. Therefore, the analysis of online social networks has attracted extensive attention in both theory and practice. In this field, one of the most fundamental problems is influence maximization problem [9, 25]. Formally, Kempe et al. [15] are the first to formulate influence maximization as a discrete optimization problem. The problem aims to find at most K seeds in a network to maximize the expected number of activated nodes under a certain propagation model. Furthermore, Kempe et al. propose two basic influence propagation models, i.e., Independent Cascade (IC) model and Linear Threshold (LT) model. In general, the IC model mainly emphasizes the individual interaction and influence among friends, while the LT model focuses on the influence of group behavior to others. Under these two models, Kempe et al. show that the problem is NP-hard. The influence maximization problem is also very useful in some other domains, such as recommendation services [8, 14], rumor control [27, 35], network monitoring [17] and influential twitters selection [2, 33].

In most of the previous research on influence maximization problem, they focus primarily on maximizing the number of influenced users or blocking the influence spread of competitors in the social networks [3, 5, 13, 16, 18, 22, 24, 29, 30, 32]. However, in many real-world marketing campaigns, companies are usually more concerned about precision marketing within a finite time horizon. In other words, before the specified marketing deadline, they try to narrow the scope of product promotion to those potential high-value users, who are very interested in the product and are more likely to purchase it, rather than all users. Moreover, because of clearer promotion goals and more accurate resource allocation, this kind of marketing can avoid some invalid promotions, and greatly reduce many unnecessary expenses. Therefore, it is considered very effective and reasonable in practice.

The above situation is not scarce in the real world. Let us consider several realistic scenarios. In order to market an e-sports match among the public, the marketers tend to target the sale of tickets primarily to young people before the opening date. This is mainly because compared with other groups, the young people pay more attention to the e-sports and are more likely to buy tickets to watch it live. Moreover, the people to be influenced after the match would not bring any revenue. In addition, conference organizers wish to invite some top experts with similar research fields or interests to attend their conference before it starts. It may be a good idea for the organizers to first know some friends of those experts. Then, by them or their friends, the organizers can know those experts finally. In particular, one may argue that if the marketers or organizers have known their target groups, they do not need any marketing, but directly deliver advertising messages or send invitations. However, since there may exist both social and physical distances between them in reality, which means that the marketers or organizers may be viewed as strangers or untrustworthy people by their target groups, these ways no longer work. On the contrary, the target groups are more willing to trust their friends and accept their suggestions actually.

Motivated by these realistic scenarios, we are very interested in exploring a new problem of maximizing the influence on a specific set of target users within a bounded time by nurturing a small number of seeders (i.e., the initial adopters) in a social network. Whereas there is relatively little work that has fully taken into account this problem. Moreover, it

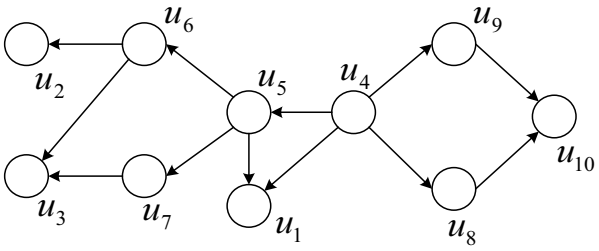

Fig. 1 The example of a directed graph with ten nodes

can see that the most influential nodes in the whole network may not be directly applied to this problem. We take the following example to illustrate such an observation. Consider a directed graph $\mathcal{G}'$ with ten nodes shown in Fig. 1, it assumes that the activation probability on each edge is 0.5, the bounded time is 2 and the set of target nodes contains $\{u_1, u_2, u_3\}$. We can find that when considering the influence spread on the graph $\mathcal{G}'$, node u_4 should be selected as the seed. This is because it achieves the maximal influence spread in $\mathcal{G}'$. However, for the influence spread on those target nodes within the given bounded time, the seed is node u_5, but not node u_4. It demonstrates that the proposed problem is actually very different from the traditional influence maximization problem. Additionally, there are many potentially promising applications for further research on the proposed problem in practice, such as optimizing location or item selection by the targeted preference and behavior analysis, targeted customer service, more accurate advertising and recommendations. In this sense, it is very essential to further investigate the targeted influence spread when time is bounded in social networks.

In this paper, we focus on a more realistic Time-Bounded Targeted Influence Maximization (TB-TIM) problem in social networks, which is a novel variant of influence maximization. This problem asks for identifying a seed set of size at most K in a network to maximize the influence on a specific set of target nodes within a bounded time under the IC model. We show that the problem is NP-hard[1], and its influence spread function maintains the properties of monotonicity and submodularity. To solve the problem, we propose an effective greedy algorithm that can provide a solution with $(1 - e^{-1})$ approximation ratio. However, this algorithm has the high computational cost and low efficiency in large networks. Therefore, instead of the computationally expensive simulation-based method, we further propose two efficient heuristic methods to greatly speed up the seed selection in the network.

To summarize, the main contributions of this paper are as follows.

- We study a more realistic TB-TIM problem. This problem is NP-hard, and computing the influence spread of a seed set on the target nodes within a given bounded time is #P-hard. Moreover, the influence spread function has the desirable monotonicity and submodularity.

[1] NP-hard problem refers to a problem that all NP problems can be reduced to within polynomial time complexity, while NP problem is a problem that can verify a solution in polynomial time. In general, #P-hard problem is more complicated than NP-hard problem. Therefore, in practice, it usually needs to find the approximate solutions for such problems.

- We propose a greedy algorithm with theoretical guarantee to effectively solve the problem. To further improve its calculational efficiency in the seed selection, we propose two efficient heuristic algorithms.
- We evaluate the performance of the proposed methods over several real-world social networks of different sizes and structural features, the experimental results demonstrate that the proposed methods are effective and efficient.

The rest of this paper is structured as follows. Section 2 reviews the related work on influence maximization. In the Sect. 3, we give the definition of the TB-TIM problem. In the Sect. 4, we propose several approximate algorithms. Section 5 presents the experimental results and analysis. Finally, we conclude this paper and discuss several future research directions in the Sect. 6.

2 Related work

To approximately solve the influence maximization problem, a line of algorithms have been actively studied. On the one hand, it focuses on greedy algorithm and its enhancements. Kempe et al. [15] first propose a greedy hill-climbing algorithm. To overcome its computational deficiency, Leskovec et al. [17] propose an efficient method based on Cost-Effective Lazy Forward (CELF), which can reduce many unnecessary calculations. It has been reported that this method achieves about 700 times improvement on the greedy algorithm. Kim et al. [16] propose a random walk and rank merge based pruning method, which can efficiently find and filter out many uninfluential nodes. In [1, 6, 12], the authors exploit the communities of a network and devise more efficient algorithms. Additionally, on the other hand, it explores heuristic algorithms to cut down the computational cost in evaluating the influence spread. Chen et al. [5] consider that influence flows only via the maximum influence paths among nodes, and propose Maximum Influence Arborescence (MIA) algorithm and the extended prefix excluding MIA (PMIA) algorithm. Borgs et al. [3] propose a near-linear time algorithm based on random reverse reachable set. This method runs in $O(kl^2(m+n)\varepsilon^{-3}log^2n)$ expected time and provides a $(1-e^{-1}-\varepsilon)$ approximate solution with at least $(1-n^{-l})$ probability. Tang et al. [30] propose the Two-phase Influence Maximization (TIM) method that effectively reduces the number of the random reverse reachable samples. The TIM method can obtain the same approximate guarantee as the method proposed by Borgs et al. while achieving much higher empirical efficiency. Furthermore, Tang et al. [29] propose the improved Influence Maximization via Martingales (IMM) method. This method uses the martingale technique under the triggering model, which is more efficient in practice. Nguyen et al. [22] develop the Stop-and-Stare Algorithm (SSA) and its dynamic version (D-SSA), which can also provide $(1-e^{-1}-\varepsilon)$ approximate guarantee. Recently, Ohsaka et al. [23] devise an efficient and scalable algorithm for influence graph reduction under the IC model. Wang et al. [32] propose a bottom-k sketch based reverse influence sampling algorithm for both IC model and LT model.

However, all of these works mainly focus on the classical influence maximization problem that makes great effort to maximize the spread of influence in the whole network. Moreover, they do not consider both target nodes and temporal constraint in the influence diffusion.

Several work about targeted influence propagation and maximization are discussed. In [10, 11], the authors consider finding k most influential users or investigating optimal

influence propagation policies for a user in a network. Wong et al. [34] focus on the maximum flow problem, where it adds k edges into a flow graph to maximize the flow increment from a source node s to a sink node t. But this work has never involved influence maximization. Li et al. [19] study the problem that maximizes the influence spread over the users related to a certain topic or query keywords. Su et al. [26] consider the problem that finds a seed set to maximize the influence spread over the users, who have topic and geographical preferences on promotional products. However, none of these works has taken into account temporal constraint information in the influence diffusion, which is also a very important factor for successful and effective marketing in reality. Our proposed problem focuses on maximizing the influence on a specific set of target users within a finite time horizon, and these target users may be arbitrarily large or dispersed in a network. Moreover, it can be considered as an important complement to these works, and can closely mirror many real-world marketing scenarios.

3 Problem definition

We first introduce the basic IC model. Then, we give the definition of the TB-TIM problem under this model. Table 1 lists the notations that are used extensively in the rest of this paper

3.1 Independent cascade model

The IC model is widely used in the influence maximization problem. A social network is modeled as a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \omega)$, where $\mathcal{V}$ is a set of nodes, $\mathcal{E}$ is a set of edges and $\omega(u, v)$ is a weight function on each edge (u, v), which represents the probability that node v is activated by u. If $(u, v) \notin \mathcal{E}$, it satisfies $\omega(u, v) = 0$. In the IC model, each node has only two states, which is either active or non-active. Moreover, the state of each node can switch from non-active to active, but not vice verse. In general, the IC model works as follows. At the time step 0, a seed set $\mathcal{S}$ is selected and becomes activated initially. The influence diffusion proceeds in the discrete time steps $t = 0, 1, 2, \ldots$. Let $\mathcal{S}_t$ be the set of activated nodes at

Table 1 Notation explanation

Notation	Description
$\mathcal{G} = (\mathcal{V}, \mathcal{E})$	A directed graph with node set $\mathcal{V}$ and edge set $\mathcal{E}$
K	The number of seeds to be selected in $\mathcal{G}$
$\mathcal{L}$	A non-empty set of target nodes chosen from $\mathcal{V}$
$\omega(u, v)$	The activation probability on the edge (u, v) in $\mathcal{E}$
$\mathcal{I}_{\mathcal{L}}(\mathcal{S}\|\tau)$	The influence spread of a seed set $\mathcal{S}$ on the target set $\mathcal{L}$ within the bounded time τ in $\mathcal{G}$
$Pr(\mathcal{P}(u, v))$	The probability of node v is activated by node u along the propagation path $\mathcal{P}(u, v)$
$\mathcal{P}_{max}(u, v)$	The propagation path with the maximal influence probability between nodes u and v
$Pr(\mathcal{S}, v)$	The probability of node v is activated by a seed set $\mathcal{S}$
$Pr_t(\mathcal{S}, u)$	The probability of node u is activated by a seed set $\mathcal{S}$ at the time step t
$Pr(\mathcal{S}, u\|\tau)$	The probability of node u can be activated by a seed set $\mathcal{S}$ within the bounded time τ
$\Delta_u \mathcal{I}_{\mathcal{L}}(\mathcal{S}\|\tau)$	The incremental influence spread of node u with a seed set $\mathcal{S}$ on the target set $\mathcal{L}$ within the bounded time τ in $\mathcal{G}$

the time step t ($t \geq 0$), and it has $\mathcal{S}_0 = \mathcal{S}$. At the time step $t+1$, each active node in $\mathcal{S}_t$ has a single chance to independently activate each of its currently non-active neighbor v with an activation probability, where $v \in \mathcal{V} \setminus \cup_{0 \leq i \leq t} \mathcal{S}_i$[2]. Once the node v is activated, it stays active and continues to activate its non-active neighbors similar to the above process in the next time step. Furthermore, any node can only be activated at most once in this model. When there are no more nodes to be activated at a time step, the influence diffusion process terminates.

3.2 Problem definition and its NP-hardness

Given a positive integer K, a bounded time τ and a non-empty set $\mathcal{L} = \{u_1, u_2, \ldots, u_N\}$ ($1 \leq N < |\mathcal{V}|$), where $\mathcal{L} \subseteq \mathcal{V}$ is a specific set of target nodes. We define $\mathcal{I}_{\mathcal{L}}(\cdot) : 2^{\mathcal{V}} \rightarrow \mathbb{R}$ as a set function such that $\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$ is the influence probability that the target set $\mathcal{L}$ is activated by a seed set $\mathcal{S}$ within the bounded time τ in $\mathcal{G}$ when the influence diffusion process ends. The objective of the TB-TIM problem is to find an optimal seed set $\mathcal{S}^*$ of size at most K in $\mathcal{V} \setminus \mathcal{L}$ to maximize the influence spread $\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$ under the IC model, which can be formally expressed as $\mathcal{S}^* = argmax\{\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau) | \mathcal{S} \subseteq \mathcal{V} \setminus \mathcal{L}, |\mathcal{S}| \leq K\}$.

Theorem 1 *The TB-TIM problem is NP-hard under the IC model.*

Proof When the target set $\mathcal{L}$ is $\mathcal{V}$, the bounded time τ is infinite and it selects a seed set $\mathcal{S}$ from $\mathcal{V}$ (i.e., $\mathcal{S} \subseteq \mathcal{V}$), the traditional influence maximization problem can be regarded as a special case of the TB-TIM problem. It is well known that any generalization of a NP-hard problem is also NP-hard. Because the traditional influence maximization problem has been proven to be NP-hard under the IC model [5, 15], it can imply that the TB-TIM problem is NP-hard. □

4 Approximate algorithms

To solve the TB-TIM problem, we first propose an effective greedy algorithm with approximate guarantee. Then, to implement this algorithm, we propose two efficient heuristic methods to approximate the influence spread calculation.

4.1 Greedy algorithm

For any two sets $\mathcal{S}_1$ and $\mathcal{S}_2$ where $\mathcal{S}_1 \subseteq \mathcal{S}_2 \subseteq \mathcal{V}$, a set function $\mathcal{F} : 2^{\mathcal{V}} \rightarrow \mathbb{R}$ is monotone if $\mathcal{F}(\mathcal{S}_1) \leq \mathcal{F}(\mathcal{S}_2)$. Meanwhile, for any $w \in \mathcal{V} \setminus \mathcal{S}_2$, the set function $\mathcal{F}$ is submodular if $\mathcal{F}(\mathcal{S}_1 \cup \{w\}) - \mathcal{F}(\mathcal{S}_1) \geq \mathcal{F}(\mathcal{S}_2 \cup \{w\}) - \mathcal{F}(\mathcal{S}_2)$. In the TB-TIM problem, its influence spread function is monotone and submodular.

Theorem 2 *The influence spread function $\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$ is monotone and submodular under the IC model.*

[2] The symbol "\" represents the difference set in the set operation.

Proof Consider the "live-edge" model proposed in [15], it flips a coin once for each edge (u, v) with bias $\omega(u, v)$. In this situation, the edge (u, v) is "living" with probability $\omega(u, v)$ and "blocking" with probability $1 - \omega(u, v)$. Moreover, all coin-flips in the above process are independent of each other. Therefore, it can generate a random graph $X = (\mathcal{V}, \mathcal{E}')$ $(\mathcal{E}' \subseteq \mathcal{E})$, and its probability is $Pr(X) = \prod_{(u,v)\in\mathcal{E}'} \omega(u, v) \prod_{(u',v')\in\mathcal{E}\setminus\mathcal{E}'} (1 - \omega(u', v'))$. We define $\mathcal{X}$ as the set of all possible random graphs generated from $\mathcal{G}$ with a seed set $\mathcal{S}$. For any X in $\mathcal{X}$, $\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$ can be calculated as $\sum_{X\in\mathcal{X}} Pr(X)\mathcal{I}_{\mathcal{L}}(\mathcal{S};X|\tau)$, where $\mathcal{I}_{\mathcal{L}}(\mathcal{S};X|\tau)$ refers to the influence probability of the seed set $\mathcal{S}$ can activate the target set $\mathcal{L}$ within τ in X. According to the "live-edge" model, $\mathcal{I}_{\mathcal{L}}(\mathcal{S};X|\tau)$ equals $\sum_{u\in\mathcal{L}} \mathbb{I}(\mathcal{S}, u;X|\tau)$, where $\mathbb{I}(\mathcal{S}, u;X|\tau)$ is 1 if there exists at least one "living" path from some nodes in $\mathcal{S}$ to u within τ in X, and otherwise it is 0. Therefore, $\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$ is calculated as follows.

$$\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau) = \sum_{X\in\mathcal{X}} \sum_{u\in\mathcal{L}} Pr(X)\mathbb{I}(\mathcal{S}, u;X|\tau) \tag{1}$$

For any $v \in \mathcal{V} \setminus \mathcal{S} \cup \mathcal{L}$ in X, it is not hard to find that $\mathcal{I}_{\mathcal{L}}(\mathcal{S};X|\tau) \leq \mathcal{I}_{\mathcal{L}}(\mathcal{S} \cup \{v\};X|\tau)$, which means that $\mathcal{I}_{\mathcal{L}}(\mathcal{S};X|\tau)$ is monotone. Due to $Pr(X) \in (0, 1]$, $\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$ is monotone. In addition, for the sets $\mathcal{S}_1$ and $\mathcal{S}_2$, and any $w \in \mathcal{V} \setminus \mathcal{S}_2 \cup \mathcal{L}$, it considers that a node u in $\mathcal{L}$ is reachable from $\mathcal{S}_2 \cup \{w\}$ in X, which means that there is at least one "living" path from $\mathcal{S}_2 \cup \{w\}$ to u within τ, and the node u is not reachable from $\mathcal{S}_2$. Due to $\mathcal{S}_1 \subseteq \mathcal{S}_2$, node u must not be reachable from $\mathcal{S}_1$, but it can be reachable from $\mathcal{S}_1 \cup \{w\}$ within τ in X. It implies that $\mathcal{I}_{\mathcal{L}}(\mathcal{S}_1 \cup \{w\};X|\tau) - \mathcal{I}_{\mathcal{L}}(\mathcal{S}_1;X|\tau) \geq \mathcal{I}_{\mathcal{L}}(\mathcal{S}_2 \cup \{w\};X|\tau) - \mathcal{I}_{\mathcal{L}}(\mathcal{S}_2;X|\tau)$ in X. Therefore, $\mathcal{I}_{\mathcal{L}}(\mathcal{S};X|\tau)$ is submodular. Since $\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$ is a non-negative linear combination of the submodular functions, $\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$ is submodular. □

Given a non-negative, monotone and submodular function, we have the following important theorem [21].

Theorem 3 *For a non-negative, monotone and submodular function σ, let $\mathcal{S}$ be a set of size K generated by the greedy algorithm. Then, the set $\mathcal{S}$ satisfies $\sigma(\mathcal{S}) \geq (1 - e^{-1})\sigma(\mathcal{S}^*)$, where $\mathcal{S}^*$ is the optimal solution.*

According to Theorem 2 and 3, we propose a greedy algorithm with $(1 - e^{-1})$ approximation ratio to effectively solve the TB-TIM problem. Algorithm 1 presents the pseudocode of the greedy algorithm. We can see that its time complexity is $O(K(|\mathcal{V}| - |\mathcal{L}|)T(\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)))$, where $|\mathcal{V}|$ is the number of nodes in $\mathcal{V}$, $|\mathcal{L}|$ is the number of nodes in $\mathcal{L}$ and $T(\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau))$ is the maximum running time for calculating $\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$ in $\mathcal{G}$. However, before selecting a new seed in each iteration, this algorithm must equally evaluate each node in $\mathcal{V} \setminus \mathcal{L}$, which is very time-consuming in large networks. Therefore, we consider using the CELF optimization technique to accelerate selecting the seeds in this algorithm. In this case, when the incremental influence spread of certain nodes in the previous iterations are less than those results for other nodes in the current iteration, these nodes do not need to be evaluated repeatedly in the current iteration. As a result, some nodes can be filtered out in the seed selection.

ALGORITHM 1: Simple Greedy Algorithm

Input: $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \omega), \tau, \mathcal{L}, K$.
Output: $\mathcal{S}$.
Initialize: $\mathcal{S} \leftarrow \emptyset$;
for $i \leftarrow 1$ *to* K **do**
  $u \leftarrow argmax\{\mathcal{I}_{\mathcal{L}}(\mathcal{S} \cup \{v\}|\tau) - \mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau) | v \in \mathcal{V} \setminus \mathcal{L}\}$;
  $\mathcal{S} \leftarrow \mathcal{S} \cup \{u\}$;
  $\mathcal{V} \leftarrow \mathcal{V} \setminus \{u\}$;
end
return $\mathcal{S}$.

4.2 Calculation of the influence spread

In the greedy algorithm, an essential building block is to calculate the influence spread $\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$ of a given seed set $\mathcal{S}$ in the graph, whose special case has been reported to be #P-hard [5, 31]. Therefore, to improve the performance of calculating $\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$, instead of using simulation-based method based on the equation (1), we propose an efficient Time-bounded Maximal Propagation Probability based heuristic (TMPP) method, which integrates the information of target nodes and temporal constraint simultaneously.

For any two nodes u and v in a graph, we define a propagation path from node u to v as $\mathcal{P}(u, v) = (u = u_1, u_2, \ldots, u_q = v)\,(q \geq 2)$, which is an acyclic sequence of nodes. The edge linked with adjacent nodes in the propagation path $\mathcal{P}(u, v)$ refers to $e_i = (u_i, u_{i+1}) \in \mathcal{E}$ $(i = 1, 2, \ldots, q-1)$, and the influence probability of $\mathcal{P}(u, v)$ is $Pr(\mathcal{P}(u, v)) = \prod_{i=1}^{q-1} \omega(e_i)$. Obviously, it can find that the longer the propagation path, the smaller its influence probability. In particular, if there are multiple propagation paths between nodes u and v in the graph, we only choose the propagation path with the maximal influence probability. Accordingly, we define $\mathcal{P}_{max}(u, v)$ as the propagation path with the maximal influence probability from node u to v, i.e., $\mathcal{P}_{max}(u, v) = argmax\{Pr(\mathcal{P}) | \mathcal{P} \in \mathcal{P}(u, v|\mathcal{G})\}$, where $\mathcal{P}(u, v|\mathcal{G})$ refers to the set of all propagation paths from node u to v in $\mathcal{G}$. Additionally, it may occur that the maximal influence probabilities of certain paths are too small. In fact, they have very little impact on the calculation of the influence spread, and can be ignored. Therefore, we use a small threshold $\eta (\eta > 0)$ to prune those paths whose the maximal influence probabilities do not exceed η. In this situation, if $Pr(\mathcal{P}_{max}(u, v)) < \eta$, it considers that node u can not activate v through the propagation path $\mathcal{P}_{max}(u, v)$. For the calculation of $\mathcal{P}_{max}(u, v)$ in a graph, when translating the activation probability $\omega(e)$ on each edge e to a distance weight $-log(\omega(e))$, it is equivalent to finding the shortest path from node u to v with distance smaller than $-log(\eta)$. Therefore, it allows for Dijkstra shortest path-based algorithm to calculate it efficiently.

Let $Pr(\mathcal{S}, u|\tau)$ denote the probability that a node u in $\mathcal{L}$ is activated by a seed set $\mathcal{S}$ within the bounded time τ. For the TB-TIM problem, $\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$ is calculated as $\sum_{u \in \mathcal{L}} Pr(\mathcal{S}, u|\tau)$. To calculate the probability $Pr(\mathcal{S}, u|\tau)$ more efficiently, we build a tree structure, which includes all important propagation paths from other nodes to node u and takes u as its root. We define $Pr_t(\mathcal{S}, u)$ as the probability of u being activated by $\mathcal{S}$ at the time step t. Since there is a finite time horizon τ for the influence spread of the seed set $\mathcal{S}$ on the target nodes, it means that all possible time steps at which $\mathcal{S}$ can activate u within τ need to be considered. In this case, $Pr(\mathcal{S}, u|\tau)$ equals $\sum_{t \leq \tau} Pr_t(\mathcal{S}, u)$. Therefore, $\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$ can be calculated as follows.

$$\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau) = \sum_{u \in \mathcal{L}} \sum_{t \leq \tau} Pr_t(\mathcal{S}, u) \tag{2}$$

In the Eq. (2), it needs to calculate the probability $Pr_t(\mathcal{S}, u)$ for any node u in $\mathcal{L}$ at a time step t. To tackle such an issue, we employ a dynamic programming-based algorithm [4] in the trees. Let $\mathcal{N}_i(u)$ be the set of in-neighbour nodes of u. When the time step $t = 0$, because node u must not be in $\mathcal{S}$, it satisfies $Pr_t(\mathcal{S}, u) = 0$. When the time step $t > 0$, $Pr_t(\mathcal{S}, u)$ is recursively calculated as $\prod_{v \in \mathcal{N}_i(u)}(1 - \sum_{i=0}^{t-2} Pr_i(\mathcal{S}, v)Pr(v, u)) - \prod_{v \in \mathcal{N}_i(u)}(1 - \sum_{i=0}^{t-1} Pr_i(\mathcal{S}, v)Pr(v, u))$ in the tree. Furthermore, when a node v in $\mathcal{S}$, it has $Pr_t(\mathcal{S}, v) = 1$ for $t = 0$ and $Pr_t(\mathcal{S}, v) = 0$ for $t > 0$. Because it focuses on the spread of influence on the target set $\mathcal{L}$, we pre-compute the tree structures only for each node in $\mathcal{L}$ instead of all nodes in $\mathcal{V}$, and make use of these trees to calculate the influence on $\mathcal{L}$ by the end of time step τ. Moreover, when selecting a seed in each iteration, we only need to evaluate those nodes in the trees of the target nodes, rather than traversing all other nodes that are not in the trees. This is mainly because they would not have any influence on the target nodes. Algorithm 2 presents the pseudocode of calculating $\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$ with a given seed set $\mathcal{S}$ in a graph. The time complexity of this algorithm is $O(min(k_i, h_m)|\mathcal{L}|n_a)$, where k_i is the number of nodes in the current seed set $\mathcal{S}$, h_m is the maximum height in the trees and n_a is the average number of nodes in the trees.

ALGORITHM 2: Calculating $\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$

Input: $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \omega), \tau, \eta, \mathcal{S}, \mathcal{L}$.
Output: $\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$.
Initialize: $\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau) \leftarrow 0$;
for *each node* $u \in \mathcal{L}$ **do**
  $Pr(\mathcal{S}, u|\tau) \leftarrow 0$;
  while $t \leq \tau$ **do**
    Calculate $Pr_t(\mathcal{S}, u)$;
    $Pr(\mathcal{S}, u|\tau) \leftarrow Pr(\mathcal{S}, u|\tau) + Pr_t(\mathcal{S}, u)$;
  end
  $\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau) \leftarrow \mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau) + Pr(\mathcal{S}, u|\tau)$;
end
return $\mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$.

4.3 Fast calculation of the incremental influence spread

In each iteration of the greedy algorithm, it has to exactly calculate the incremental influence spread of each node on the target set $\mathcal{L}$ within the bounded time τ, i.e., $\Delta_u \mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau) = \mathcal{I}_{\mathcal{L}}(\mathcal{S} \cup \{u\}|\tau) - \mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$. However, when the number of nodes in a graph and the size of selected seed set are both large, this process is very expensive and time-consuming. Therefore, instead of calculating $\Delta_u \mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$ for each node u directly, we consider approximately estimating their values to greatly improve the efficiency of selecting the seed in each iteration. Due to the correlations between different seeds to other nodes in the influence diffusion, it satisfies $\Delta_u \mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau) \leq \mathcal{I}_{\mathcal{L}}(\{u\}|\tau)$. We fully employ the influence spread of single nodes, and propose a Fast Incremental Influence Spread based heuristic (FIIS) method to estimate $\Delta_u \mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$ approximately. Follow the work [20], we can approximate $\Delta_u \mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$ by multiplying a reasonable scale factor on $\mathcal{I}_{\mathcal{L}}(\{u\}|\tau)$. For each node $u \in \mathcal{V} \setminus \mathcal{S} \cup \mathcal{L}$, $\Delta_u \mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$ is estimated as follows.

$$\Delta_u \mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau) \approx \mathcal{I}_{\mathcal{L}}(\{u\}|\tau) SF(\{u\}) \tag{3}$$

In the Eq. (3), $\mathcal{I}_{\mathcal{L}}(\{u\}|\tau)$ is calculated based on the Eq. (2) when selecting the first seed in the greedy algorithm. Therefore, it can be used directly in this equation. $SF(\{u\})$ denotes the scale factor that belongs to (0,1], and it is calculated as follows.

$$SF(\{u\}) = \frac{\sum_{v \in \mathcal{N}_o(u)} Pr(u,v)(1 - Pr(\mathcal{S},v))\mathcal{I}_{\mathcal{L}}(\{v\}|\tau)}{\sum_{v \in \mathcal{N}_o(u)} Pr(u,v)\mathcal{I}_{\mathcal{L}}(\{v\}|\tau)} \tag{4}$$

In the above Eq. (4), $\mathcal{N}_o(u)$ represents the set of out-neighbour nodes of u. Let $\mathcal{N}_o(\mathcal{S})$ be the set of out-neighbour nodes of $\mathcal{S}$. For the IC model, when node v is in $\mathcal{N}_o(u) \cap \mathcal{N}_o(\mathcal{S})$, the probability $Pr(\mathcal{S},v)$ is $1 - \prod_{s \in \mathcal{S}}(1 - Pr(s,v))$ for any $s \in \mathcal{S}$ and $(s,v) \in \mathcal{E}$. When node v is in $\mathcal{N}_o(u) \setminus \mathcal{N}_o(\mathcal{S})$, $Pr(\mathcal{S},v)$ is 0. Furthermore, if node v is in $\mathcal{L}$, it considers that $\mathcal{I}_{\mathcal{L}}(\{v\}|\tau)$ is 1. In the $i(i > 1)$ iteration of the greedy algorithm, it can fast calculate the incremental influence spread $\Delta_u \mathcal{I}_{\mathcal{L}}(\mathcal{S}|\tau)$ of each node u by making use of the Eqs. (3) and (4), and select all remaining seeds efficiently. Due to it only reuses the influence spread of single nodes, and does not need to calculate the incremental influence spread of each node exactly, FIIS method can significantly improve the computational efficiency in the seed selection.

5 Experiments

We conduct extensive experiments over several real-world social networks to evaluate the performance of the proposed algorithms on various metrics. Furthermore, we also investigate the affect of some important parameters on their performance.

5.1 Experimental setup

We first introduce four real-world social network datasets. Then, we present all evaluated algorithms. Finally, we set the parameters. The code for each evaluated algorithm is implemented in C++, and all experiments are run on windows machine with an Intel Core 3.30GHz CPU and 24GB memory.

5.1.1 Experimental datasets

Four social network datasets [28] of different sizes are used, and their basic statistics are summarized in Table 2. The first dataset is Wiki-vote network, which is a voting history network from Wikipedia. Nodes in Wiki-vote represent Wikipedia users and directed edges represent the voting relationships between users. The second dataset, Epinions, is a

Table 2 The Statistics of Four Social Networks

Networks	Wiki-vote	Epinions	Email	LiveJournal
No. of Nodes	7115	76K	265K	1.3M
No. of Edges	104K	509K	420K	4.47M
Average Degree	29.2	13.4	3.17	6.76
Direction	directed	directed	directed	directed

who-trust-whom network of a popular review site, where nodes represent members of the site and a directed edge from u to v means v trusts u. The third dataset is Email network, where nodes represent email addresses and a directed edge from i to j means i sends at least one email to j. The last dataset is LiveJournal network, where nodes represent users and directed edges represent the friendships between them.

5.1.2 Evaluated algorithms

To evaluate the performance, we compare our proposed algorithms[3] with several other widely used heuristic algorithms[4]. All evaluated algorithms are presented as follows. Largest Degree method (LD). It selects nodes with the largest degrees in the whole graph, which is also used as a baseline method in [15]. Random method. It randomly selects nodes in a graph, which is popularly used by the work [4, 5, 15]. IMM method. One of the state-of-the-art heuristic algorithms for the traditional influence maximization problem proposed by [29]. For the purpose of comparisons, it does not include both target nodes and temporal constraint in the seed selection in the whole graph. The seeds selected by this method is used as the baseline solution for the TB-TIM problem. TMPP method. It calculates the influence spread based on the maximal propagation paths among nodes in the greedy algorithm, which considers the target nodes and temporal constraint information. FIIS method. It approximately estimates the incremental influence spread of each node in the seed selection in the greedy algorithm.

5.1.3 Parameters setting

To simulate the TB-TIM problem, we randomly pick a subset of nodes from $\mathcal{V}$ as the target set $\mathcal{L}$. The size of the target set $\mathcal{L}$ is defined as N. The threshold η is set to 0.001 to achieve the trade-off between the calculation of the influence spread and running time. For the activation probability on each edge, we apply the weighted cascade model [5, 15, 29, 30], where the probability is the reverse of the indegree of a node.

5.2 Experimental results and analysis

We present the experimental results and analysis for each method over the four social networks. We evaluate their performance on various metrics, such as the quality of seed set, running time. Moreover, we further evaluate the affect of some important parameters on the influence spread.

5.2.1 Quality of seed set

The quality of seed set is evaluated mainly based on the influence spread on the target set within a bounded time in a network. Figure 2 shows the influence spread of each evaluated

[3] The proposed algorithms are based on the greedy algorithm in Algorithm 1, and their difference is the method of calculating the incremental influence spread.

[4] We do not compare the greedy algorithm using Monte Carlo simulation. It mainly considers that the number of possible random graphs is exponential and usually very large, and a sufficient number of random simulations are required to obtain the accurate estimates with high probability. As a result, the time consumption of this method is too high for all social network datasets.

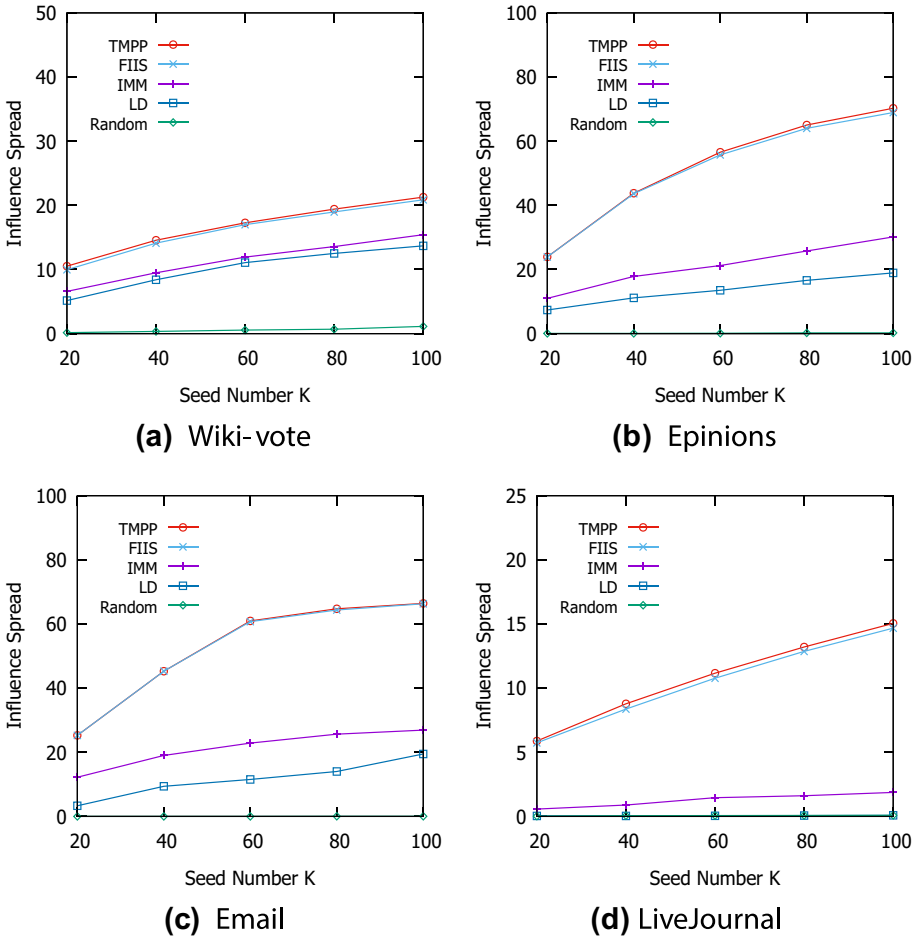

Fig. 2 The results of the influence spread over all social networks

method with varying K over the four social networks when N is 200 and τ is 5. From this figure, we can clearly observe that the influence spread of each method increases with K grows. This result keeps in line with the practical situations, where a larger number of seeders usually achieve the larger influence on a set of target users within a finite time horizon. In this figure, we can also observe that TMPP and FIIS methods achieve the similar influence spread, which are larger than other methods. Therefore, it verifies their effectiveness for solving the TB-TIM problem. Meanwhile, IMM and LD methods for the traditional influence maximization problem achieve the lower influence spread. It demonstrates that the influential nodes in the whole network may not work well for the TB-TIM problem. Unsurprisingly, Random method has the least influence spread in all methods.

5.2.2 Running time

Figure 3 shows the time taken by the evaluated methods with varying K over the four social networks when N is 200 and τ is 5. We do not include LD and Random methods due to

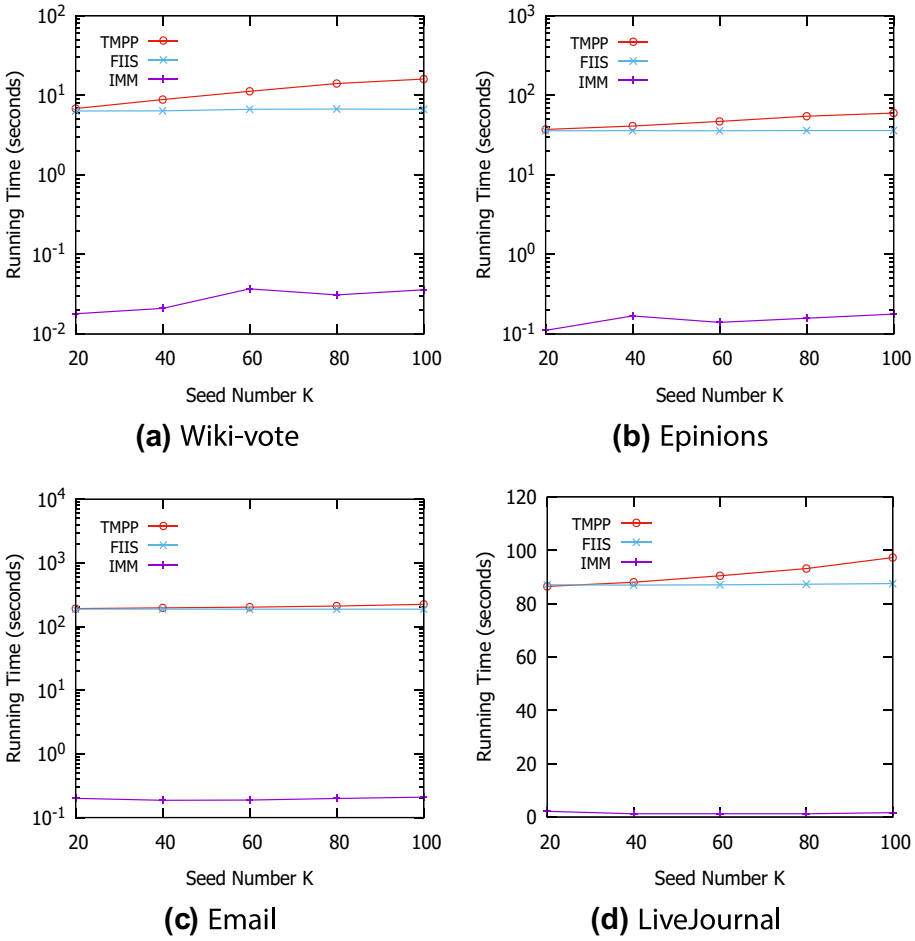

Fig. 3 The results of running time over all social networks

their running time is too trivial for all social networks. In this figure, we can find that the running time of TMPP and FIIS methods is relatively small in all networks. Moreover, the running time of FIIS method is almost unchanged even for the large social networks. Therefore, these results can verify the efficiency of the proposed methods in the seed selection for the TB-TIM problem. Because it does not need to exactly calculate the incremental influence spread of each node in the seed selection, the time consumption of IMM method is very small in the networks.

5.2.3 The affect of the size of target set on influence spread

To investigate the affect of the size of target set in the TB-TIM problem, we evaluate the performance of influence spread for different N in social networks. Figure 4 shows the influence spread of each method over the Wiki-vote and Epinions social networks when K is 50 and τ is 5. From this figure, we can see that the influence spread of each method also increases with N grows. More specifically, TMPP and FIIS methods have the similar

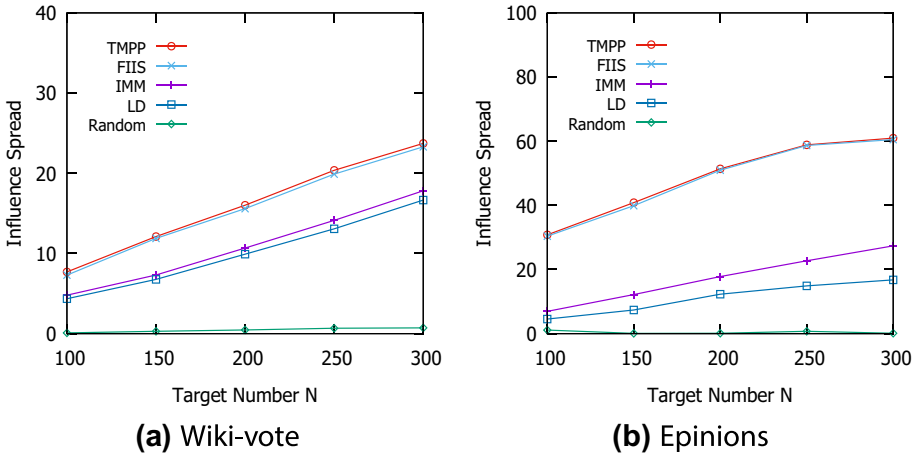

Fig. 4 The results of the influence spread for different N over the Wiki-vote and Epinions social networks

influence spread, which are larger than other evaluated methods. IMM, LD and Random methods achieve the lower influence spread in these two social networks.

5.2.4 The affect of bounded time on influence spread

We further study the affect of temporal constraint in the TB-TIM problem. Figure 5 shows the influence spread of each method with varying τ over the Wiki-vote and Epinions social networks when K is 50 and N is 150. In this figure, we can find that the influence spread increases with τ grows. However, when τ exceeds about three propagation hops, the influence spread almost no longer increases. This result is consistent with some previous

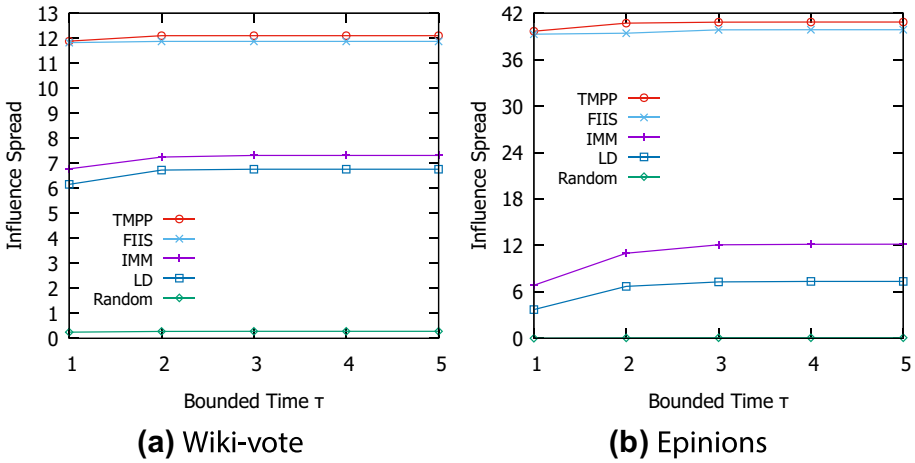

Fig. 5 The results of the influence spread for different τ over the Wiki-vote and Epinions social networks

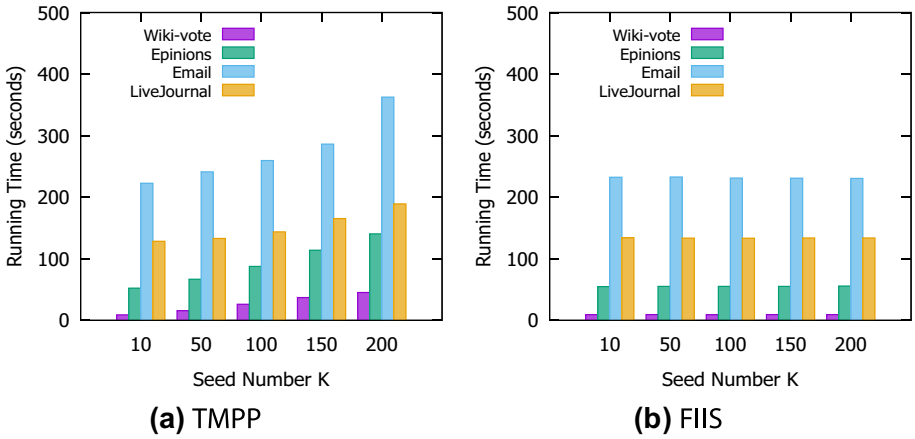

Fig. 6 The results of the running time for TMPP and FIIS methods over all social networks

measurement-driven studies that the spread of influence is basically limited within only few propagation hops from the sources in many real-world social network services. Moreover, it also demonstrates that the influential nodes for the TB-TIM problem are actually near the target nodes. Therefore, we can conclude that for the TB-TIM problem, the bounded time of influence propagation has an important impact on the influence spread.

5.2.5 The scalability of the proposed methods

We evaluate the scalability of TMPP and FIIS methods in the social network datasets of different sizes, which is measured by the running time. Figure 6 shows the running time with varying K over the four social networks when N is 300 and τ is 5. In this figure, we can see that TMPP and FIIS methods take relatively little time. Furthermore, we can find from Fig. 6(b) that FIIS method is more efficient in the large networks. For example, it only takes no more than four minutes to finish in the large Email and LiveJournal social networks, even for the large K. As expected, it can finish much faster for the relatively small Wiki-vote and Epinions social networks. It demonstrates that TMPP and FIIS methods can solve the TB-TIM problem efficiently when handling the large networks in practice.

6 Conclusion and future work

In this work, we address the TB-TIM problem in social networks, which can closely mirror many real-world marketing scenarios. To solve this problem, we develop an effective greedy algorithm with theoretical guarantee. Moreover, we further propose several heuristic methods to significantly improve the computational efficiency. Our empirical experiments over the real-world social networks of different sizes show that the proposed algorithms outperform intuitive baselines in the effectiveness and efficiency, and can scale to large networks in practice.

This work also inspires us a number of extensions and promising directions for future research. Because this work only focuses on the IC model, it is possible to study the TB-TIM problem under other propagation models. Furthermore, in addition to the social

relationships, the influence on users may also be from external sources (e.g., TV, newspapers, broadcast, etc.) in reality. Therefore, it is very interesting to further explore how social connections together with the external sources affect the spread of influence on the target users in the TB-TIM problem.

References

1. Ahmad A, Ahmad T, Bhatt A (2020) HWSMCB: A community-based hybrid approach for identifying influential nodes in the social network. Physica A: Statistical Mechanics and its Applications 545
2. Bakshy E, Hofman JM, Mason WA, Watts DJ (2011) Everyone's an influencer: Quantifying influence on Twitter. In: Proceedings of the 4th ACM International Conference on Web Search and Data Mining. pp 65–74
3. Borgs C, Brautbar M, Chayes J, Lucier B (2014) Maximizing social influence in nearly optimal time. In: Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms. pp 946–957
4. Chen W, Collins A, Cummings R, Ke T, Liu Z, Rincon D, Sun X, Wang Y (2011) Influence maximization in social networks when negative opinions may emerge and propagate. In: Proceedings of the 11th SIAM International Conference on Data Mining. pp 379–390
5. Chen W, Wang C, Wang Y (2010) Scalable influence maximization for prevalent viral marketing in large-scale social networks. In: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp 1029–1038
6. Chen Y-C, Zhu W-Y, Peng W-C, Lee W-C, Lee S-Y (2014) CIM: Community based influence maximization in social networks. ACM Trans Intell Syst Technol 5(2):1–31
7. Chevalier JA, Mayzlin D (2006) The effect of word-of-mouth on sales: Online book reviews. J Market Res 43(3):345–354
8. D'Angelo Gianlorenzo, Severini Lorenzo, Velaj Yllka (2019) Recommending links through influence maximization. Theor Comput Sci 764:30–41
9. Domingos P, Richardson M (2001) Mining the network value of customers. In: Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp 57–66
10. Guler B, Varan B, Tutuncuoglu K, Nafea M, Zewail AA, Yener A, Octeau D (2014) Optimal strategies for targeted influence in signed networks. In: Proceedings of the IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. pp 906–911
11. Guo J, Zhang P, Zhou C, Cao Y, Guo L (2013) Personalized influence maximization on social networks. In: Proceedings of the 22nd ACM International Conference on Information and Knowledge Management. pp 199–208
12. Guojie S, Xiabing Z, Yu W, Kunqing X (2015) Influence maximization on large-scale mobile social network: A divide-and-conquer method. IEEE Trans Parallel Distrib Syst 26(5):1379–1392
13. Hong W, Qian C, Tang K (2020) Efficient minimum cost seed selection with theoretical guarantees for competitive influence maximization. IEEE Transactions on Cybernetics 2:1–14
14. Huang H, Shen H, Meng Z (2019) Item diversified recommendation based on influence diffusion. Inf Process Manag 56(3):939–954
15. Kempe D, Kleinberg J, Tardos É (2003) Maximizing the spread of influence through a social network. In: Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp 137–146
16. Kim S, Kim D, Jinoh O, Hwang J-H, Han W-S, Chen W, Hwanjo Y (2017) Scalable and parallelizable influence maximization with random walk ranking and rank merge pruning. Inf Sci 415:171–189
17. Leskovec J, Krause A, Guestrin C, Faloutsos C, Van Briesen J, Glance N (2007) Cost-effective outbreak detection in networks. In: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp 420–429
18. Li H, Bhowmick SS, Cui J, Gao Y, Ma J (2015) GetReal: Towards realistic selection of influence maximization strategies in competitive networks. In: Proceedings of ACM SIGMOD International Conference on Management of Data. pp 1525–1537
19. Li Y, Zhang D, Tan K-L (2015) Real-time targeted influence maximization for online advertisements. VLDB Endowment 8(10):1070–1081
20. Liu B, Cong G, Xu D, Zeng Y (2012) Time constrained influence maximization in social networks. In: Proceedings of the 12th IEEE International Conference on Data Mining. pp 439–448

21. Nemhauser GL, Wolsey LA, Fisher ML (1978) An analysis of the approximations for maximizing submodular set functions. Math Program 14:265–294
22. Nguyen HT, Thai MT, Dinh TN (2016) Stop-and-stare: Optimal sampling algorithms for viral marketing in billion-scale networks. In: Proceedings of ACM SIGMOD International Conference on Management of Data, pp 695–710
23. Ohsaka N, Sonobe T, Fujita S, Kawarabayashi K (2017) Coarsening massive influence networks for scalable diffusion analysis. In: Proceedings of ACM SIGMOD International Conference on Management of Data. pp 635-650
24. Pham CV, Duong HV, Bui BQ, Thai MT (2018) Budgeted competitive influence maximization on online social networks. In: Proceedings of International Conference on Computational Social Networks. pp 13–24
25. Richardson M, Domingos P (2002) Mining knowledge-sharing sites for viral marketing. In: Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp 61–70
26. Sen S, Li X, Cheng X, Sun C (2018) Location-aware targeted influence maximization in social networks. J Assoc Inf Sci Technol 69(2):229–241
27. Shi Q, Wang C, Ye D, Chen J, Feng Y, Chen C (2019) Adaptive influence blocking: Minimizing the negative spread by observation-based policies. In: Proceedings of the 35th IEEE International Conference on Data Engineering. pp 1502-1513
28. SNAP Datasets (2014) http://snap.stanford.edu/data/
29. Tang Y, Shi Y, Xiao X (2015) Influence maximization in near-linear time: A martingale approach. In: Proceedings of ACM SIGMOD International Conference on Management of Data. pp 1539–1554
30. Tang Y, Xiao X, Shi Y (2014) Influence maximization: Near-optimal time complexity meets practical efficiency. In: Proceedings of ACM SIGMOD International Conference on Management of Data. pp 75–86
31. Valiant LG (1979) The complexity of enumeration and reliability problems. SIAM J Comput 8(3):410–421
32. Wang X, Zhang Y, Zhang W, Lin X, Chen C (2017) Bring order into the samples: A novel scalable method for influence maximization. IEEE Trans Knowl Data Eng 29(2):243–256
33. Weng J, Lim EP, Jiang J, He Q (2010) Twitterrank: Finding topic-sensitive influential Twitterers. In: Proceedings of the 3rd ACM International Conference on Web Search and Data Mining. pp 261–270
34. Wong P, Sun C, Lo E, Yiu ML, Wu X, Zhao Z, Hubert Chan T-H, Kao B (2017) Finding k most influential edges on flow graphs. Inf Syst 65:93–105
35. Zhu J, Ni P, Wang G (2020) Activity minimization of misinformation influence in online social networks. IEEE Transactions on Computational Social Systems 7(4):897–906