Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Rumor correction maximization problem in social networks

Yapu Zhang [a], Wenguo Yang [a,*], Ding-Zhu Du [b]

[a] *University of Chinese Academy of Sciences, Beijing, China*
[b] *University of Texas at Dallas, Richardson, USA*

A R T I C L E   I N F O

Article history:
Received 15 May 2020
Received in revised form 22 December 2020
Accepted 6 February 2021
Available online 9 February 2021
Communicated by M. Latapy

Keywords:
Rumor correction
Seed nodes
Boost nodes
Approximation ratio

A B S T R A C T

Admittedly, innovations can spread rapidly in online social networks, while the spread of malicious rumors can lead to a series of negative consequences. Therefore, it is necessary to take effective measures to limit the influence of negative information. In reality, people will become an adopter of innovations after being influenced by their friends. Meanwhile, they can be more likely to become a follower if they have received relevant information in advance. Motivated by these observations, we study the rumor correction maximization problem using both seed and boost nodes. We first focus on the boost nodes and propose the *Boosting Rumor Correction Maximization* (BRCM) problem under the *Boosting Independent Cascade* model. We prove that the BRCM problem is NP-hard, and the objective function is non-submodular. To handle it, we devise an efficient algorithm with a data-dependent approximation ratio. To explore the seed nodes, the *Seed Selection* problem and *Minimum Seed Selection* problem are proposed, respectively. Accordingly, we design two efficient algorithms. Finally, extensive empirical results in three networks manifest the efficiency of our approaches and show superiority over other baselines.

1. Introduction

Nowadays, many people have integrated large-scale online social sites into their daily life. As a result, online social networks have become a powerful tool to spread innovations. The research on information diffusion has crucial applications in viral marketing and rumor blocking. Motivated by these applications, there is extensive research on information spreading in social networks.

Important research on information propagation can trace to the *Influence Maximization* (IM) problem, which is first formulated by Domingos and Richardson [1,2]. Kempe, Kleinberg, and Tardos [3] then considered influence as a combinatorial optimization, and proposed *Independent Cascade* (IC) and *Linear Threshold* (LT) models. Since then, there has been extensive study of the IM problem, from both theoretical and practical views [4–9]. However, the rapid spread of information is not always beneficial. Such negative information and malicious rumor may lead to a series of adverse effects, like reputation damage, economic decline, and even global panic. Therefore, it is necessary to study the *Rumor Containment* (RC) problem.

Generally, a directed graph denotes a social network. The nodes and edges describe the users and their relationships, respectively. Unlike the traditional IM problem, the RC problem aims to limit the influence of rumors [10–13]. Take Fig. 1 as an instance. Assume that each grey node is infectious (i.e., the adopter of the rumor), each white node is inactive, and the influence probability in each edge is 0.5. If the seed's budget is one, node u is the best choice to be the singleton seed node

[*] Corresponding author.
E-mail addresses: zhangyapu16@mails.ucas.ac.cn (Y. Zhang), yangwg@ucas.ac.cn (W. Yang), dzdu@utdallas.edu (D.-Z. Du).

https://doi.org/10.1016/j.tcs.2021.02.014
0304-3975/© 2021 Elsevier B.V. All rights reserved.

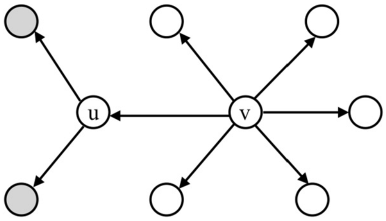


Fig. 1. Example illustrating the RCM problem.

for the RC problem since it can correct more infectious nodes. However, node v can activate more nodes in the network, and it is the best choice for the IM problem.

In this paper, different from the previous work, we take both seed nodes and boost nodes into account to study the *Rumor Correction Maximization* (RCM) problem. The boost nodes can be more likely to become the adopters when they have received relevant information in advance [14]. However, it may not be the adopters without the influence of neighbors. More specifically, a company competes for a market with others. It can not only provide some influential users with free samples but also some individuals with leaflets. Moreover, the cost of the leaflets is much less than the free samples. This company wishes that it can obtain the maximum profit through these two strategies. In this scenario, we can consider the users with free samples as the seed nodes and the users with leaflets as boost nodes.

Suppose that there are two kinds of information in a social network, namely rumor and truth. We consider the diffusion of rumor has terminated, and define the users who are influenced by rumor as the infectious nodes. In this paper, we try to market truth and hope that more infectious nodes become the adopter of truth. We call the infectious users who eventually accept truth as corrected nodes. More formally, there are two kinds of information (i.e., rumor and truth) spreading in a social network. We assume that the propagation process of the rumor has stopped and know the infectious nodes. Given a budget, the RCM problem asks for both seed nodes and boost nodes to maximize the expected number of corrected nodes.

In summary, our contributions are the following:

- We study a novel *Rumor Correction Maximization* (RCM) problem, considering both seed nodes and boost nodes. According to the boost node, we present the *Boosting Independent Cascade* (BIC) model, an extension of the IC model. Furthermore, we propose the *Boosting Rumor Correction Maximization* (BRCM) problem.
- We discuss that the BRCM problem is NP-hard under the BIC model, and computing the spread of rumor correction is #P-hard. Moreover, the objective is neither submodular nor supermodular. Based on the *k-Potentially Reverse Reachable* (k-PRR) graphs, we first estimate the objective function. Then, we construct the submodular lower and upper bounds. Furthermore, we devise an approximate solution with a data-dependent factor using the *Reverse Influence Sampling* (RIS) technique and the *Sandwich Approximation* (SA) strategy.
- To explore the seed nodes, we propose the *Seed Selection* (SS) problem and *Minimum Seed Selection* (MSS) problem, respectively. The efficient algorithms for them are presented in this paper. Finally, we manifest the efficiency of our methods and show superiority over other baselines using extensive experimental results.

The rest of this paper is organized as below. Section 2 reviews the related works. We show the diffusion model, define the boosting rumor correction maximization problem, and devise the efficient algorithms in Section 3 and Section 4. We propose and solve the seed selection problem and the minimum seed selection problem in Section 5. Section 6 studies the experimental results. And we conclude our works in Section 7.

2. Background and related work

Influence maximization (IM) Problem. Let a directed graph $G = (V, E)$ be a social network, in which V is nodes set and E is edges set. Given a network G and a constant k, the IM problem asks for k nodes to maximize the influence on a diffusion model. The problem is NP-hard, and the objective function is non-negative, monotone and submodular function [3]. Here, a set function $f : 2^V \to R^+$ is *submodular* if $f(S \cup \{v\}) - f(S) \geq f(T \cup \{v\}) - f(T)$ for any set $S \subseteq T \subseteq V$ and node $v \in V \backslash T$. f is *monotone*, if $f(S) \leq f(T)$ for any set $S \subseteq T \subseteq V$. Then, a $(1 - 1/e)$-approximate solution is derived using the *Greedy Hill Climbing* algorithm [15]. However, this greedy algorithm can hardly handle large-scale networks since computing the influence spread is #P-hard under both IC and LT models [4,5]. Recently, Borgs et al. [16] made a theoretical breakthrough. Using *Reverse Influence Sampling* (RIS) technique, they devised an efficient algorithm. Motivated by this method, Tang et al. [17,18], Nguyen et al. [19] and Tang et al. [20] then proposed more efficient algorithms, respectively. Meanwhile, some researchers study non-submodular influence maximization problem [21–23].

Rumor Containment (RC) Problem. Different from the IM problem, the RC problem aims to limit the spread of influence. Currently, there are two main techniques to solve it. On the one hand, some researchers solve the RC problem by changing the structure of social networks. These methods try to remove the most influential nodes or the key edges between two nodes. Z. Wang et al. [24] address this problem by discovering and blocking a certain number of users who have not been influenced by rumors. Kimura et al. [25] control the influence from the view of preventing a certain number of links in the

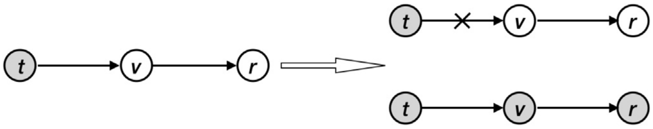


Fig. 2. Example illustrating the diffusion process.

network. On the other hand, people focus on spreading positive information in such a way that the influence of rumors can be limited. X. He et al. [26] try to block the influence diffusion of one item by choosing some seed nodes of its competition item. Saxena et al. [27] aims at finding a subset of nodes to maximize the number of users who have been exposed to the rumor but would like to believe the truth.

Rumor Propagation Model. There are many models to describe the rumor propagation process. Maki-Thompson [28], and Daley-Kendall [29] rumor models are among the earliest stochastic process. In their work, people are divided into three types: ignorants, spreaders, and stiflers. Each node selects a state in continuous time according to Markov chains. Considering the process on random networks, Watt [30] devised a model of global cascades. Each node is assigned a threshold, and it can become active if at least a threshold fraction of its neighbors are active. Most of the researchers discussed the RC problem based on the epidemic models, such as susceptible-infected (SI) model, SI-susceptible (SIS) model, and SI-recovery (SIR) model. These epidemic models are based on the dynamics of differential equations. Furthermore, given the network topology, various information propagation models are considered. Each node chooses to adopt the information from its active neighbors in discrete time with a success probability. For instance, J. Zhu et al. [31] proposed the activity minimization of misinformation influence problem on an independent cascade model.

3. Model and problem formulation

3.1. Model

Given a network $G = (V, E)$ with $|V| = n$ and $|E| = m$, there are two influence probabilities p_{uv} and p'_{uv} (with $p'_{uv} \geq p_{uv}$) on each edge e_{uv}. In the network, each node is infectious (i.e., an adopter of rumor), or active (i.e., an adopter of the truth), or inactive. Suppose the process of spreading the rumor has terminated and let the set of infectious nodes be R. Given a set of truth nodes T and a set of boost nodes B, the *Boosting Independent Cascade* (BIC) model is as below.

- At step $t = 0$, we activate the truth set T.
- At step $t \geq 1$, each newly-activated node u only has one chance to influence each out-neighbor v. If v is a boost node, its in-neighbor u will activate v with probability p'_{uv}. Otherwise, u influences v with probability p_{uv}. Moreover, u can not influence any other nodes after step t.
- The above process terminates when no nodes will be further activated.

To illustrate this process better, we show a simple example in Fig. 2. We let the grey node t be a truth node, and node r be an infectious node. Let the influence probability $p_{tv} = p_{vr} = 0.5$ and the boost influence probability $p'_{tv} = p'_{vr} = 1$. As shown in Fig. 2, if v is not the boost node, then the process may end at step $t = 1$ since v is not activated by the node t. If v is the boost node, then v must be activated by t since the activation probability $p'_{tv} = 1$. Furthermore, v will influence the infectious node r, succeeding with probability p_{vr}.

3.2. Problem definition

Given the infectious nodes set R, truth set T, and boost set B, let $f_R(T, B)$ be the expected number of corrected nodes in R. There may arise a question of how to select both T and B such that $f_R(T, B)$ is maximized under a limited budget. We call this problem as *Rumor Correction Maximization* (RCM) problem, and this problem is hard to answer directly. First, even if T is known, the computation of the objective function is #P-hard, and the issue of finding the size-k boost set B is NP-hard, which has been proved in this paper. Second, different combinations of T and B will cause different results, and there are exponential possibilities. To handle it, we give a feasible solution T, and then select the boost set B. In the following, we will first discuss the selection of B when T is known.

Definition 1. Let $G = (V, E)$ be a directed graph with two influence probabilities p_{uv} and p'_{uv} on each edge e_{uv}. Given an infectious nodes set R and a seed set T, the *Boosting Rumor Correction Maximization* (BRCM) problem finds a set B with k nodes to maximize $f_R(T, B)$. That is,

$$\begin{aligned} &\max f_R(T, B), \\ &s.t. |B| = k. \end{aligned} \tag{1}$$

Theorem 1. *The BRCM problem is NP-hard under the BIC model.*

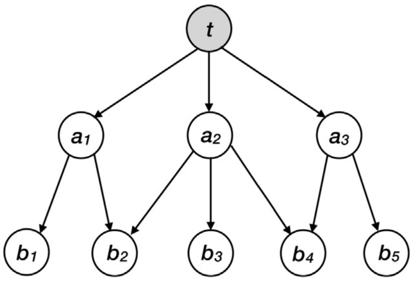


Fig. 3. Example illustrating the NP-hardness: $U = \{u_1, u_2, \ldots, u_5\}$, $S = \{S_1, S_2, S_3\}$, where $S_1 = \{u_1, u_2\}$, $S_2 = \{u_2, u_3, u_4\}$, and $S_3 = \{u_4, u_5\}$.

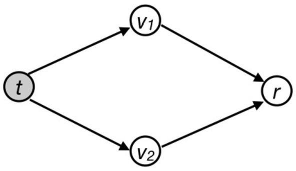


Fig. 4. Example illustrating non-submodularity.

Proof. We show it by reducing from the set cover problem [32]. Denote by $U = \{u_1, u_2, \ldots, u_n\}$ the ground set, and $S = \{S_1, S_2, \ldots, S_m\}$ a collection of sets. A set cover problem is to identify if there are k sets in S such that their union is equal to U.

Consider an arbitrary example as follows. We create a corresponding directed tripartite graph with $1+n+m$ nodes. Fig. 3 shows how we construct the graph. Let node t be the singleton seed node for truth T, and $R = \{a_1, a_2, \ldots, a_m, b_1, b_2, \ldots, b_n\}$, where node a_i corresponding to set S_i, and node b_j corresponding to node u_j. If S_i contains u_j, there is a directed edge from a_i to b_j with $p_{a_i,b_j} = p'_{a_i,b_j} = 1$. For every node $a_i \in R$, there is a directed edge from t to a_i with $p_{t,a_i} = 0$ and $p'_{t,a_i} = 1$.

We observe that the set cover problem is equivalent to see if there is a boost set B of k nodes in this graph such that $f_R(T, B) \geq n + k$. Thus, the theorem follows. □

Theorem 2. *Given sets R, T and B, computing $f_R(T, B)$ is #P-hard.*

Proof. We prove it by reducing from counting problem of $s - t$ connectedness in a directed graph [33]. And the details of this proof are in the appendix. □

Lemma 1. *The objective function of the BRCM problem is neither submodular nor supermodular.*

Proof. Fig. 4 presents a counterexample. Let the infectious nodes set $R = \{r\}$ and the truth set $T = \{t\}$. Suppose that the influence probability and boost influence probability on each edge is 0.5 and 1, respectively. We have $0.3125 = f_R(T, \{r\}) - f_R(T, \emptyset) \leq f_R(T, \{v_1, r\}) - f_R(T, \{v_1\}) = 0.375$, and $0.1875 = f_R(T, \{v_2\}) - f_R(T, \emptyset) \geq f_R(T, \{v_1, v_2\}) - f_R(T, \{v_1\}) = 0.125$. Thus, the lemma is proved. □

4. Approximation algorithm

4.1. Generate a random k-PRR graph

Definition 2. Given a graph $G = (V, E)$, a node $r \in R$ and an integer k, we first construct a deterministic graph G' corresponding to the influence propagation on each edge $e_{uv} \in E$. In this graph G', each edge e_{uv} is *live* with probability p_{uv}, *live-upon-boost* with probability $p'_{uv} - p_{uv}$, and *blocked* with probability $1 - p'_{uv}$. A *k-Potentially Reverse Reachable* (k-PRR) graph g is a subgraph of G' containing all paths between nodes in T and r with the *non-blocked* edges, and each path has at most k *live-upon-boost* edges. We say the k-PRR graph is random if node r is chosen from R uniformly at random.

We call that a path is *live* if each edge of this path is *live*. For a *live-upon-boost* edge e_{uv}, we also call the edge is *live upon boosting* v. If a path is not *live*, and each edge on this path is either *live* or *live upon boosting* v in B, then the path is *live-upon-boost*. Also, we define the path as *live upon boosting* B. For any path in a random k-PRR graph, the number of *live-upon-boost* edges is no more than k. Fig. 5 presents an instance of the 2-PRR graph. There are 7 nodes and 6 edges in this example. Let node r be the singleton infectious node, and truth set $T = \{t_1, t_2, t_3\}$. The *live*, *live-upon-boost*, and *blocked* edges are presented by solid, dashed, and dotted arrows with crosses, respectively. The subgraph in the box represents a

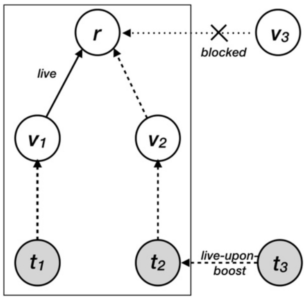


Fig. 5. Example of a 2-PRR graph.

Algorithm 1 Generate a k-PRR graph.

```
Input: $G=(V,E), k, R, T, r$
Output: $g$
Generate a graph $g$ with a singleton node $r$
Compute a deterministic graph $G'$
Create a queue $Q$ with the node $r$
Initialize $d[r] \leftarrow 0$ and $d[v] \leftarrow \infty, \forall v \neq r$
Initialize $visit\_set = \{r\}$
while $Q$ is not empty do
    $v \leftarrow Q.dequeue()$
    for $u \in N^+(v)$ do
        if $d[u] < d[v] + weight[e_{uv}]$ then
            if $u \in visit\_set$ then
                Continue
            end if
        else
            $d[u] = d[v] + weight[e_{uv}]$
            $visit\_set.add(u)$
        end if
        if $d[u] > k$ then
            Continue
        else
            Add edge $e_{uv}$ into $g$
            $Q.enqueue(u)$
        end if
    end for
end while
return $g$
```

2-PRR graph for node r, which contains 5 nodes and 4 edges. In this 2-PRR graph, the path from t_1 to r is *live upon boosting* v_1.

Let $N^+(v)$ and $N^-(v)$ be the set of in-neighbors and out-neighbors for node v, respectively. Next, we show how to generate a random k-PRR graph and conclude this process in Algorithm 1.

We first construct a graph g with the singleton node r. A deterministic graph G' is generated by the status of each edge (i.e., *live*, *live-upon-boost*, or *blocked*). If the edge e_{uv} is *live-upon-boost*, we set $weight[e_{uv}]=1$. Otherwise, we have $weight[e_{uv}]=0$. Among all paths from v to r, we define by $d[v]$ the minimum size of *live-upon-boost* edges. Initially, we have $d[r]=0$ and $d[v]=\infty$ for any $v \neq r$. Then, a queue Q with the singleton element r is created. To generate a k-PRR graph, we use Breadth-First Search (BFS) method, and the detailed backward BFS is as follows. We repeatedly dequeue and enqueue the node via the minimum number of *live-upon-boost* edges, until Q is empty. For the node v that newly out of the queue Q, we take each edge e_{uv}, satisfying $d[u] \leq k$, into the graph g. The graph g obtained by Algorithm 1 is k-PRR graph for node r.

We give a set function $I_g : 2^V \to \{0,1\}$. For any boost set B, $I_g(B)=1$ if and only if there is a *live* path or *live upon boosting* B path from node t in T to r. Otherwise, $I_g(B)=0$.

Lemma 2. *Given the sets* R, T *and* B, *we have* $f_R(T,B) = |R| \cdot E[I_g(B)]$.

Proof. For a fixed node $r \in R$, we can conduct a k-PRR graph g according to Algorithm 1. Observe that node r is corrected by T means that there is a *live* path or *live upon boosting* B path from nodes in T to r. Therefore, node r can be corrected in this graph g if and only if $I_g(B)=1$.

Furthermore, $f_R(T,B)/|R|$ means the probability that a selected node $r \in R$ can be corrected by T. Let g be a k-PRR graph for node r. Since the randomness of g, $E[I_g(B)]$ equals the probability that there is a *live* path or *live upon boosting* B path from nodes in T to r. Thus, the lemma follows. □

4.2. Submodular bounds

Given any boost set $B \subseteq V$, we define $I_g^-(B) = \max\{I_g(\{v\}) | v \in B\}$. For the example in Fig. 5, we have $I_g(\{v_2, r\}) = 1$, and $I_g^-(\{v_2, r\}) = 0$. Based on the definition of I_g^-, we design a submodular lower bound as shown in Lemma 3.

Lemma 3. *Given sets R and T, let $f_R^-(T, B) = |R| \cdot E[I_g^-(B)]$ for any boost set $B \subseteq V$. f_R^- is submodular and $f_R^-(T, B) \le f_R(T, B)$ for any $B \subseteq V$.*

Proof. By the definitions of I_g and I_g^-, we have $I_g^-(B) \le I_g(B)$ for all $B \subseteq V$. Thus, it is only needed to prove that the function I_g^- is submodular. Given any two boost sets B, B' (with $B \subseteq B' \subseteq V$) and a node $b \in V \setminus B'$, we are expected to show that

$$I_g^-(B' \cup \{b\}) - I_g^-(B') \le I_g^-(B \cup \{b\}) - I_g^-(B). \tag{2}$$

Consider the following three cases:
(1) If $I_g^-(B) = 1$, then $I_g^-(B \cup \{b\}) = I_g^-(B') = I_g^-(B' \cup \{b\}) = 1$;
(2) If $I_g^-(B) = I_g^-(B') = 0$, then $I_g^-(B \cup \{b\}) = I_g^-(B' \cup \{b\}) = I_g^-(\{b\})$;
(3) If $I_g^-(B) = 0$, and $I_g^-(B') = 1$, then $I_g^-(B \cup \{b\}) \le I_g^-(B' \cup \{b\}) = 1$.
Obviously, Eq. (2) holds in any case, and the lemma follows. □

Different from the original objective function, in a random k-PRR graph g, the submodular lower bound only allows that there is at most one *live-upon-boost* edge in each path from nodes in T to r.

Let $I_g^+ : 2^V \to \{0, 1\}$ and g be a k-PRR graph for node r. $I_g^+(B) = 1$ if and only if there is a *live* path or a *live upon boosting* B' path from nodes in T to r with $B \cap B' \ne \emptyset$. Here, for any path, B' is the minimum set such that the path is *live-upon-boost*. For an example in Fig. 5, we have $I_g^+(\{v_2\}) = 1$ and $I_g(\{v_2\}) = 0$. Based on the definition of I_g^+, we can also devise a submodular upper bound of the original function as follows.

Lemma 4. *Given sets R and T, let $f_R^+(T, B) = |R| \cdot E[I_g^+(B)]$ for any boost set $B \subseteq V$. f_R^+ is submodular and $f_R(T, B) \le f_R^+(T, B)$ for any $B \subseteq V$.*

Proof. By the definitions of I_g and I_g^+, we have $I_g(B) \le I_g^+(B)$ for any $B \subseteq V$. Similar to Lemma 3, we can obtain its submodularity. □

4.3. Estimation

Let g be a k-PRR graph for node r and $\Phi_g = \{v \in V : I_g^-(\{v\}) - I_g^-(\emptyset) = 1\} = \{v \in V : I_g(\{v\}) - I_g(\emptyset) = 1\}$. For any node $v \in \Phi_g$, it implies that there will be a path with *live upon boosting* v in the graph g. For brevity, we let $\Delta_B f_R^-(T) = f_R^-(T, B) - f_R(T, \emptyset)$ and $\Delta_B f_R^+(T) = f_R^+(T, B) - f_R(T, \emptyset)$.

Lemma 5. *Given sets R and T, $\Delta_B f_R^-(T) = |R| \cdot \Pr[B\, covers\, \Phi_g]$ for any $B \subseteq V$.*

Proof. Let g be a random k-PRR graph. If there is no any *live* path, but at least one path is *live upon boosting* $v \in B$, then $I_g^-(B) - I_g(\emptyset) = 1$. Otherwise, $I_g^-(B) - I_g(\emptyset) = 0$. Moreover, if B covers Φ_g (i.e., $B \cap \Phi_g \ne \emptyset$), we have $I_g^-(B) - I_g(\emptyset) = 1$ and $E[I_g^-(B) - I_g(\emptyset)] = \Pr[B\, covers\, \Phi_g]$. The lemma is proved, since $\Delta_B f_R^-(B)/|R| = E[I_g^-(B) - I_g(\emptyset)]$. □

Similarly, we define a set $\Psi_g = \{v \in V : I_g^+(\{v\}) - I_g^+(\emptyset) = 1\} = \{v \in V : I_g^+(\{v\}) - I_g(\emptyset) = 1\}$.

Lemma 6. *Given sets R and T, $\Delta_B f_R^+(T) = |R| \cdot \Pr[B\, covers\, \Psi_g]$ for any $B \subseteq V$.*

Proof. We omit this proof since it is similar to Lemma 5. □

Denote by $\mathcal{G}$ a collection of k-PRR graphs with $|\mathcal{G}| = \theta$, i.e., $\mathcal{G} = \{g_1, g_2, \ldots, g_\theta\}$. Accordingly, we have $\Phi = \{\Phi_{g_1}, \Phi_{g_2}, \ldots, \Phi_{g_\theta}\}$. Let $\Lambda_\Phi(B)$ be the fraction of sets in Φ covered by B. According to the Chernoff bounds and Lemma 5, we can estimate $f_R^-(T, B)$ by using $\frac{|R|}{\theta} \cdot \sum_{i=1}^{\theta} I_{g_i}(\emptyset) + |R| \cdot \Lambda_\Phi(B)$, when θ is sufficiently large. And the Chernoff bounds are as follows.

Lemma 7. *([17]) Let $x_1, x_2, \ldots, x_\theta$ be independent and identically distributed random variables sampled from a distribution on* $[0, 1]$ *with a mean μ, and let $X = \sum_{i=1}^{\theta} x_i$. For any $\delta \ge 0$,*

Algorithm 2 Greedy Max-Coverage.

```
Input: $k, \Phi$
Output: $S$
Initialize $S = \emptyset$
for $i = 1$ to $k$ do
    $v_i \leftarrow$ the node that covers the most sets in $\Phi$
    $S \leftarrow S \cup \{v_i\}$
    remove all sets in which $v_i$ appears
end for
```

$$\begin{aligned}&\Pr[X-\theta\mu \geq \delta\cdot\theta\mu] \leq \exp(-\tfrac{\delta^2}{2+\frac{2}{3}\delta}\theta\mu),\\&\Pr[X-\theta\mu \leq -\delta\cdot\theta\mu] \leq \exp(-\tfrac{\delta^2}{2}\theta\mu).\end{aligned} \tag{3}$$

Suppose that truth set T is known, and the number of k-PRR graphs θ is large enough. We may consider obtaining an approximate solution for the submodular lower bound $f_R^-(T, B)$ using a Greedy Max-Coverage algorithm, which is concluded in Algorithm 2. At every iteration, it chooses the node with the maximum marginal coverage.

Next, we discuss the size of $\mathcal{G}$ and give an approximate solution with the performance guarantee for the submodular bound.

Lemma 8. *Let parameters ε and $\delta \in (0,1)$ be fixed. Algorithm 2 guarantees a $(1-1/e-\varepsilon)$-approximate solution for $\max\{f_R^-(T,B) : B \subseteq V, |B| = k\}$ with at least $1-\delta$ probability, if*

$$|\Phi| = \theta \geq \frac{2|R|\Big((1-1/e)\sqrt{\ln\frac{2}{\delta}}+\sqrt{(1-1/e)(\ln\binom{n}{k}\ln\frac{2}{\delta}}\Big)^2}{\varepsilon^2 k}.$$

Proof. This proof is similar to [18], and we show this in the appendix. □

Algorithm 3 Node Selection for Lower Bound.

```
Input: $G, R, T, k, \varepsilon, \delta$
Output: $B_L$
Initialize $\theta_{max}$ and $\theta_0$ by using Eq. (4) and Eq. (5), respectively
Generate two collections of $k$-PRR graphs $\mathcal{G}_1$ and $\mathcal{G}_2$, with $|\mathcal{G}_2| = |\mathcal{G}_2| = \theta_0$
Create two sets $\Phi_1$ and $\Phi_2$ with respect to $\mathcal{G}_1$ and $\mathcal{G}_2$, respectively
$i_{max} \leftarrow \lceil \log_2 \frac{\theta_{max}}{\theta_0} \rceil$
for $i = 1$ to $i_{max}$ do
    $B_L \leftarrow$ a size-$k$ subset based on $\Phi_1$ according to Algorithm 2
    Compute $\mu(B_L^*)$ and $\nu(B_L)$ by Eq. (7) and Eq. (8), respectively, with $\delta_1 = \delta_2 = \delta/(3i_{max})$
    $\alpha \leftarrow \nu(B_L)/\mu(B_L^*)$
    if $\alpha \geq 1-1/e-\varepsilon$ then
        return $B_L$
    end if
    Generate two new collections of $k$-PRR graphs $\mathcal{G}_1$ and $\mathcal{G}_2$, with $|\mathcal{G}_2| = |\mathcal{G}_2| = \theta_0$
    Double the sizes of $\Phi_1$ and $\Phi_2$ according to newly-built $\mathcal{G}_1$ and $\mathcal{G}_2$
end for
return $B_L$
```

To minimize the running time, we utilize the strategy in [20] and obtain an approximate solution. The basic idea is shown in Algorithm 3. It first creates two collections of sets Φ_1 and Φ_2 based on the k-PRR graphs. Next, it obtains a set B_L, which is a maximum coverage set of Φ_1, according to Algorithm 2. Next, the approximate ratio is computed using the fraction between the lower bound of the set B_L and the upper bound of the optimal solution B_L^*. The lower and upper bounds are generated by Φ_2 and Φ_1, respectively. If the ratio is less than $1-1/e-\varepsilon$, it inserts new sets into Φ_1 and Φ_2. Otherwise, it returns the result. In this algorithm, we set

$$\theta_{max} = \frac{2|R|((1-1/e)\sqrt{\ln\frac{6}{\delta}}+\sqrt{(1-1/e)(\ln\binom{n}{k}\ln\frac{6}{\delta}})^2}{\varepsilon^2 k}, \tag{4}$$

and

$$\theta_0 = \theta_{max}\cdot\frac{\varepsilon^2 k}{|R|}. \tag{5}$$

Let B_L^* be the optimal solution, B_L be the solution obtained by Algorithm 2, and $maxMC(B, l)$ be the top l nodes with the maximum marginal coverage in Φ concerning B. There is an upper bound of $\Lambda_\Phi^u(B_L^*)$ as follows.

$$\Lambda^u_\Phi(B^*_L) = \min_{0\le i\le k} \big(\Lambda_\Phi(B_i) + \sum_{v\in maxMC(B_i,k)} (\Lambda_\Phi(B_i\cup\{v\}) - \Lambda_\Phi(B_i))\big), \tag{6}$$

where B_i is obtained from Algorithm 2 at the i-th step.

Then, an upper bound of $\Delta f^-_R(T, B^*_L)$ and a lower bound of $\Delta f^-_R(T, B_L)$ can be constructed as follows.

$$\mu(B^*_L) = (\sqrt{\Lambda^u_{\Phi_1}(B^*_L) + \frac{\ln(1/\delta_1)}{2}} + \sqrt{\frac{\ln(1/\delta_1)}{2}})^2 \cdot \frac{|R|}{\theta}, \tag{7}$$

and

$$\nu(B_L) = \big((\sqrt{\Lambda_{\Phi_2}(B_L) + \frac{2\ln(1/\delta_2)}{9}} - \sqrt{\frac{\ln(1/\delta_2)}{2}})^2 - \frac{\ln(1/\delta_2)}{18}\big) \cdot \frac{|R|}{\theta}. \tag{8}$$

Lemma 9. *Let ε and $\delta \in (0,1)$ be fixed. Algorithm 3 ensures a $(1-1/e-\varepsilon)$-approximation for $\max\{f^-_R(T,B) : B\subseteq V, |B|=k\}$ with at least $1-\delta$ probability, when sets R and T are known.*

Proof. In [20], both $\Pr[\Delta f^-_R(T,B^*_L) \le \mu(B^*_L)] \ge 1-\delta_1$ and $\Pr[\Delta f^-_R(T,B_L) \ge \nu(B_L)] \ge 1-\delta_2$ hold. Moreover, we have $\frac{\nu(B_L)}{\mu(B^*_L)} \le \frac{\nu(B_L)+f^-_R(T,\emptyset)}{\mu(B^*_L)+f^-_R(T,\emptyset)}$. If $\alpha \ge 1-1/e-\varepsilon$, by union bound, Algorithm 3 can ensure an α-approximation with at least $1-2\delta/3$ probability in the first $i_{max}-1$ iterations. According to Lemma 8, in the last iteration, a $(1-1/e-\varepsilon)$-approximate solution is derived with at least a probability of $1-\delta/3$. Thus, we prove this lemma. □

4.4. Data-dependent approximation

We obtain Algorithm 4 by integrating *Influence Reverse Sampling* (RIS) technique [16,18,20] with the *Sandwich Approximation* (SA) strategy [21]. Using Algorithm 3, we first obtain a size-k nodes set B_L. Furthermore, we only use Ψ_1 and Ψ_2 in place of Φ_1 and Φ_2 in Algorithm 3, respectively. Then, we can have other nodes set B_U. We choose the top k nodes with maximum coverage in Ψ_1 as a solution B_O for the original problem. Finally, among sets B_O, B_L, and B_U, we return the set with the largest estimated rumor correction as the final solution.

Algorithm 4 Sandwich Approximation.

```
Input: $G, R, T, k, \varepsilon, \delta$
Output: $B$
Create the empty sets $B_O$, $B_L$, $B_U$, and $B$
$B_L \leftarrow$ a size-$k$ set obtained by Algorithm 3
$B_U \leftarrow$ a size-$k$ set obtained by Algorithm 3, in which replacing $\Phi_1$ and $\Phi_1$ with $\Psi_1$ and $\Psi_1$, respectively
$B_O \leftarrow$ the top $k$ nodes with the maximum coverage in $\Lambda_{\Psi_1}$
return $B \leftarrow \arg\max\{f_R(T,B_L), f_R(T,B_U), f_R(T,B_O)\}$
```

Theorem 3. *Let B^*_O, B^*_L, and B^*_U be the optimal solutions for maximizing the original function, lower bound, and upper bound, respectively. Algorithm 4 derives a $\max\{\frac{f^-_R(T,B^*_L)}{f_R(T,B^*_O)}, \frac{f_R(T,B_U)}{f^+_R(T,B_U)}\}(1-1/e-\varepsilon)$-approximate solution with at least $1-\delta$ probability.*

Proof. Based on Lemma 9, we have

$$\begin{aligned} f_R(T,B_L) &\ge f^-_R(T,B_L) \ge (1-1/e-\varepsilon) f^-_R(T,B^*_L) \\ &\ge \frac{f^-_R(T,B^*_L)}{f_R(T,B^*_O)} \cdot (1-1/e-\varepsilon) f_R(T,B^*_O), \end{aligned}$$

and

$$\begin{aligned} f_R(T,B_U) &\ge \frac{f_R(T,B_U)}{f^+_R(T,B_U)} f^+_R(T,B_U) \ge \frac{f_R(T,B_U)}{f^+_R(T,B_U)} (1-1/e-\varepsilon) f^+_R(T,B^*_U) \\ &\ge \frac{f_R(T,B_U)}{f^+_R(T,B_U)} (1-1/e-\varepsilon) f^+_R(T,B^*_O) \\ &\ge \frac{f_R(T,B_U)}{f^+_R(T,B_U)} (1-1/e-\varepsilon) f_R(T,B^*_O). \end{aligned}$$

Let $B = \arg\max\{f_R(T,B_L), f_R(T,B_U), f_R(T,B_O)\}$, we have

$$f_R(T,B) \ge \max\{\frac{f^-_R(T,B^*_L)}{f_R(T,B^*_O)}, \frac{f_R(T,B_U)}{f^+_R(T,B_U)}\}(1-1/e-\varepsilon) \cdot f_R(T,B^*_O) \quad \square$$

Theorem 4. *Denote by EPT the expected number of edges used for creating a random k-PRR graph. Algorithm 4 runs* $O(EPT \cdot ((k\ln n + \ln(1/\delta)(n+m)\varepsilon^{-2})))$ *expected time when* $\delta \le 1/2$.

Proof. By definitions, the expected complexity is $O(EPT)$ for generating a random k-PRR graph g. Combined with the results in [20], Algorithm 3 runs $O(EPT \cdot ((k\ln n + \ln(1/\delta)(n+m)\varepsilon^{-2})))$ expected time. Furthermore, it also requires $O(EPT \cdot ((k\ln n + \ln(1/\delta)(n+m)\varepsilon^{-2})))$ expected time at step 4 and step 5, respectively. Thus, the theorem is proved. □

5. Seed selection

5.1. Seed selection problem

Next, let us discuss how to choose the seed set T for the RCM problem. Given a deterministic graph g, define by $\sigma_g(T)$ the set of nodes which are influenced by T. By definitions, we have $f_R(T, \emptyset) = E[|R \cap \sigma_g(T)|]$. For brevity, we denote $f_R(T, \emptyset)$ by $f_R(T)$. Given a budget k, the *Seed Selection* (SS) problem is defined as follows.

Problem 1.

$$\begin{aligned} &\max f_R(T) = E[|R \cap \sigma_g(T)|], \\ &s.t.\ T \subseteq V, |T| = k. \end{aligned} \tag{9}$$

Lemma 10. $E[|R \cap \sigma_g(T)|]$ *is submodular for any* $T \subseteq V$.

Proof. Given a subset $T \subseteq V$ and a node $v \in V \setminus T$, we have $|R \cap \sigma_g(T \cup \{v\})| - |R \cap \sigma_g(T)| = |R \cap (\sigma_g(T \cup \{v\}) \setminus \sigma_g(T))|$. According to the property of influence spread σ_g in [3], we have $|\sigma_g(T \cup \{v\}) \setminus \sigma_g(T)| \le |\sigma_g(S \cup \{v\}) \setminus \sigma_g(S)|$ for any $S \subseteq T$. Thus, the lemma follows. □

Definition 3. *Reverse Reachable* (RR) set S_g for root node r contains all nodes that can reach node r in a given graph g. S_g is a random RR set, if node r is chosen uniformly at random from R.

Lemma 11. *Let* S_g *be a random RR set. For any subset* $T \subseteq V$,

$$f_R(T) = E[|R \cap \sigma_g(T)|] = |R| \cdot \Pr[T\,covers\,S_g]. \tag{10}$$

Proof. Given a random RR set S_g and set T, let node r be the root node of graph g. $\Pr[T\,covers\,S_g]$ means the probability that there is a reachable path from nodes in T to r. Moreover, $f_R(T)/|R|$ is the probability that a random node r from R can be activated by the truth set T. Furthermore, node r is activated by set T denotes T can reach node r. Thus, the lemma is proved. □

Let $\mathcal{S}$ be a collection of RR sets. Denote by $\Lambda_{\mathcal{S}}(T)$ the fraction of RR sets in $\mathcal{S}$ that covered by T. For any set T, we use $|R| \cdot \Lambda_{\mathcal{S}}(T)$ as an unbiased estimation of $f_R(T)$, according to Lemma 11. To obtain an approximate solution, we also utilize the algorithm proposed by Jing et al. [20]. Different from Algorithm 3, we generate two sets of RR sets, i.e., $\mathcal{S}_1$ and $\mathcal{S}_2$. And we use $\mathcal{S}_1$ and $\mathcal{S}_2$ in place of Φ_1 and Φ_2 in Algorithm 3.

Theorem 5. *([20]) The algorithm described above provides* $(1 - 1/e - \varepsilon)$*-approximate solution for SS problem with at least probability* $1 - \delta$. *And it takes* $O((k\ln n + \ln(1/\delta))(n+m)\varepsilon^{-2})$ *expected time.*

5.2. Minimum seed selection problem

In most cases, we have no idea about the exact number of seed nodes and boost nodes. Thus, it is desired to study the strategy for allocating budget on both seeding and boosting. Given a parameter $\alpha \in [0, 1]$, we propose the *Minimum Seed Selection* (MSS) problem in the following. The parameter represents the desired minimum fraction of infectious nodes corrected by truth set T. The MSS problem tries to give the minimum size of set T such that T can correct at least $\alpha \cdot |R|$ infectious nodes.

Problem 2.

$$\begin{aligned} &\min |T|, \\ &s.t.\ f_R(T) \ge \alpha \cdot |R|, \\ &\quad T \subseteq V. \end{aligned} \tag{11}$$

Algorithm 5 Minimum Seed Selection.

```
Input: $G, k, \varepsilon, \delta$
Output: $T$
$\Upsilon_1 \leftarrow 1 + (1+\varepsilon) \cdot \frac{4(4-e)\ln(2/\delta)}{\varepsilon^2}$
Generate a set of RR sets $\mathcal{S}$ with $|\mathcal{S}| = \Upsilon_1/\alpha$.
while $\Lambda_{\mathcal{S}}(T) < \alpha$ do
    $t \leftarrow$ the node with the maximum marginal coverage
    $T \leftarrow T \cup \{t\}$
end while
```

Table 1
Important information for networks.

Name	Wikipedia	Email	Stanford
#Nodes	7.1K	265K	282K
#Edges	104K	420K	2.3M

According to Lemma 11, we know $\Lambda_{\mathcal{S}}(T)$ is an unbiased estimation of $f_R(T)$. Given a collection of RR sets $\mathcal{S}$, there is a straightforward way to solve the MSS problem. That is, we choose the element with the largest marginal coverage into the truth set T at each iteration, until $\Lambda_{\mathcal{S}}(T) > \alpha$. To control the performance, we should determine the size of $\mathcal{S}$.

Lemma 12. *([34]) Let x_i be a random variable distributed in $[0, 1]$ with mean μ, $\Upsilon = 4(4-e)\ln(2/\delta)/\varepsilon^2$, and $\Upsilon_1 = 1 + (1+\varepsilon)\Upsilon$. If $\sum_{i=1}^{\theta} x_1 \geq \Upsilon_1$, then $\Pr[(1-\epsilon)\mu \leq \frac{\sum_{i=1}^{\theta} x_1}{\theta} \leq (1+\epsilon)\mu]$ and $E[\theta] \leq \Upsilon_1/\mu$.*

Based on the results from Dagum et al. [34], we may set $|\mathcal{S}| = \Upsilon_1/\alpha$ and ensure $|R| \cdot \Lambda_{\mathcal{S}}(T)$ can best estimate $f_R(T)$. A heuristic algorithm is concluded in Algorithm 5. We first construct a set of RR sets $\mathcal{S}$ with $|\mathcal{S}| = \Upsilon_1$. Then, we select the element with the maximum marginal coverage at each iteration, until $\Lambda_{\mathcal{S}}(T) \geq \alpha$.

Theorem 6. *Let EPT be the expected number of edges that point to the nodes in a random RR set. Algorithm 5 runs in $O(EPT \cdot \Upsilon/\alpha)$.*

Proof. On the one hand, it requires O(EPT) expected time in generating a random RR set, and the total number of RR sets is Υ/α. On the other hand, the time of both selecting the maximum marginal coverage and computing $\Lambda_{\mathcal{S}}(T)$ are linear to the total number of RR sets. Thus, the theorem is proved. □

6. Experiments

6.1. Experiment setup

Datasets. We use Wikipedia, Email, and Stanford datasets in our experiments. All these three directed networks can be obtained from SNAP [35]. The important information on these graphs is presented in Table 1.

- Wikipedia: Wikipedia is constructed by using the voting activity in elections. Each node means a Wikipedia user, and each edge from i to j denotes that individual i votes on individual j.
- Email: The information was collected from email data in a European research institution. Each email address is considered as a node. If i sends one message to j, then there is a directed edge between nodes i and j.
- Stanford: This dataset comes from the Stanford site. Each node represents a page from Stanford University, and each directed edge means the hyperlink between them.

Settings. For these three networks, the propagation probability of edge e_{uv} is chosen from $[0, 0.1]$ uniformly at random. Based on [14], we let the boost influence probability $p'_{uv} = 1 - (1 - p_{uv})^2$. We randomly select 200 nodes as rumor source, and let one simulation of their spread results as the set of infectious nodes R. Unless otherwise specified, we let $\varepsilon = 0.5$ and $\delta = 1/|V|$ in this paper. For a fair comparison, we evaluate the results of rumor correction using 10,000 Monte-Carlo estimations in each algorithm. All algorithms used in our experiments are as follows.

Algorithms.

- SS: The algorithm is basically similar to Algorithm 3. It is mainly used to solve SS problem and the difference from Algorithm 3 is shown in Section 5.
- MSS: This is Algorithm 5 proposed in this paper, and we set $\varepsilon = 0.1$.
- SA: It is Algorithm 4 in this paper.
- IMM: The algorithm is proposed in [18], and it is mainly used to solve the IM problem.
- Random: The method is a baseline, in which the nodes are randomly selected.

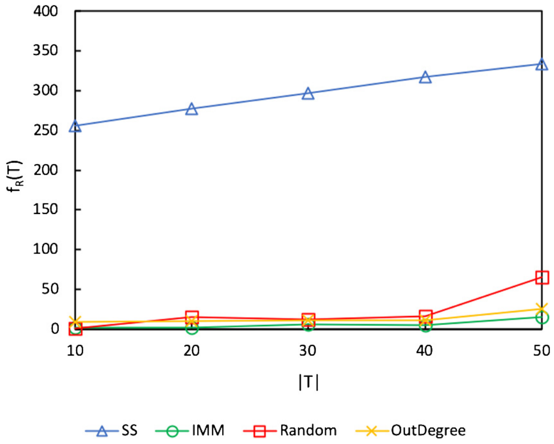


Fig. 6. Performance in different budgets.

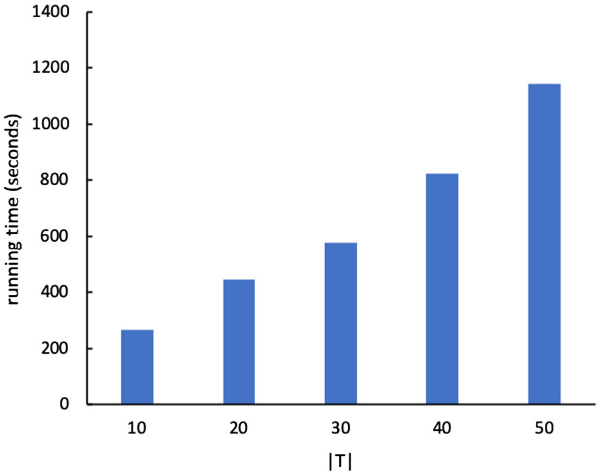


Fig. 7. Running time of the SS algorithm.

- OutDegree: This approach is widely used in various influence diffusion models. The idea is to choose nodes with the highest out-degree.

We implemented our algorithms using python. Moreover, all experiments were completed on a machine with a 2.3 GHz Intel Core i5 processor and 16 GB memory.

6.2. Experiment results

Seed selection. Fig. 6 compares the rumor correction of solutions obtained by different algorithms on the Stanford network. We choose the budget of truth set T from 10 to 50. As shown in Fig. 6, it is obvious that our proposed algorithm outperforms other methods. Meanwhile, the expected number of rumor correction is increasing by the number of seed nodes. Fig. 7 shows the running time corresponding to Fig. 6. The running time is also increasing by the size of seed nodes. It is mainly because the computation of the estimated approximation ratio and the required number of RR sets are increasing.

Minimum seed selection. To prove the efficiency of our algorithms, we compute the fraction of infectious nodes corrected by the solutions returned from Algorithm 5. On these three networks, we take the value of parameter α from 0.1 to 1, receptively. As shown in Fig. 8, the ratio increases as the value of α increases on each dataset. For a fixed α, the results do not vary much for the three networks. And the results are basically equal to α.

Boost nodes selection. Let 30 nodes from the SS algorithm be the truth set T. Fig. 9 shows the increment of rumor correction by the influence of boost nodes. As the size of boost nodes increases, the increment also increases. Our proposed algorithm (i.e., SA algorithm) outperforms other approaches on each network. The results of both the IMM algorithm and the Random algorithm perform badly on each network. However, we observe that the result is good for the OutDegree algorithm on the Email graph. That may be because there are some nodes with high out-degree that play an important

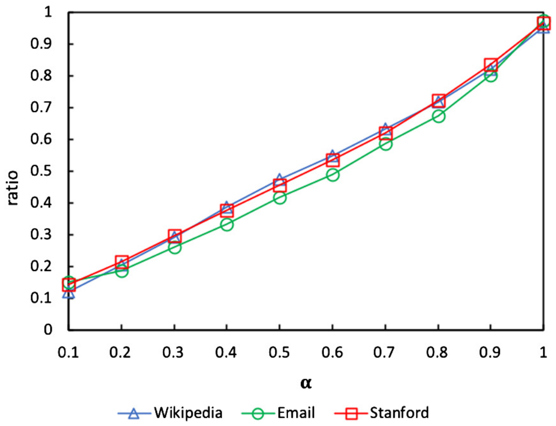
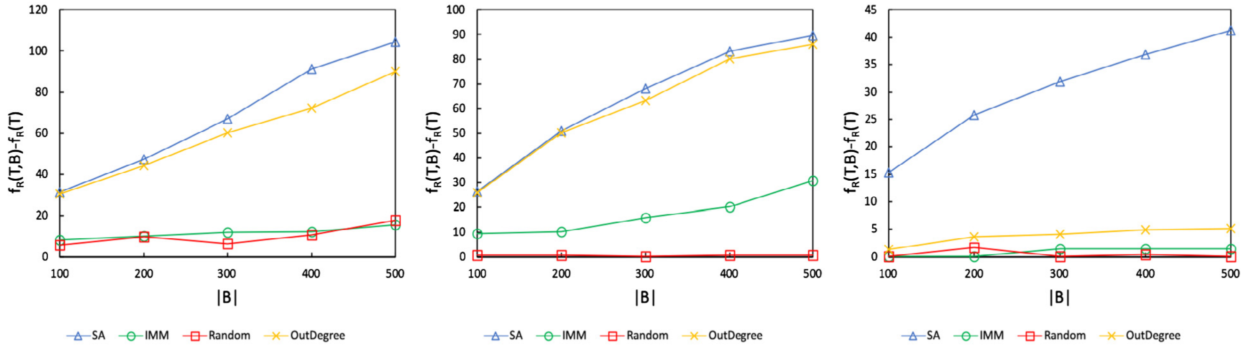


Fig. 8. The fraction of infectious nodes corrected by truth set T.



(a) Wikipedia (b) Email (c) Stanford

Fig. 9. The increment in different algorithms with budget k from 100 to 500.

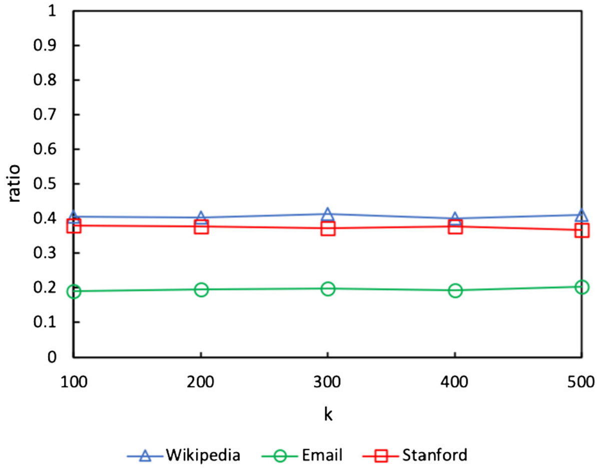


Fig. 10. Approximation ratio.

role in the network. Furthermore, it performs badly at Stanford. Our method has a data-dependent approximate factor. As shown in section 4, the SA algorithm can return a $\max\{\frac{f_R^-(T,B_L^*)}{f_R(T,B_O^*)}, \frac{f_R(T,B_U)}{f_R^+(T,B_U)}\}(1-1/e-\varepsilon)$-approximate solution. Since the precise approximation ratio is intractable to compute, we give the computable lower bound $\frac{f_R(T,B_U)}{f_R^+(T,B_U)}(1-1/e-\varepsilon)$. Fig. 10 shows the factor $\frac{f_R(T,B_U)}{f_R^+(T,B_U)}$ on three datasets. Different datasets have different ratios. Roughly, the ratio does not change much concerning the number of boost nodes. Moreover, we show the running time of the SA algorithm when the number

Table 2
Running time.

Name	Wikipedia	Email	Stanford
time	73 s	746 s	2425 s

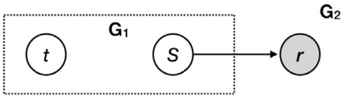


Fig. 11. Example illustrating the #P-hardness.

of boost nodes equals 100 in Table 2. As the size of networks increases, the running time increases. That is mainly because the running time of generating k-PRR graphs increases.

7. Conclusion

In this paper, we focus on studying a novel rumor correction maximization problem, which aims to control the influence of rumors. We first study the *Boosting Rumor Correction* problem and prove the hardness of the problem and property of the objective function. Combined with the *Reverse Influence Sampling* technique and *Sandwich Approximation* strategy, we develop an efficient approximation algorithm with a data-dependent guarantee. Then, we focus on the seed selection process. The *Seed Selection* problem and *Minimum Seed Selection* problem are proposed, respectively. Accordingly, two proposed algorithms solve them well. The experimental results on three networks manifest the efficiency of our methods.

There are several future directions for this research. First, we mainly separate the rumor correction maximization problem into two parts, i.e., boosting and seeding, in this paper. We should further focus on the problem that allows us to decide how to allocate budget more freely. Second, we can consider similar issues under other influence diffusion models. Third, our problem in the boosting process is non-submodular. We can explore different strategies for non-submodular optimization.

Declaration of competing interest

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

Acknowledgement

This work was done during Yapu Zhang's visiting the University of Texas at Dallas as a visiting scholar funded by the China Scholarship Council. And the work is supported partially by National Science Foundation under Grant No. 1907472, and it is supported partially by the National Natural Science Foundation of China under Grants No. 12071459, 1991022.

Appendix A. Proof of Theorem 2

As shown in Fig. 11, an instance of this problem is to count the number of subgraphs of $G_1 = (V, E)$ in which t is connected to s. For each edge $e_{uv} \in E$, let the influence probability $p_{uv} = 0.5$ and the boost influence probability $p'_{uv} = 1$. The $s - t$ connectedness counting problem is equivalent to computing the probability that t is connected to s. Furthermore, we add an edge e_{sr} with $p_{sr} = 0.5$ and $p'_{sr} = 1$, and let $G_2 = (V \cup \{r\}, E \cup \{e_{sr}\})$. Let $T = \{t\}$ and $R = B = \{r\}$. Notice that the probability that t is connected to s is exactly equal to $f_R(T, B)$ for graph G_2. Then, we can solve the $s - t$ connectedness counting problem by computing the BRCM problem. Since the $s - t$ connectedness counting problem is #P-complete [33], the computation of our problem is also #P-hard.

Appendix B. Proof of Lemma 8

Given sets R and T, suppose that B_L^* is the optimal solution for $\max\{f_R^-(T, B) : B \subseteq V, |B| = k\}$ and the optimal value $f_R^-(T, B_L^*) = OPT_L$. Before our proof of Lemma 8, we claim two facts in the following.

Claim 1. *Let $\delta_1 \in (0, 1)$, $\varepsilon_1 \geq 0$, and $\theta_1 = \frac{2\cdot|R|\cdot\log(1/\delta_1)}{OPT_L\cdot\varepsilon_1^2}$. If $\theta \geq \theta_1$, then $\frac{|R|}{\theta}\cdot\sum_{i=1}^{\theta} I_{g_i}(\emptyset) + |R|\cdot\Lambda_\Phi(B_L^*) \geq (1-\varepsilon_1)OPT_L$ holds with at least $1-\delta_1$ probability.*

Proof. For each $g_i \in \mathcal{G}$, we define a random variable $x_i = I_{g_i}(\emptyset) + \min\{1, |B_L^* \cap \Phi_{g_i}|\}$. And we know $x_i \in \{0, 1\}$ by the definitions. Using the Chernoff bounds,

$$\begin{aligned}
&\Pr[\tfrac{|R|}{\theta}\cdot\textstyle\sum_{i=1}^{\theta}x_i\le(1-\varepsilon_1)OPT_L]\\
&=\Pr[\textstyle\sum_{i=1}^{\theta}x_i\le(1-\varepsilon_1)\cdot\theta\cdot\frac{OPT_L}{|R|}]\\
&=\Pr[\textstyle\sum_{i=1}^{\theta}x_i-\theta\cdot\frac{OPT_L}{|R|}\le-\varepsilon_1\cdot\theta\cdot\frac{OPT_L}{|R|}]\\
&\le\exp(-\tfrac{\varepsilon_1^2}{2}\cdot\theta\cdot\tfrac{OPT_L}{|R|})\\
&\le\exp(-\tfrac{\varepsilon_1^2}{2}\cdot\theta_1\cdot\tfrac{OPT_L}{|R|})\\
&=\delta_1
\end{aligned}\tag{12}$$

Thus, this claim is proved. □

Claim 2. *Let $\delta_2\in(0,1)$, $\varepsilon_1\le\varepsilon$, and $\theta_2=\frac{(2-2/e)\cdot|R|\cdot\log(\binom{n}{k}/\delta_2)}{OPT_L\cdot(\varepsilon-(1-1/e)\cdot\varepsilon_1)^2}$. If $\theta\ge\theta_2$, then $f_R^-(T,B_L)\ge(1-1/e-\varepsilon)OPT_L$ holds with at least $1-\delta_2$ probability, where B_L is the set obtained from the Greedy Max-Coverage Algorithm.*

Proof. For each $g_i\in\mathcal{G}$, we define a random variable $x_i=I_{g_i}(\emptyset)+\min\{1,|B_L^*\cap\Phi_{g_i}|\}$ and let $\varepsilon_2=\varepsilon-(1-1/e)\cdot\varepsilon_1$. Given a size-$k$ node set B, B is bad if $f_R^-(T,B)\le(1-1/e-\varepsilon)OPT_L$. First, we will show that Algorithm 2 returns each bad size-k node set B with at most $\delta_2/\binom{n}{k}$ probability.

By the Chernoff bounds,

$$\begin{aligned}
&\Pr[\tfrac{|R|}{\theta}\cdot\textstyle\sum_{i=1}^{\theta}x_i-f_R^-(T,B)\ge\varepsilon_2OPT_L]\\
&=\Pr[\textstyle\sum_{i=1}^{\theta}x_i-\theta\cdot\frac{f_R^-(T,B)}{|R|}\ge\frac{\varepsilon_2OPT_L}{f_R^-(T,B)}\cdot\theta\cdot\frac{f_R^-(T,B)}{|R|}]\\
&\le\exp(-\tfrac{\varepsilon_2^2{OPT_L}^2}{2f_R^-(T,B)+\frac{2}{3}\varepsilon_2OPT_L}\cdot\tfrac{\theta}{|R|})
\end{aligned}\tag{13}$$

Assume that $f_R^-(T,B)\le(1-1/e-\varepsilon)OPT_L$,

$$\begin{aligned}
&\text{r.h.s of Eq. (13)}\\
&\le\exp(-\tfrac{(\varepsilon-(1-1/e)\cdot\varepsilon_1)^2OPT_L^2}{2(1-1/e-\varepsilon)OPT_L+\frac{2}{3}(\varepsilon-(1-1/e)\cdot\varepsilon_1)OPT_L}\cdot\tfrac{\theta}{|R|})\\
&\le\exp(-\tfrac{(\varepsilon-(1-1/e)\cdot\varepsilon_1)^2OPT_L}{2-2/e}\cdot\tfrac{\theta}{|R|})\\
&\le\exp(-\tfrac{(\varepsilon-(1-1/e)\cdot\varepsilon_1)^2OPT_L}{2-2/e}\cdot\tfrac{\theta_2}{|R|})\\
&\le\delta_2/\tbinom{n}{k}
\end{aligned}\tag{14}$$

There are $\binom{n}{k}$ bad size-k boost sets, and each of them has at most $\delta_2/\binom{n}{k}$ probability to be returned. By the union bound, none of them will be returned with at least $1-\delta_2$ probability.

By Claim 1 and Algorithm 2, we have $\frac{|R|}{\theta}\cdot\sum_{i=1}^{\theta}I_{g_i}(\emptyset)+|R|\cdot\Lambda_\Phi(B_L^*)\ge(1-\varepsilon_1)OPT_L$ and $\Lambda_\Phi(B_L)\ge(1-1/e)\cdot\Lambda_\Phi(B_L^*)$. The following inequality holds with $1-\delta_2$ probability.

$$\begin{aligned}
f_R^-(T,B_L)&\ge\tfrac{|R|}{\theta}\cdot\textstyle\sum_{i=1}^{\theta}I_{g_i}(\emptyset)+|R|\cdot\Lambda_\Phi(B_L)-\varepsilon_2OPT_L\\
&\ge\tfrac{|R|}{\theta}\cdot\textstyle\sum_{i=1}^{\theta}I_{g_i}(\emptyset)+(1-1/e)(1-\varepsilon_1)\cdot\Lambda_\Phi(B_L^*)-\varepsilon_2OPT_L\\
&\ge(1-1/e-\varepsilon)OPT_L
\end{aligned}\tag{15}$$

Thus, Claim 2 is proved. □

By Claim 1 and Claim 2, given any $\varepsilon_1\le\varepsilon$ and any $\delta_1,\delta_2\in(0,1)$ with $\delta_1+\delta_2\le\delta$, if $\theta\ge\max\{\theta_1,\theta_2\}$, Algorithm 2 ensures a size-k nodes set of $(1-1/e-\varepsilon)$-approximation for maximizing the submodular lower bound with at least $1-\delta$ probability.

In [18], there is a method of setting θ such that minimizing the expected running time while ensuring solution quality. Let $\delta_1=\delta_2=1/(2\delta)$, and θ is minimized when $\theta_1=\theta_2$. In that case,

$$\theta=\frac{2|R|\big((1-1/e)\sqrt{\ln\frac{2}{\delta}}+\sqrt{(1-1/e)(\ln\binom{n}{k}\ln\frac{2}{\delta}}\big)^2}{\varepsilon^2OPT_L}.\tag{16}$$

Note that the calculation of θ requires the optimal value of OPT_L. For convenience, we can use $f_R(T,\emptyset)$ instead of OPT_L. Moreover, we can replace it by k since $OPT_L\le k$ and the computation of $f_R(T,\emptyset)$ is #P-hard. Thus, we can set

$$\theta=\frac{2|R|\big((1-1/e)\sqrt{\ln\frac{2}{\delta}}+\sqrt{(1-1/e)(\ln\binom{n}{k}\ln\frac{2}{\delta}}\big)^2}{\varepsilon^2k}.\tag{17}$$

References

[1] P. Domingos, M. Richardson, Mining the network value of customers, in: Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2001, pp. 57–66.

[2] M. Richardson, P. Domingos, Mining knowledge-sharing sites for viral marketing, in: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2002, pp. 61–70.

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

[4] W. Chen, C. Wang, Y. Wang, 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, ACM, 2010, pp. 1029–1038.

[5] W. Chen, Y. Yuan, L. Zhang, Scalable influence maximization in social networks under the linear threshold model, in: 2010 IEEE International Conference on Data Mining, IEEE, 2010, pp. 88–97.

[6] W. Chen, H. Zhang, Complete submodularity characterization in the comparative independent cascade model, Theor. Comput. Sci. 786 (2019) 78–87.

[7] Y. Zhang, X. Yang, S. Gao, W. Yang, Budgeted profit maximization under the multiple products independent cascade model, IEEE Access 7 (2019) 20040–20049.

[8] C. Gao, H. Du, W. Wu, H. Wang, Viral marketing of online game by DS decomposition in social networks, Theor. Comput. Sci. 803 (2020) 10–21.

[9] Y. Zhang, J. Guo, W. Yang, W. Wu, Targeted activation probability maximization problem in online social networks, IEEE Trans. Netw. Sci. Eng. (2020) 1, https://doi.org/10.1109/TNSE.2020.3037106.

[10] S. Li, Y. Zhu, D. Li, D. Kim, H. Huang, Rumor restriction in online social networks, in: 2013 IEEE 32nd International Performance Computing and Communications Conference (IPCCC), IEEE, 2013, pp. 1–10.

[11] C. Budak, D. Agrawal, A. El 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.

[12] L. Fan, Z. Lu, W. Wu, B. Thuraisingham, H. Ma, Y. Bi, Least cost rumor blocking in social networks, in: 2013 IEEE 33rd International Conference on Distributed Computing Systems, IEEE, 2013, pp. 540–549.

[13] A. Tong, D.-Z. Du, W. Wu, On misinformation containment in online social networks, in: Advances in Neural Information Processing Systems, 2018, pp. 341–351.

[14] Y. Lin, W. Chen, J.C. Lui, Boosting information spread: an algorithmic approach, in: 2017 IEEE 33rd International Conference on Data Engineering (ICDE), IEEE, 2017, pp. 883–894.

[15] G.L. Nemhauser, L.A. Wolsey, M.L. Fisher, An analysis of approximations for maximizing submodular set functions—I, Math. Program. 14 (1) (1978) 265–294.

[16] C. Borgs, M. Brautbar, J. Chayes, B. Lucier, Maximizing social influence in nearly optimal time, in: Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SIAM, 2014, pp. 946–957.

[17] Y. Tang, X. Xiao, Y. Shi, Influence maximization: near-optimal time complexity meets practical efficiency, in: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, ACM, 2014, pp. 75–86.

[18] Y. Tang, Y. Shi, X. Xiao, Influence maximization in near-linear time: a martingale approach, in: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, ACM, 2015, pp. 1539–1554.

[19] H.T. Nguyen, M.T. Thai, T.N. Dinh, Stop-and-stare: optimal sampling algorithms for viral marketing in billion-scale networks, in: Proceedings of the 2016 International Conference on Management of Data, ACM, 2016, pp. 695–710.

[20] J. Tang, X. Tang, X. Xiao, J. Yuan, Online processing algorithms for influence maximization, in: Proceedings of the 2018 International Conference on Management of Data, 2018, pp. 991–1005.

[21] W. Lu, W. Chen, L.V. Lakshmanan, From competition to complementarity: comparative influence diffusion and maximization, Proc. VLDB Endow. 9 (2) (2015) 60–71.

[22] Z. Wang, Y. Yang, J. Pei, L. Chu, E. Chen, Activity maximization by effective information diffusion in social networks, IEEE Trans. Knowl. Data Eng. 29 (11) (2017) 2374–2387.

[23] V. Chaoji, S. Ranu, R. Rastogi, R. Bhatt, Recommendations to boost content spread in social networks, in: Proceedings of the 21st International Conference on World Wide Web, ACM, 2012, pp. 529–538.

[24] S. Wang, X. Zhao, Y. Chen, Z. Li, K. Zhang, J. Xia, Negative influence minimizing by blocking nodes in social networks, in: Workshops at the Twenty-Seventh AAAI Conference on Artificial Intelligence, 2013.

[25] M. Kimura, K. Saito, H. Motoda, Minimizing the spread of contamination by blocking links in a network, in: AAAI, vol. 8, 2008, pp. 1175–1180.

[26] X. He, G. Song, W. Chen, Q. Jiang, Influence blocking maximization in social networks under the competitive linear threshold model, in: Proceedings of the 2012 SIAM International Conference on Data Mining, SIAM, 2012, pp. 463–474.

[27] A. Saxena, W. Hsu, M.L. Lee, H. Leong Chieu, L. Ng, L.N. Teow, Mitigating misinformation in online social network with top-k debunkers and evolving user opinions, in: Companion Proceedings of the Web Conference 2020, WWW '20, Association for Computing Machinery, New York, NY, USA, 2020, pp. 363–370.

[28] D.J. Daley, D.G. Kendall, Stochastic rumours, IMA J. Appl. Math. 1 (1965) 42–55.

[29] D.M. Thompson, Mathematical models and applications: with emphasis on the social, life, and management sciences, ACM 10 (12) (1973).

[30] D.J. Watts, A simple model of global cascades on random networks, Proc. Natl. Acad. Sci. USA 99 (9) (2002) 5766–5771.

[31] J. Zhu, P. Ni, G. Wang, Activity minimization of misinformation influence in online social networks, IEEE Trans. Comput. Soc. Syst. 7 (4) (2020) 897–906, https://doi.org/10.1109/TCSS.2020.2997188.

[32] R.M. Karp, Reducibility among combinatorial problems, in: Complexity of Computer Computations, Springer, 1972, pp. 85–103.

[33] L.G. Valiant, The complexity of enumeration and reliability problems, SIAM J. Comput. 8 (3) (1979) 410–421.

[34] P. Dagum, R. Karp, M. Luby, S. Ross, An optimal algorithm for Monte Carlo estimation, SIAM J. Comput. 29 (5) (2000) 1484–1496.

[35] J. Leskovec, A. Krevl, SNAP datasets: Stanford large network dataset collection, http://snap.stanford.edu/data, Jun. 2014.