

CRB: A new rumor blocking algorithm in online social networks based on competitive spreading model and influence maximization

Chen Dong(董晨), Gui-Qiong Xu(徐桂琼), and Lei Meng(孟蕾)
Citation: Chin. Phys. B, 2024, 33 (**8**): 088901. DOI: 10.1088/1674-1056/ad531f

What follows is a list of articles you may be interested in

WT-FCTGN: A wavelet-enhanced fully connected time-gated neural network for complex noisy traffic flow modeling

Zhifang Liao(廖志芳), Ke Sun(孙轲), Wenlong Liu(刘文龙), Zhiwu Yu(余志武), Chengguang Liu(刘承光), and Yucheng Song(宋禹成)
Chin. Phys. B, 2024, 33 (**7**): 078901. DOI: 10.1088/1674-1056/ad3349

Identifying influential spreaders in social networks: A two-stage quantum-behaved particle swarm optimization with Lévy flight

Pengli Lu(卢鹏丽), Jimao Lan(揽继茂), Jianxin Tang(唐建新), Li Zhang(张莉), Shihui Song(宋仕辉), and Hongyu Zhu(朱虹羽)
Chin. Phys. B, 2024, 33 (**1**): 018901. DOI: 10.1088/1674-1056/acd3e0

Identifying multiple influential spreaders in complex networks based on spectral graph theory

Dong-Xu Cui(崔东旭), Jia-Lin He(何嘉林), Zi-Fei Xiao(肖子飞), and Wei-Ping Ren(任卫平)
Chin. Phys. B, 2023, 32 (**9**): 098904. DOI: 10.1088/1674-1056/acac0b

AIGCrank: A new adaptive algorithm for identifying a set of influential spreaders in complex networks based on gravity centrality

Ping-Le Yang(杨平乐), Lai-Jun Zhao(赵来军), Chen Dong(董晨),Gui-Qiong Xu(徐桂琼), and Li-Xin Zhou(周立欣)
Chin. Phys. B, 2023, 32 (**5**): 058901. DOI: 10.1088/1674-1056/ac8e56

Subtle role of latency for information diffusion in online social networks

Fei Xiong(熊菲), Xi-Meng Wang(王夕萌), Jun-Jun Cheng(程军军)
Chin. Phys. B, 2016, 25 (**10**): 108904. DOI: 10.1088/1674-1056/25/10/108904

CRB: A new rumor blocking algorithm in online social networks based on competitive spreading model and influence maximization

Chen Dong(董晨), Gui-Qiong Xu(徐桂琼)†, and Lei Meng(孟蕾)

School of Management, Shanghai University, Shanghai 200444, *China*

(Received 25 February 2024; revised manuscript received 14 May 2024; accepted manuscript online 3 June 2024)

The virtuality and openness of online social platforms make networks a hotbed for the rapid propagation of various rumors. In order to block the outbreak of rumor, one of the most effective containment measures is spreading positive information to counterbalance the diffusion of rumor. The spreading mechanism of rumors and effective suppression strategies are significant and challenging research issues. Firstly, in order to simulate the dissemination of multiple types of information, we propose a competitive linear threshold model with state transition (CLTST) to describe the spreading process of rumor and anti-rumor in the same network. Subsequently, we put forward a community-based rumor blocking (CRB) algorithm based on influence maximization theory in social networks. Its crucial step is to identify a set of influential seeds that propagate anti-rumor information to other nodes, which includes community detection, selection of candidate anti-rumor seeds and generation of anti-rumor seed set. Under the CLTST model, the CRB algorithm has been compared with six state-of-the-art algorithms on nine online social networks to verify the performance. Experimental results show that the proposed model can better reflect the process of rumor propagation, and review the propagation mechanism of rumor and anti-rumor in online social networks. Moreover, the proposed CRB algorithm has better performance in weakening the rumor dissemination ability, which can select anti-rumor seeds in networks more accurately and achieve better performance in influence spread, sensitivity analysis, seeds distribution and running time.

Keywords: online social networks, rumor blocking, competitive linear threshold model, influence maximization

PACS: 89.75.Fb **DOI:** 10.1088/1674-1056/ad531f

1. Introduction

With the growth of online social networks (OSNs), such as Facebook, Twitter and WeChat, social media has become an important platform for users to keep in touch with others, and stay up to date with breaking events and live news.[1–4] However, the rise and development of OSNs platform is also a double-edged sword. On the one hand, the platform provides an efficient and convenient channel for users to facilitate the information interaction. On the other hand, the platform reduces the cost of propagating rumor inadvertently, which in turn becomes the breeding ground for rumor.[5–8] Rumors are typically defined as inaccurate and misleading false statements about an event.[9–12] When users are exposed to new information and are unsure of its credibility, they may be easily activated by rumor and become new spreaders, which leads to the rapid propagation of rumors.[13–16] It is conceivable that the unrestrained spreading of rumor poses a huge threat to social stability, economic prosperity and public sentiment. The large-scale propagation of rumor may greatly endanger people's daily lives, thus it is urgent to search for an efficient strategy for limiting the rumor propagation in OSNs.

The spreading dynamics of rumor is similar to epidemic propagation to some extent. Considering network topological structure and characteristics of users, researchers have designed a series of models to simulate the rumor propagation process.[17–21] For example, He *et al.*[17] assumed that negative and positive information can spread in the same network, while the negative influence can be counteracted by positive one. Tan *et al.*[18] put forward an improved susceptible–infected–recovered (SIR) model for blocking misinformation propagation in OSNs, which is based on the assumption that nodes in I state can become S state through self-healing, and spreaders can change back to recovery state through human intervention. Sun *et al.*[19] established a comprehensive propagation model by considering some psychological factors of nodes, where both trust mechanism and correlation mechanism are fused to describe the dynamics of rumor on heterogeneous and homogenous networks. Manouchehri *et al.*[20] proposed an improved independent cascade (IC) model to simulate the information propagation in OSNs, they investigated the influence blocking problem by considering both deadline and time delay during rumor propagation process. Luo *et al.*[21] designed an improved SIR model with nonlinear incidence and time delay in heterogeneous networks, which fully considers intervention mechanism and multilingual environment in the process of rumor propagation.

In recent years, a series of solutions have been put forward from multiple aspects to solve the rumor blocking maximization (RBM) problem, which can be roughly divided into three categories.[22–27] The first one is to block specific

†Corresponding author. E-mail: xugq@staff.shu.edu.cn

© 2024 Chinese Physical Society and IOP Publishing Ltd

nodes.[22,23] This kind of methods tends to remove several influential nodes, which plays an important role in information spreading, so that the rumor propagation can be blocked as much as possible. The second one is to remove edge for rumor,[24,25] which selects a set of influential edges according to the designated criteria, and deletes these edges to ensure that the spreading of rumor is limited. The third one is to spread positive information for combating rumors,[26,27] whose core idea is to choose a subset of influential nodes as the anti-rumor, such that the positive information can be adopted by majority of users. In other words, if users adopt anti-rumor information, they will no longer be affected by the rumor, i.e., they will immune to rumor.

As stated above, although a number of propagation models (for simulating the spreading process of information) and anti-rumor seeds selection algorithms (for spreading truth) have been developed to solve the RBM problem, there are still some problems that need to be addressed. On the one hand, typical propagation models pay more attention on the activation process of nodes with only one state, while the rumor propagation process always exists competition between negative and positive information. A novel propagation model should be constructed to simulate the competitive spreading process between rumor and anti-rumor. On the other hand, most of the current anti-rumor seeds selection algorithms are compromising between computational complexity and result accuracy, while ignoring the local topological characteristics that the community structure brings, and the "rich-club" phenomenon among seed nodes.

In this work, we study the rumor blocking maximization problem and put forward a new algorithm to address the rumor propagation problem. First, we propose a novel propagation model called the competitive linear threshold model with state transition (CLTST) to examine the RBM problem. As an extension of the traditional LT model, CLTST describes the process of rumor and anti-rumor spreading in the network simultaneously. In CLTST, an inactive node will be activated when the combined affect of node's active in-neighbors is larger than its active threshold, and the activated nodes will transform theirs state when total influence from the active in-neighbors is changed. Then, under the framework of CLTST model, a new community-based rumor blocking algorithm (CRB) based on influence maximization theory is put forward, which aims to identify a set of influential seeds that propagate anti-rumor information to other nodes. The proposed CRB algorithm comprehensively considers the information of community structure, the information about node itself and the mutual interactions with its neighbors, which consists of three phases, community detection, selection of candidate anti-rumor seeds and generation of anti-rumor seed set. The major contributions of this paper are summarized as follows:

(1) Based on the LT model, we propose a CLTST propagation model to simulate the competitive spreading process of rumor and anti-rumor, which introduces active threshold and decision threshold to take effect in infecting inactive nodes or transforming the state of active nodes. In addition, each activated node will transform its state with the change of total influence from the active in-neighbors.

(2) We put forward the CRB algorithm to identify a set of influential nodes as the source of anti-rumor information, which utilizes the information of community structure to disperse seed distribution and reduce computation cost, and the candidate anti-rumor seeds are obtained by evaluating the nodes' influence in communities. After that, an adaptive search strategy is put forward to obtain the best anti-rumor seed set solution from the candidates.

(3) Under the proposed CLTST model, our CRB algorithm is compared with other six popular algorithms on nine online social networks, and the experimental results show that the performance of CRB is superior to those of the contrast methods.

The rest of this work is organized as follows: Section 2 briefly reviews the related researches. The RBM problem and CLTST model are introduced in Section 3. Section 4 illustrates our proposed community-based rumor blocking algorithm in detail. Next, some essential experimental setup, including the evaluation metrics, compared algorithms and dataset are listed in Section 5. Subsequently, the obtained results and related statistical analysis are shown in Section 6. Finally, we conclude the whole work in Section 7.

2. Related work

In this section, we survey the existing researches regarding rumor blocking problem, which can be roughly divided into three families: blocking nodes, removing edges and spreading positive information for rumor blocking.

2.1. Blocking nodes

The first family is blocking some influential nodes such that the diffusion of rumor in OSNs is minimized. Recently, numerous researches introduced static topological characteristics to evaluate the importance of nodes in the propagation process of rumor.[28–34] For example, Wu and Pan[28] constructed two competitive propagation models to simulate the propagation process of information, and a set of influential nodes with the largest contagious probability were blocked to address the influence blocking problem. Pham *et al.*[29] proposed a time-constraint deterministic LT model to investigate how to limit the spreading of rumor in OSNs, and a novel greedy algorithm

was put forward to find several nodes whose removal will minimize the spread of rumor. Taking into account the community structure of networks, Zheng and Pan[30] focused on the least cost rumor community blocking optimization problem, which guarantees that the number of activated nodes is less than a given positive integer when the process of rumor propagation is terminated. Yan and his coworkers[31] defined the minimizing rumor influence problem, and the activation probability of each user was minimized by blocking a set of influential nodes. Ding *et al.*[32] proposed a dynamic rumor propagation model to get a balance between the effectiveness of rumor blocking and computational cost, and a set of nodes were blocked to keep the propagation scale of rumor as low as possible. Yao *et al.*[33] classified users in a network into different groups and put forward five types of controlling measures for fast controlling of rumors. According to integer linear programming, Yang *et al.*[34] deduced a mathematical programming formulation to simplify the rumor blocking problem, which aims to minimize the propagation of rumor by blocking a subset of influential nodes.

2.2. Removing edges

The second category is removing some edges that takes effect in rumor propagation, which is an effective approach to minimize the negative impact of rumor. In other words, the RBM problem is usually equivalent to the problem of minimizing the influence of rumor, and the selection of influential edges is vital for designing the blocking strategy.[35–40] Based on the high-betweenness immunization sequence, Schneider *et al.*[35] developed an immunization approach to minimize the spreading of rumor, where the sum of the largest connected clusters in graphs is considered as node's spread susceptibility. Tong *et al.*[36] assumed that the eigenvalue of network matrix is an important indicator to evaluate the propagation susceptibility of nodes, and tended to identify a set of influential edges whose removal will minimize the eigenvalue of matrix. Yao *et al.*[37] designed a new algorithm for identifying a subset of edges to minimize the spread of misinformation, and an iterative greedy algorithm was proposed to add edges with the maximum marginal gain into edge seed set in each iteration. Dey and Roy[38] constructed a random walk model for rumor blocking in OSNs, which blocks a set of influential edges with the highest betweenness centrality to limit the propagation of rumor. Yan *et al.*[39] presented a greedy algorithm to address the problem of rumor spread minimization, and a set of edges with the maximum marginal gain in the graph were removed to block activated nodes as much as possible. Jiang *et al.*[40] constructed a general target information disseminating model to investigate the propagation paths of rumor, and introduced a random walk algorithm to block a budget-limited set of influential paths.

2.3. Spreading positive information

The third family aims to spread positive information to others, such that the negative influence of rumor will minimize. In recent years, several researches about positive influence maximization have been achieved.[41–47] Budak *et al.*[41] investigated the problem of two competitive events propagation in the same network, where a greedy approach is introduced to select a set of seeds for spreading positive influence to reduce the number of users who accept rumor as much as possible. Tong *et al.*[42] constructed an extended diffusion model to solve the multi-campaign spreading problem, which proposes a hybrid sampling method to obtain positive seed set, and considers that nodes more likely to be influenced by misinformation needs more attention. Yang *et al.*[43] designed a competitive model to simulate the propagation process of rumor and truth in the same network, the model assumed that the probability of users activated by positive message depends on various social factors, and users who adopt rumor may reverse their state as the state of neighbors altering. Xiao *et al.*[44] proposed a group behavior model for modeling the process of rumor and anti-rumor spreading in networks, which takes into account the complexity and diversity of rumor propagation feature space, and takes full advantage of representation learning in data feature extraction. Based on tensor completion, evolutionary game theory and sparse representation, Li *et al.*[45] put forward a game diffusion model to analyze the relationship between rumor and anti-rumor in diffusion process, and an improved graph convolutional network was presented to quantify the probability that user's behavior being activated. Xiang *et al.*[46] investigated the problem of rumor blocking with pertinence set, and proposed a novel hybrid greedy framework to find a set of truth seeds. He *et al.*[47] focused on the problem of positive influence maximization within a limited time, which devises a novel measure to select seed nodes by making full use of inverted index, forward index and cost delay.

The core of positive influence maximization is selecting a set of influential nodes as the source of anti-rumor. To address this problem, various algorithms have been put forward from different viewpoints.[48–53] Xie *et al.*[48] designed an adaptive degree-based heuristic algorithm to identify some influential nodes as seed spreaders, considering that nodes have more neighbors with the seed set unlikely to be selected as seeds. Yang *et al.*[49] developed a community-based influence maximization algorithm to choose several seed spreaders, they devised a recursive clustering approach to partition entire network into communities, and the topological potential "peak–slope–valley" structure was introduced to determine seeds.

Beni and his coworkers[50] developed the module-based influence maximization algorithm to select influential nodes, where graph modules were identified on basis of the information of node itself and neighbors. In the framework of the gravity centrality model, Xie and his coworkers[51] proposed a novel measure to determine a set of seed nodes in hypergraphs, where the influence of nodes is measured by incorporating the degree and the higher-order distance of nodes. Umrawal *et al.*[52] constructed a community-aware divide-and-conquer framework to address the influence maximization problem and proposed a novel progressive budgeting scheme to identify core nodes in each community. According to local topology structure and group trust, Guo *et al.*[53] developed a novel algorithm for addressing influence maximization problem, which fully considers the influence of group during the propagation of information, and the local structure information of different groups.

3. Model formation

Let $G = (V, E, w)$ be a directed network, where V and E represent the node set and the edge set, respectively. Let N and M represent the number of nodes and edges of G, respectively. If exists a direct edge from node i to node j, we decide that $(i, j) \in E$. Here $w_{i,j}(1 \leq i, j \leq N, i \neq j)$ represents the propagation probability to directed edge $(i, j) \in E$. If $(i, j) \notin E$, $w_{i,j} = 0$; otherwise, $w_{i,j} = 1/d_j^{\text{in}}$, where d_j^{in} is the number of in-degree neighbors of node j.

3.1. Rumor blocking maximization problem

The main idea of the RBM problem is to select some influential seed nodes for spreading anti-rumor information, so as to minimize the number of users activated by rumor. Here, S_{R} is the source of rumor spreading process, and S_{A} is the source of anti-rumor diffusion process. When the propagation process is terminated, $\sigma(S_{\text{R}}, S_{\text{A}})$ denotes the total number of nodes that are activated by S_{R} rather than S_{A}. Mathematically, it can be formulated as

$$\begin{cases} f(S) = \arg\min \sigma(S_{\text{R}}, S_{\text{A}}), \\ \text{subject to } |S_{\text{R}}| = |S_{\text{A}}| \leq k, \end{cases} \tag{1}$$

where $f(S)$ represents the minimum number of nodes that are activated by rumor rather than anti-rumor.

3.2. Competitive linear threshold model

In existing researches, most of the diffusion models assume that when one person is activated by one piece of information, he/she remains in that state forever and never changes his/her mind. This assumption may work well for the product marketing related to purchase behavior, once a purchase has been made, it is difficult to reverse the user's behavior. However, in the process of information propagation, new information gathered from various sources may prompt users to change their attitudes towards a news event.

Since threshold models are relatively appropriate to describe collective behaviors,[34] this work intends to improve linear threshold model to characterize the competitive propagation dynamics of rumor and anti-rumor in social networks. To do this, we propose a competitive linear threshold model with state transition (CLTST) to simulate the dissemination process of rumor and anti-rumor in the same network. In the CLTST model, nodes have three alternative states: R state (adopting rumor), A state (adopting anti-rumor), inactive (adopting neither rumor or anti-rumor) and one transient state: influenced (considering to adopt rumor or anti-rumor). The transition of four types of nodes is shown in Fig. 1, and the state transition rules are defined as follows: (1) For a user u in inactive state, if the influence from the activated in-neighbors is larger than or equal to its current active threshold (θ_u), this user transforms to influenced state. (2) For a user u in influenced state, if the influence from the activated R-state in-neighbors is larger than or equal to its current decision threshold (θ_u^{R}), this user transforms to R state; otherwise, this user transforms to A state. (3) For a user u in R state, if the influence from the activated R-state in-neighbors is smaller than its current decision threshold (θ_u^{R}), this user transforms to A state. (4) For a user u in A state, if the influence from the activated R-state in-neighbors is larger than or equal to its current decision threshold (θ_u^{R}), this user transforms to R state.

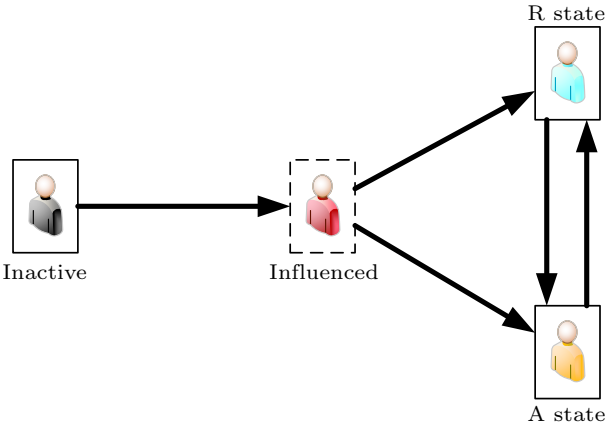


Fig. 1. The proposed competitive linear threshold model with state transition.

The specific idea of the CLTST model is as follows. Initially, S_{R} and S_{A} denote the source seed set of R state and A state, respectively. Moreover, nodes that not belong to S_{R} and S_{A} are set to inactive state. At each timestamp, if the influence from the activated in-neighbors is larger than or equal to its current active threshold, an inactive node u will be activated to participate into the information propagation process. In other words, an inactive node u transforms to influenced state when

it satisfies the following condition:

$$\sum_{v\in\Gamma^{\rm in}(u)} w_{v,u}\cdot\beta^t(u)\geq\theta^t(u), \tag{2}$$

where $\Gamma^{\rm in}(u)$ denotes the in-neighbor set of node u; $\beta^t(u)$ represents the active probability that node u is activated at time t, which depends on the influence of node u's activated in-neighbors. The value of $\beta^t(u)$ is calculated by

$$\beta^t(u)=1-\prod_{v\in\Gamma^{\rm in}(u)}[(1-w_{v,u}\cdot x_v(t))\cdot(1-x_u(t))], \tag{3}$$

where $x_v(t)=1$ means that node v is activated, and $x_v(t)=0$ means that node v is inactive. In Eq. (2), $\theta^t(u)$ is the active threshold of node u at t timestamp, which is calculated by

$$\theta^t(u)=\frac{|\Gamma_{t-1}|}{N}, \tag{4}$$

with the current active threshold $\theta^t(u)$ of node u being defined as the ratio of active nodes at $t-1$ timestamp.

A node u will become R state or A state when it is activated. If the total influence from the active R-state in-neighbors is larger than or equal to its current decision threshold, activated node u will adopt rumor and aggressively spread. Mathematically, it can be formulated as

$$\frac{\sum_{v\in\{\Gamma^{\rm in}(u)\cap\Gamma^{\rm R}_{t-1}\}}\beta^t(v)\cdot w_{v,u}}{\sum_{v\in\{\Gamma^{\rm in}(u)\cap\Gamma_{t-1}\}}\beta^t(v)\cdot w_{v,u}}\geq\theta^t_{\rm R}(u), \tag{5}$$

$$\theta^t_{\rm R}(u)=\frac{|\Gamma^{\rm R}_{t-1}|}{|\Gamma_{t-1}|}, \tag{6}$$

where Γ_{t-1} and $\Gamma^{\rm R}_{t-1}$ represent the set of active nodes and active R-state nodes at $(t-1)$ timestamp, respectively. Similarly, the current decision threshold $\theta^t_{\rm R}(u)$ of node u is defined as the ratio of active R-state nodes at $t-1$ timestamp. On the contrary, if node u does not satisfy the above inequality, it will transform to A state.

Furthermore, in most of the existing competitive diffusion models, once a node is activated by one type of information, the activated node does not change until the propagation process is terminated. However, people's perception towards an event will change according to new information they gathered, and users who spread rumor will also transform their stance after receiving positive information.[43] Inspired by this, we allow that active nodes will transform their states with the change of total influence of its active in-neighbors. If the influence from the activated R-state in-neighbors of node u is larger than or equal to its current decision threshold $\theta^t_{\rm R}(u)$, the active R-state node u will remain its state. Otherwise, node u will become A state. Similarly, the active A-state nodes also undergo the state transforms when the total influence from its active in-neighbors is change. When no more nodes in the network can be activated and the state of nodes remains stable, the propagation process will be terminated.

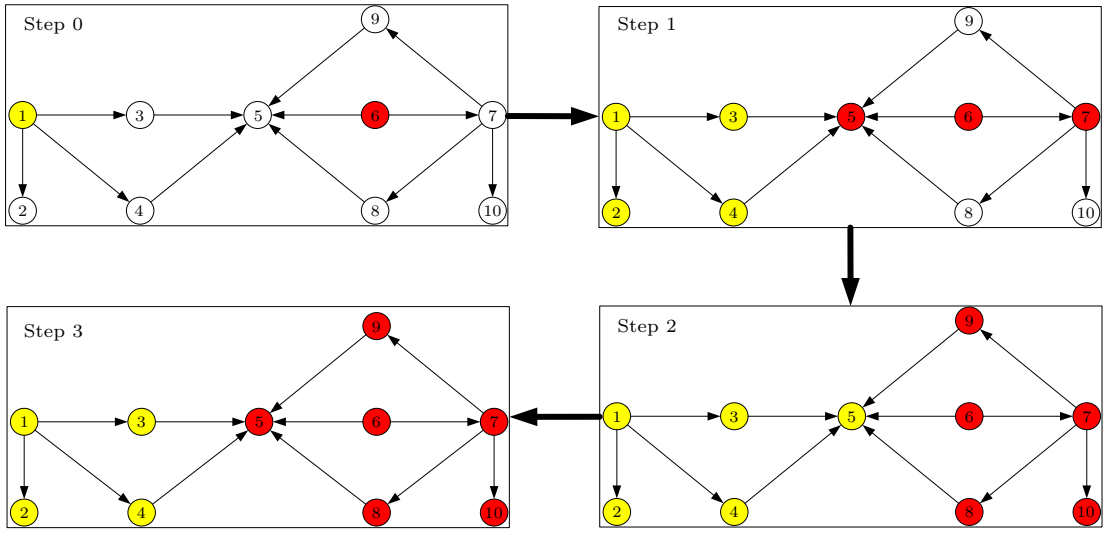


Fig. 2. A simple network with ten nodes for illustrating the CLTST model.

In the proposed CLTST model, the states of nodes are divided into more detailed parts, and two parameters called active threshold and decision threshold are introduced to take effect in infecting inactive nodes or transforming the state of active nodes, so as to fit the real situation of rumor propagation process as much as possible. Compared with the existing researches, the proposed model can portray the interaction mode of nodes and the mechanism of rumor propagation more accurately. To illustrate the proposed CLTST model more clearly, Fig. 2 gives an example for explaining the state transition process of nodes. Initially, it is assumed that $S_{\rm R}=1$ and $S_{\rm A}=6$. In time step 1, the active threshold ($\theta^1(u)=1/5$) and decision

threshold ($\theta_{\mathrm{R}}^{1}(u) = 1/2$) for each node are calculated according to Eqs. (4) and (6). The active probability for nodes 2, 3, 4, 5, 7 are calculated as Eq. (3), and the result is $\beta^{1}(2) = \beta^{1}(3) = \beta^{1}(4) = \beta^{1}(7) = 1, \beta^{1}(5) = 1/5$. According to the state transfer rules, nodes 2, 3 and 4 transform to R state, and nodes 5 and 7 transform to A state. In time step 2, $\theta^{2}(u) = 7/10$, $\theta_{\mathrm{R}}^{2}(u) = 4/7$, $\beta^{2}(5) = \beta^{2}(7) = \beta^{2}(8) = \beta^{1}(4) = \beta^{2}(10) = 1$. Nodes 7, 8 and 10 transform to A state since these three nodes does not satisfy Eq. (5). For node 5, the total influence from the active in-neighbors is greater than its current active threshold, so node 5 transforms to R state. In time step 3, the total influence from the active in-neighbors of node 5 has changed, and is smaller than its current active threshold ($\theta_{\mathrm{R}}^{3}(u) = 1/2$), so node 5 transforms to A state again. Meanwhile, the propagation process is terminated.

4. Rumor blocking maximization algorithm

As mentioned above, a number of algorithms have been established for blocking rumor diffusion. However, there are still some problems that need to be addressed. On the one hand, due to the high computational cost, greedy based algorithms are easily constrained in large-scale networks.[54,55] On the other hand, community detection is beneficial in reducing the computational cost of algorithms and dispersing the selected seeds, most of the existing research works neglect the community structure information of networks in selecting anti-rumor seed set.[56,57]

To address these problems, this work puts forward a new community-based rumor blocking algorithm (CRB) based on influence maximization theory. The core step of our algorithm is to select a set of influential seeds spreading anti-rumor information to other nodes, which consists of three phases: (1) community detection, (2) selection of candidate anti-rumor seeds, (3) generation of anti-rumor seed set. In order to achieve rapid dissemination of information, the distribution of anti-rumor seed nodes in the network should be relatively dispersed, for this purpose we first perform community detection by employing the classic Louvain algorithm. After the initial community division, a number of small-scale communities are merged for reducing computational cost and accelerating information propagation. Subsequently, to ensure that the selected candidate anti-rumor seeds are fully dispersed in the network, the number of candidate anti-rumor seeds in each community is determined by community size and the closeness of network. Finally, based on the influence of the node itself and the interaction information between nodes and its neighboring node set, an adaptive search strategy is employed to yield a satisfactory solution within a reasonable time and avoids premature convergence. Details of the CRB algorithm are illustrated in the following subsections, and all the key parameters and symbols are presented in Table 1.

Table 1. Notations.

Parameter	Description
G	An unweighted and directed network
V	Set of nodes in G
E	Set of edges in G
N	Number of nodes in G
M	Number of edges in G
$w_{i,j}$	The propagation probability between nodes i and j
d_j^{in}	The number of in-degree neighbors of node j
S_{R}	Rumor seed set
S_{A}	Anti-rumor seed set
$\beta^{t}(u)$	The active probability that node u is activated at time t
$\theta^{t}(u)$	The active threshold of node u at time t
$\theta_{\mathrm{R}}^{t}(u)$	The decision threshold of node u at time t
k	Number of source seeds
W_{C_i}	The community weight of community C_i
θ_{c}	The community influence threshold
$C = \{C_1, C_2, \ldots, C_p\}$	Initial community structure
$C_M = \{C_1, C_2, \ldots, C_{p_1}\}$	Optimized community structure
CN_{C_i}	The number of candidate anti-rumor seeds at community C_i
PLC_x	The potential influence score of node x
CSS	Optimized candidate anti-rumor seed set
$\mathrm{IS} = \{X_1, X_2, \ldots, X_k\}$	Initial anti-rumor candidate solution
CS	Current solution after each iteration (In the first iteration, CS = IS)
DEDV(CS)	The DEDV fitness function value of current solution CS
SA	Search area consisting of nodes in the CSS that do not belong to the CS
S'	A newly generated solution by selecting a seed in CS to randomly replace one node in SA
SS_{A}	Anti-rumor seed set obtained by the proposed algorithm

4.1. Community detection

The inherent characteristics of complex networks complicate the task about selecting some influential nodes as seeds in an entire network. As is well known, one of the best advantages of Louvain algorithm is its high efficiency.[58] Moreover, Louvain algorithm requires no prior knowledge on the community number. Inspired by this, we employ the Louvain algorithm to divide online social networks into a number of small communities. After the community detection process, those small-scale insignificant communities with few nodes and edges should be merged into large-scale communities. This is due to the fact that nodes in small communities have limited influence and they cannot spread information widely across multiple communities. The influence of each community depends on not only the size and connection strength of communities, but also the mutual attraction and information transmutation between communities. For a given network, the community structure $C = \{C_1, C_2, \ldots, C_p\}$ is obtained by employing the Louvain algorithm, where C_i represents the i-th community and p is the number of communities. The community weight is computed as follows:

$$W_{C_i} = -\frac{\lambda}{\ln |C_i|} \cdot \frac{\sum_{i,j \in C_i} d_{ij}}{2|C_i|(|C_i|-1)}, \tag{7}$$

$$\lambda = \sum_{x=1}^{|C_i|} d_x^{\text{in}} \cdot \ln d_x^{\text{in}} + d_x^{\text{out}} \cdot \ln d_x^{\text{out}}, \tag{8}$$

where $|C_i|$ is the number of nodes in community C_i, x represents the one of nodes in community C_i; d_x^{in} and d_x^{out} denote the number of in-degree neighbors and out-degree neighbors of node x, respectively; d_{ij} is the shortest distance between nodes i and j, while these two nodes are in the same community C_i.

Then, we define the community influence threshold (θ_c) to determine which communities should be merged. If the influence of community C_i is greater than or equal to θ_c, C_i is considered as a strong community and is suitable to explore influential nodes as seeds at a later step. On the contrary, when the influence of community C_i is less than θ_c, C_i is considered as a weak community and fails to go to the next step. Next, several weak communities with the closest average distance are merged into one large community, until the influence of combined community is greater than θ_c. If there exists only one weak community or the influence of combined community is smaller than θ_c, the weak community or the combined community is designated to one of the strong communities randomly. The community influence threshold θ_c is calculated as follows:

$$\theta_c = \text{Den}_{\max} \cdot \frac{|C|_{\min}}{|C|_{\max} + k + p}, \tag{9}$$

where $\text{Den}_{\max}$ is the largest density of all involved communities.[59] $|C|_{\max}$ and $|C|_{\min}$ represent the scale of the largest and smallest communities, respectively; k stands for the number of anti-rumor seeds in the network; p denotes the number of communities. After merging communities, we can obtain the optimized community structure $C_M = \{C_1, C_2, \ldots, C_{p_1}\}$, where p_1 is the number of communities and $p_1 \leq p$.

To illustrate the above community detection process, a simple network with 22 nodes and 33 edges is chosen as an example. As shown in Fig. 3, the Louvain algorithm is introduced to partition the entire network into 5 communities, and the weight of each community is $W_{C_1} = 0.0595$, $W_{C_2} = 0.0521$, $W_{C_3} = 0.0894$, $W_{C_4} = 0.1222$, and $W_{C_5} = 0.0834$. If the seed size $k = 2$, the community threshold of the simple network is $\theta_c = 0.0714$. By comparing the weight of each community and θ_c criteria, communities 3, 4 and 5 can be considered as a strong communities since W_{C_3}, W_{C_4} and W_{C_5} are greater than θ_c. On the contrary, communities 1 and 2 are considered as a weak community because the weight of these two communities are lower than θ_c. Then, communities 1 and 2 are merged into a novel community, and the weight of the new combined community is $W_{C_1}^{\text{new}} = 0.0992$, which is larger than θ_c. Therefore, the community detection and merging process is completed, and the optimized community structure of the simple network is return.

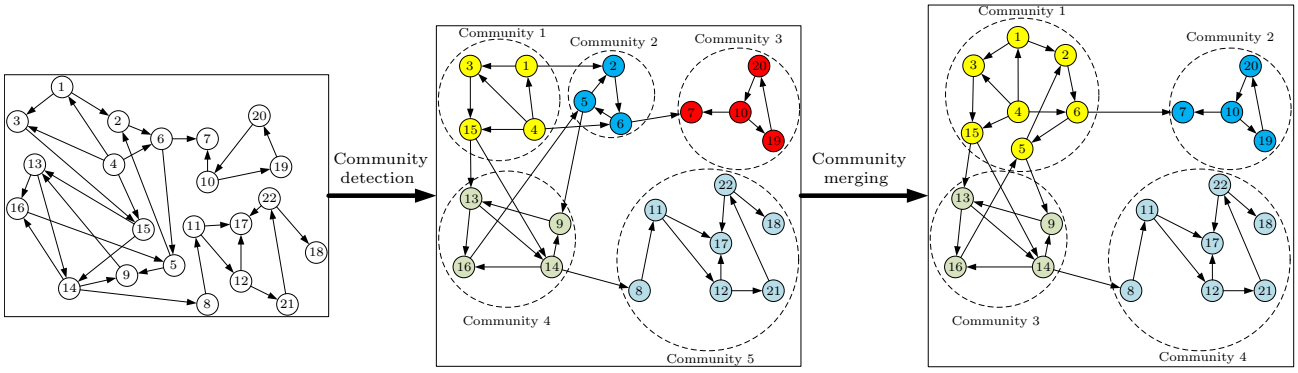


Fig. 3. An example of community detection.

Algorithm 1. Community detection

Input: Network $G=(V,E,w)$

1. Apply Louvain algorithm to obtain the initial community structure $C=\{C_1,C_2,\ldots,C_p\}$
2. Generate weak communities $C_W=\{C_{W1},C_{W2},\ldots,C_{Wx}\}$ based on community closeness
3. Set $j\leftarrow 1$
4. **while** $C_W\neq\emptyset$ **do**
5. Search for the nearest weak community C_{Wk} of C_{Wj}
6. $C'_{Wk}\leftarrow C_{Wk}\cup C_{Wj}$
7. **if** $W'_{C_{Wj}}\geq\theta_c$ **then**
8. $C_W\leftarrow\{C_W-C_{Wk}-C_{Wj}\}\cup C'_{Wk}$
9. **end if**
10. $j\leftarrow j+1$
11. **end while**
12. **return** Optimized community structure $C_M=\{C_1,C_2,\ldots,C_{p_1}\}$

The pseudo-code of the community detection is presented in Algorithm 1. Line 1 divides the entire network into several communities according to the conventional Louvain algorithm, and line 2 generates weak communities according to the closeness of each community. Lines 3–11 merge several weak communities as a novel strong community. To do this, each weak community tries to merge another weak community with the smallest average distance, and the weight of the combined community is compared with the community threshold of network. The process is terminated until the weight of combined community is larger than the community threshold of network. Line 12 returns the optimized community structure.

4.2. Selection of candidate anti-rumor seeds

Influential nodes in OSNs as seed nodes for spreading truth information can help to suppress the rapid dissemination of rumors. The aim of this stage is to derive the candidate anti-rumor seeds based on the community division results. To achieve this, we first determine the number of candidate anti-rumor seeds in each community according to the closeness of network and size of community. Then we introduce the potential local centrality to assess the node's influence by considering the information of node itself and its neighbors.

4.2.1. Determining candidate anti-rumor seeds number

From the perspective of information propagation and resource allocation, it is unreasonable to allocate the same number of anti-rumor seeds in communities of different sizes. Therefore, before yielding candidate anti-rumor seed set, we calculate the number of candidate anti-rumor seeds in each community. Specifically, the upper bound of candidate anti-rumor seed set is calculated by the closeness of networks, and the size of each community determines the distribution of candidate anti-rumor seeds. Mathematically, the number of candidate anti-rumor seeds at each community is calculated by

$$\mathrm{CN}_{C_i}=\left\lceil \exp\left(\frac{2M}{N(N-1)}\right)\cdot k\cdot\frac{|C_i|}{\sum_{i=1}^{p_1}|C_i|}\right\rceil, \tag{10}$$

where CN_{C_i} is the number of candidate anti-rumor seeds in community C_i. In Eq. (10), the number of candidate anti-rumor seeds in each community is determined according to three main aspects: $\exp(2M/N(N-1))$ is an adjust parameter to determine the selected number of candidate anti-rumor seeds in each community, k represents the number of anti-rumor seeds in the network, and $\frac{|C_i|}{\sum_{i=1}^{p_1}|C_i|}$ controls the distribution of candidate anti-rumor seeds. Taking Fig. 3 as an example, the number of candidate anti-rumor seeds at each community is $\mathrm{CN}_{C_1}=\mathrm{CN}_{C_4}=\lceil 0.7341\rceil=1$, $\mathrm{CN}_{C_2}=\mathrm{CN}_{C_3}=\lceil 0.4195\rceil=1$. In other words, we select one node as candidate anti-rumor seed in each community.

4.2.2. Selecting candidate anti-rumor seeds

For each community, the candidate anti-rumor seeds has faster and broader influence propagation than other general nodes.[60–62] The higher the propagation capability, the larger probability that a node will be selected as a candidate anti-rumor seed. Inspired by three degree of separation[63,64] and community structure in networks, we define the potential local centrality (PLC) for calculating the propagation capability of nodes in each community. The proposed algorithm comprehensively considers the information about the source node itself, its nearest neighbors and the next nearest neighbors, as well as the mutual interactions between them. Concretely, the potential influence score of each node can be calculated by

$$Q_u=d^{\mathrm{in}}_{C_\zeta}(u)\cdot\sum_{t\in\Gamma^{\mathrm{in}}(u)}d_t\cdot w_{u,t}+d^{\mathrm{out}}_{C_\zeta}(u)\cdot\sum_{v\in\Gamma^{\mathrm{out}}(u)}d_v\cdot w_{u,v}, \tag{11}$$

$$\mathrm{PLC}_x=d^{\mathrm{in}}_{C_\zeta}(x)\cdot\sum_{u\in\Gamma^{\mathrm{in}}(x)}Q_u\cdot w_{x,u}+d^{\mathrm{out}}_{C_\zeta}(x)\cdot\sum_{j\in\Gamma^{\mathrm{out}}(x)}Q_j\cdot w_{x,j}, \tag{12}$$

where nodes u, t, v, x and j are in the same community C_ζ, and C_ζ is a community in optimized community structure C_M; $d^{\mathrm{in}}_{C_\zeta}(u)$ and $d^{\mathrm{out}}_{C_\zeta}(u)$ represent the number of in-neighbors and out-neighbors of node u in community C_ζ, respectively; $\Gamma^{\mathrm{in}}(u)$ and $\Gamma^{\mathrm{out}}(u)$ denote the in-neighbor and out-neighbor

set of node u, respectively; d_t denotes the number of neighbors of node t in the network. The node with larger PLC score is more likely to affect others within the same community, thus we select the top-CN_{C_i} nodes in community C_i in descending order of the PLC values, and add these nodes into candidate anti-rumor seed set. In the case of multiple nodes with the same PLC value, nodes with the largest number of neighbors are chosen as candidate anti-rumor seeds.

In order to illustrate the process of selecting candidate anti-rumor seeds, we also choose the example network mentioned in Fig. 3. In community 1, the potential influence scores of all nodes are $\{\mathrm{PLC}_1 = 6,\ \mathrm{PLC}_2 = 4,\ \mathrm{PLC}_3 = 3,\ \mathrm{PLC}_4 = 15,\ \mathrm{PLC}_5 = 4,\ \mathrm{PLC}_6 = 4,\ \mathrm{PLC}_{15} = 0\}$. Since $\mathrm{CN}_{C_i} = 1$, node 4 is selected as the candidate anti-rumor seed in community 1. In a similar way, the largest PLC values of communities are 10, 13, and 11, respectively. In summary, the candidate anti-rumor seed set in Fig. 3 is $\mathrm{CSS} = \{4, 10, 11, 13\}$.

The outline of candidate anti-rumor seeds selection is presented in Algorithm 2. Lines 1–3 calculate the potential influence score of nodes in the network, which considers the information of position, neighborhood and topology structure of nodes in the community. Lines 4–10 generate the candidate anti-rumor seed set. In order to do so, nodes in each community should be sorted by the potential influence score in descending order, then the most influential nodes in the community are selected as candidate anti-rumor seeds. Line 11 returns the candidate anti-rumor seed set CSS.

Algorithm 2. Selection of candidate anti-rumor seeds

Input: Community structure $C_M = \{C_1, C_2, \ldots, C_{p_1}\}$
1. **for** $i = 1$ *to* N **do**
2. Calculate the PLC value of node i using Eq. (12)
3. **end for**
4. Set CSS $\leftarrow \emptyset$
5. **for** $i = 1$ *to* p_1 **do**
6. Rank nodes in community C_i in descending order of the PLC value
7. Calculate CN_i in community C_i according to Eq. (10)
8. Select top-CN_i nodes in community C_i to generate set CSS_i
9. CSS $\leftarrow$ CSS $\bigcup$ CSS_i
10. **end for**
11. **return** Candidate seed set CSS

4.3. Generation of anti-rumor seed set

The goal of the proposed CRB algorithm is to find the k nodes in the network that have the largest final sphere of influence so that these k nodes act as the initial active nodes to spread truth information, resulting in the maximum number of nodes being influenced. According to the previous two phases, we obtain the candidate anti-rumor seed set for a given network. In this subsection, an adaptive optimized solution is proposed to obtain the anti-rumor seed set, which can not only maximize the total influence spreading sphere, but also minimize the overlapping influence of the seeds. The generation of the seed node set can be proceeded as follows.

4.3.1. Initializing anti-rumor seed set

Considering that the process of self-convergence speed depends on the selection of the initial anti-rumor seed solution, a high quality solution is constructed to improve the efficiency of the algorithm. We select the top-k nodes in descending order of their PLC scores, rather than randomly choose nodes with the highest degree. To be more specific, we construct an initial anti-rumor seed solution $\mathrm{IS} = \{X_1, X_2, \ldots, X_k\}$ among the candidate anti-rumor seed set CSS, where $X_i \geq X_{i+1}$ in the value of PLC.

4.3.2. Estimating expected diffusion value

To evaluate the influence spread size of each solution, the existing greedy algorithm and its various extensions tend to take the average value through extensive repeated computation, which accompanies high computational cost in large-scale datasets. Very recently, a fitness function called expected diffusion value (EDV) is put forward by Jiang and his coworkers,[65] which can be considered as the each solution's expected diffusion capability. On this basis, we propose a novel fitness function called directed expected diffusion value (DEDV) to estimate the spreading capability of each solution in directed network, which combines the propagation probability between pairs of nodes with the diffusion value within one hop. The aim of DEDV is to evaluate how many neighbors will be activated by seeds, which can be formulated as

$$\mathrm{DEDV}(S) = k + \sum_{i \in S, j \in N_1^{\mathrm{out}}(S)/S} [1 - (1 - w_{i,j})^{\delta(j)}], \tag{13}$$

where k is the scale of anti-rumor seed set S, $N_1^{\mathrm{out}}(S)/S$ represents the set of out-neighbors of set S, excluding the nodes which belong to S; $\delta(j)$ refers to the number of edges between set S and node j.

4.3.3. Optimizing anti-rumor seed set

The solution evaluation process is conducted through a local search technique. In the first solution, the current solution (CS) is consistent with the initial solution ($\mathrm{CS} = \mathrm{IS}$). The search area $\mathrm{SA} = \{\mathrm{CSS} - \mathrm{CS}\}$. In each iteration, a new solution S' is obtained by replacing a node in solution CS from the search area SA, and the interchange between nodes is performed arbitrarily. The directed expected diffusion value of solution CS and S' can be computed as

$$\mathrm{Gain}(S'|\mathrm{CS}) = \mathrm{DEDV}(S') - \mathrm{DEDV}(\mathrm{CS}). \tag{14}$$

The new solution S' will be accepted if the value of diffusion gain is positive. On the contrary, the current solution survives to go to the competition in the next iteration. For finding a better solution, the iteration process will execute

$\lceil \log(|S_A|) \cdot k \rceil$ times, which ensures that seeds in S will replace multiple times.

The pseudo-code of anti-rumor seed nodes generation is presented in Algorithm 3. Lines 1–3 obtain the initial anti-rumor seed node solution, which consists of top-k nodes selected by sorting candidate anti-rumor seed set according to the PLC value in descending order. On the basis of the DEDV strategy, lines 4–12 generate the optimize anti-rumor seed set solution by iterating multiple operations. Specifically, line 5 creates a novel solution S' by arbitrarily replacing a node in CS from SA. Then, the expected diffusion values of the CS and S' are calculated in line 6. Lines 7–11 calculate the diffusion gain resulted from the new solution, and determine which solution will go to the competition in the next iteration. The iteration will execute $\lceil \log(|S_A|) \cdot k \rceil$ times until the anti-rumor seed set with the broadest and fastest capability of information propagation is found. Finally, the best anti-rumor seed set solution is yielded in lines 13 and 14.

Algorithm 3. Generation of anti-rumor seed set

Input: Candidate seed set CSS, number of seeds k

1. Sort CSS by the PLC value in descending order
2. Select top-k nodes in CSS set to generate initial solution IS
3. Set CS = IS, SA = {CSS − CS}, $S \leftarrow \emptyset$, iteration $\leftarrow 1$
4. **while** iteration $\leq \lceil \log(|S_A|) \cdot k \rceil$ **do**
5. Replace a node in CS from SA arbitrarily to generate a new solution S'
6. Compute DEDV(CS) and DEDV(S') of current solution CS and new solution S' according to Eq. (13)
7. Gain(S'|CS) = DEDV(S') − DEDV(CS)
8. **if** $Gain(S'|CS) > 0$ **then**
9. CS $\leftarrow S'$
10. **end if**
11. iteration = iteration + 1
12. **end while**
13. $SS_A \leftarrow$ CS
14. Return anti-rumor seed set SS_A

4.4. Computational complexity analysis

For the proposed CRB algorithm, the computational complexity is separately calculated in three phases. In Algorithm 1, the time cost of Louvain algorithm is $O(N \cdot \log N)$ and generating the weak community set require $O(p)$ time, respectively. Here, p is the number of communities in the network obtained by Louvain algorithm. The time complexity of merging the weak communities is $O(x^2)$. Thus, community detection requires $O(N \cdot \log N + p + x^2)$ time. In Algorithm 2, the time cost of evaluating the potential influence score of nodes is $O(N \cdot \langle k \rangle^2)$, where $\langle k \rangle$ is the average degree of network. Selecting anti-rumor candidate seeds in each community require $O(p_1)$ time. Here, p_1 is the number of optimized communities. Therefore, the time complexity of Algorithm 2 is $O(N \cdot \langle k \rangle^2 + p_1)$. In Algorithm 3, forming initial anti-rumor seed solution requires $O(N)$ time. In lines 4–12, the time complexity of searching for optimal anti-rumor seed set through multiple iterations is $\lceil \log(|\mathrm{SA}|) \cdot k \rceil \cdot \langle k \rangle$, where |SA| is the search area and k is the anti-rumor seed size. Thus the anti-rumor seed set generation process needs $O(N + \lceil \log(|\mathrm{SA}|) \cdot k \rceil \cdot \langle k \rangle)$ time. In summary, the overall computational complexity of CRB is $O(N \cdot \log N + p + x^2 + N \cdot \langle k \rangle^2 + p_1 + N + \lceil \log(|\mathrm{SA}|) \cdot k \rceil \cdot \langle k \rangle)$. Since p, x, p_1, SA, k and $\langle k \rangle$ are much lower than N, thus the total time complexity of CRB can be simplified to $O(N \cdot (\log N + \langle k \rangle^2))$.

Table 2. The basic statistical features of nine networks with N nodes and M edges. AOD and MOD represent the average out-degree and max out-degree of the network, respectively. D is the diameter of the network.

Network	N	M	AOD	MOD	D
Trust	4658	40133	8.6159	100	9
Gnutella	6301	20777	3.2974	48	20
DBLP	12590	49759	3.9523	617	20
Chicago	12982	39018	3.0055	7	111
Google	15763	171206	10.8613	852	17
Cora	23166	91500	3.9498	104	48
Twitter	23370	33101	1.4164	238	16
Linux	30837	213954	6.9382	243	19
Epinions	75879	508837	6.7059	1801	16

5. Experimental setup

To illustrate the effectiveness of the proposed algorithm in solving the RBM problem, several groups of experiments are performed on nine online social datasets. Trust is the trust network between users of the site http://dvd.ciao.co.uk/, Gnutella is a social network among users of different Gnutella hosts, DBLP is a collaboration network in the field of scientific publications, Chicago is a co-authorship network of users in Chicago region, Google is the hyperlink network from pages within Google's own site http://www.google.com/, Cora is a collaboration network of scientific researchers, Twitter is the directed social network contains Twitter users' information, Linux is the open source collaboration network for Linux source files in version 3.16, Epinions is an on-

line social network extracted from the consumer review site http://www.Epinions.com. The basic statistical features of nine soical datasets are listed in Table 2.

5.1. Datasets

5.2. Benchmark algorithms

To evaluate the performance of the proposed algorithm, we choose six state-of-the-art rumor blocking algorithms as comparison methods, which are described as follows. All algorithms are run on a PC platform with Windows 10, 1 CPU (Intel Core i7-1165G7, 2.8 GHz) and 16 GB memory, and the software platform is Matlab R2019a and Pycharm 2022.2.4.

- HC:[66] A novel clustering algorithm, which classifies nodes into different clusters according to the correlation between nodes pairs, and the center node in each cluster are selected to construct seed set.
- HGD:[67] An optimal clustering algorithm that divides the entire network into different groups by deleting some nodes with few neighbors, and the most influential spreaders is selected according to the descending order of each node's degree.
- LKG:[68] A scalable and fast approach to detect top-k influential nodes, which divides the entire network into communities by using the conventional Louvain algorithm, and employs k-shell decomposition algorithm to identify the seeds in communities.
- PageRank:[69] A traditional algorithm for Google's web page ranking, and it is extended to find influential seed nodes in complex networks. In this study, the damping factor is set to 0.85.
- Random:[13] A baseline method that selects the anti-rumor seeds randomly.
- Unblocking:[14] A special case in which there is no positive cascade.

5.3. Evaluation metrics

Three different measures, including influence spread, the ratio of rumor seeds and the ratio of anti-rumor seeds, are employed to evaluate the performance of the proposed algorithm.

(i) Influence spread: This measure represents the number of nodes in the network that has been activated by rumor at time t. In this experiment, the ratio of these two kinds of nodes is fixed as $\lceil 0.01 \cdot N \rceil$.

(ii) The ratio of rumor seeds: The ratio of rumor seeds is used to test whether the anti-rumor seeds selected by the strategies can play a controlling role in the process of rumor propagation. The ratio of rumor seeds is fixed as $0.01 \cdot N$ and the budget of anti-rumor seeds is selected range from $[\lceil 0.001 \cdot N \rceil, \lceil 0.01 \cdot N \rceil]$.

(iii) The ratio of anti-rumor seeds: It adjusts the ratio of the anti-rumor seeds to verify whether the blocking node selected by different strategies can control the spread of rumor in the network. The ratio of anti-rumor seeds is fixed as $0.01 \cdot N$ and the budget of rumor seeds is selected range from $[\lceil 0.001 \cdot N \rceil, \lceil 0.01 \cdot N \rceil]$.

In the following experiments, 200 rumor seeds are selected randomly, and the anti-rumor seeds are determined by the strategy described as mentioned above. The experimental results show the average number of activated nodes by rumor seeds when the spreading process is terminated.

6. Experimental results and analysis

In this section, four groups of experiments are conducted for comparing the proposed CRB algorithm and other contrast methods. In experiment I, we compare the number of activated nodes with the change of time on nine datasets. In experiment Ⅱ, we evaluate the number of final activated nodes for the proposed CRB algorithm and six compared algorithms by varying the percentage of rumor and anti-rumor seeds. In experiment Ⅲ, we investigate the dispersion degree of anti-rumor seeds identified by different algorithms. In the last experiment, the running time of the proposed CRB algorithm is compared with the other six contrast algorithms.

6.1. Rumor spread

In the first set of experiments, we compare the rumor spread over time obtained by the seven algorithms. On each dataset, the number of rumor and anti-rumor seeds are fixed as the one percent of the dataset size. Figure 4 shows the experimental results of the number of R-state nodes on nine datasets for different algorithms. This set of experiment is to assess the rumor blocking capability of seven algorithms with a fixed ratio of seeds. A lower number of R-state nodes indicates a higher effect of rumor suppression by the anti-rumor seeds. As observed in Fig. 4, compared with other six algorithms, the number of nodes blocked by the proposed CRB algorithm is much larger in most of the datasets. Specifically, in four datasets including Cora, Google, Chicago and Gnutella, the number of R-state nodes activated by the proposed CRB is obviously lower than other compared algorithms through the entire spreading process. In other words, the anti-rumor seeds obtained by the proposed CRB has better performance in blocking rumor propagation. In Trust and Epinions datasets, the performance of CRB is similar to those of HGD and PageRank algorithms during the whole spreading process. In DBLP, Twitter and Linux datasets, CRB is close to HGD algorithm, and these two algorithms outperform other five meth-

ods. To sum up, the proposed CRB algorithm can select anti-rumor seeds in networks more accurately and it achieves better performance in suppressing rumor dissemination.

6.2. Sensitivity analysis

To further evaluate the effectiveness of the proposed CRB, we conduct two groups of experiments to perform the sensitivity analysis, where the activated node numbers of seven algorithms are compared by varying the percentage of rumor and anti-rumor seeds.

Firstly, we focus on the performances of the proposed CRB algorithm and six contrast algorithms when the proportion of rumor seeds varies. In this experiment, the number of rumor seeds ($|S_R|$) varies in a certain interval to better depict the performance variation, where $|S_R| = \lceil q \cdot N \rceil$ ($q \in [0, 10]$), and the number of anti-rumor seeds is fixed as $|S_A| = \lceil 0.01 \cdot N \rceil$. Figure 5 presents the number of nodes activated by rumor set under different rumor seeds proportions. Obviously, the number of R-state nodes of these seven algorithms is growing continuously as the proportion of rumor seeds increases. As depicted in Fig. 5, the rumor blocking performance of the proposed CRB algorithm always ranks first under different rumor seeds proportions. More specifically, in Gnutella, Chicago, Google and Cora datasets, the proposed CRB algorithm consistently performs better than other six compared algorithms under different rumor seeds proportions. In DBLP, Twitter and Linux datasets, CRB algorithm obtains a low rumor propagation scale similar to that of HGD algorithm, and these two algorithms outperform other five algorithms for all chosen proportions. In addition, in terms of blocking the spreading of rumor, the effectiveness of CRB, HGD and PageRank are very close in Trust and Epinions datasets.

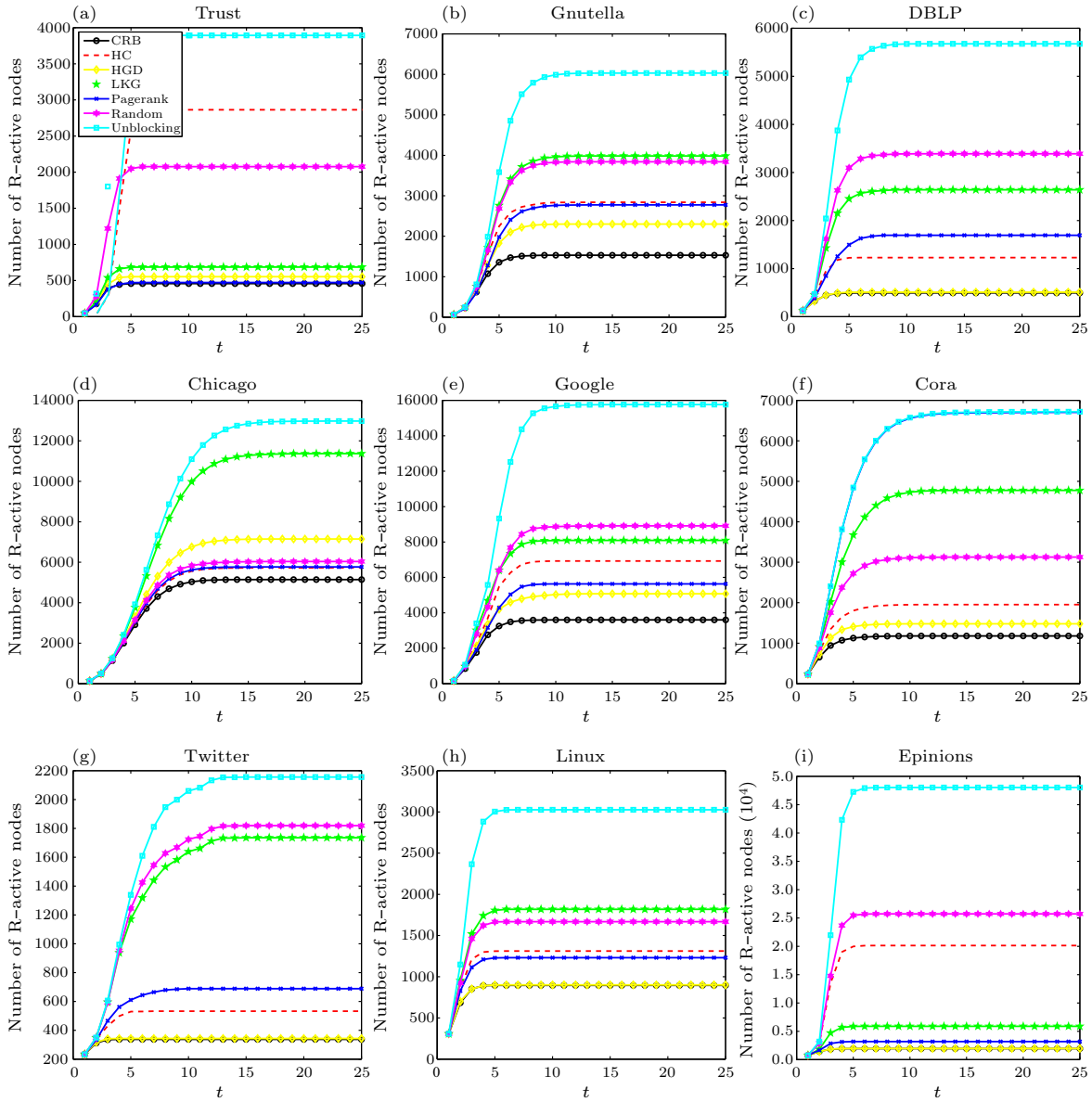


Fig. 4. The spreading influence against time t on nine datasets obtained by CRB and six comparision algorithms, where the spreading influence represents the number of activated nodes at time t in the range between 0 and 25. The ratio of rumor and anti-rumor seeds is fixed as one percent of the size of datasets. The $F(t)$ values are obtained under the CLTST model by averaging 200 independent runs.

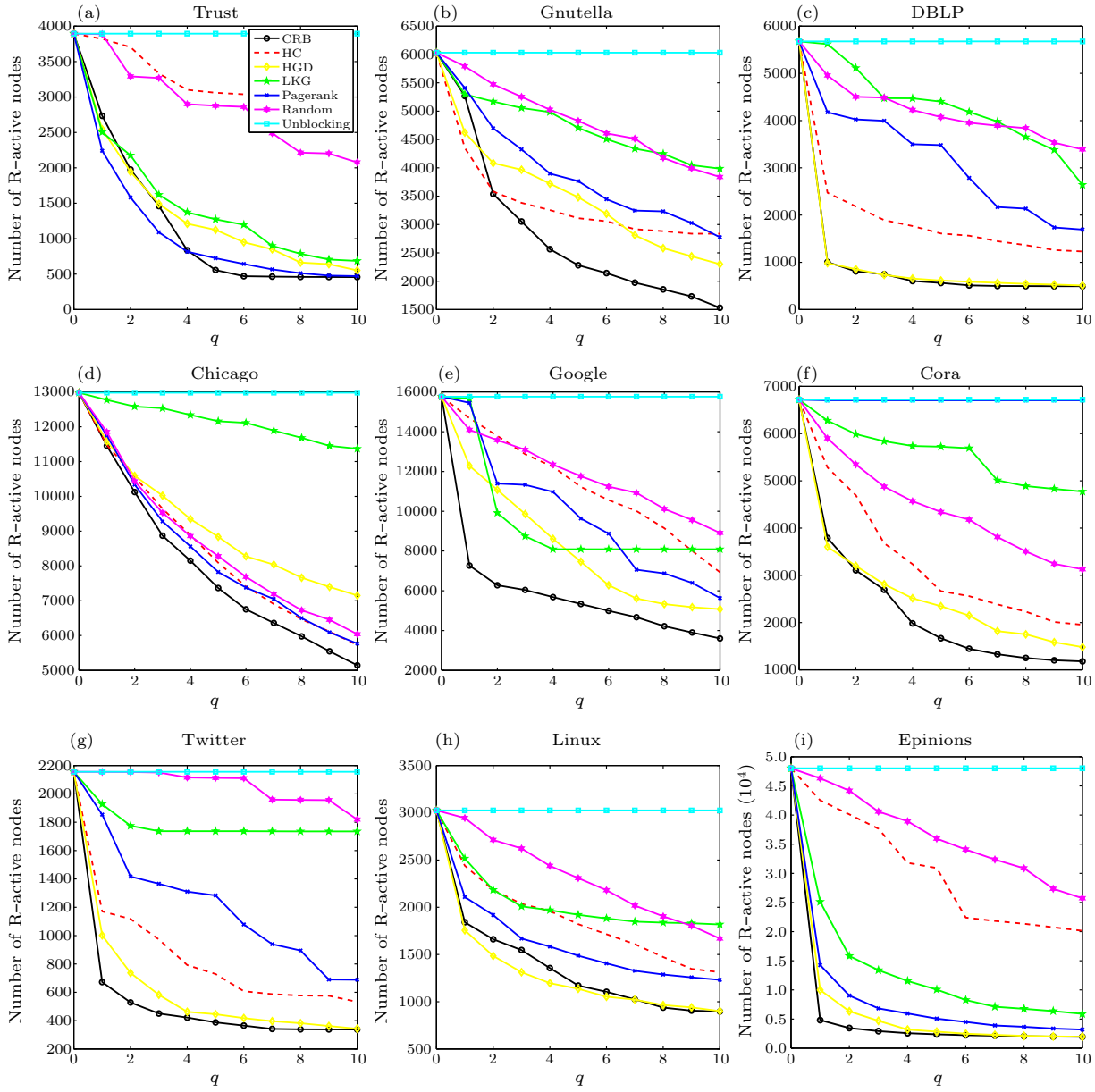


Fig. 5. The stable propagation influence of seven algorithms on nine datasets. The ratio of rumor is fixed as one percent of the size of datasets, and the ratio of anti-rumor seeds varies in a certain interval ($|S_A| \in [0.001, 0.01] \times N$) to better depict the performance variation. The stable propagation influence values are obtained under the CLTST model by averaging 200 independent runs.

Moreover, we test the performance of algorithms in rumor influence blocking under different anti-rumor seeds proportions. In this experiment, the number of anti-rumor seeds ($|S_A|$) varies in a certain interval to better depict the performance variation, where $|S_A| = \lceil q \cdot N \rceil$ ($q \in [0, 10]$), and the number of rumor seeds is fixed as $|S_R| = \lceil 0.01 \cdot N \rceil$. Figure 6 shows the rumor blocking performance of the algorithms mentioned above under different anti-rumor seeds proportions. Obviously, the number of R-state nodes of these seven algorithms is decreasing continuously as the proportion of anti-rumor seeds increases. As displayed by Fig. 6, the rumor blocking performance of the proposed CRB algorithm always ranks first under different anti-rumor seeds proportions. In four datasets including Chicago, Google, Cora and Twitter, the number of R-state nodes obtained by the proposed CRB is much lower than those of six compared algorithms. In Trust dataset, CRB performs worse than PageRank when $q \leq 4$, and the proposed algorithm has the lowest number of R-state nodes under a larger anti-rumor seeds proportion. In Gnutella dataset, CRB is inferior to other six compared algorithms when the proportion of anti-rumor seeds is low, but it outperforms all the compared algorithms when $q \geq 3$. In DBLP, Linux and Epinions datasets, the performance of CRB is similar to HGD when $q \geq 6$, and these two algorithms outperform other five algorithms under different proportions. In summary, the experimental results show that the performance of CRB is not sensitive to the rumor and anti-rumor seeds proportions.

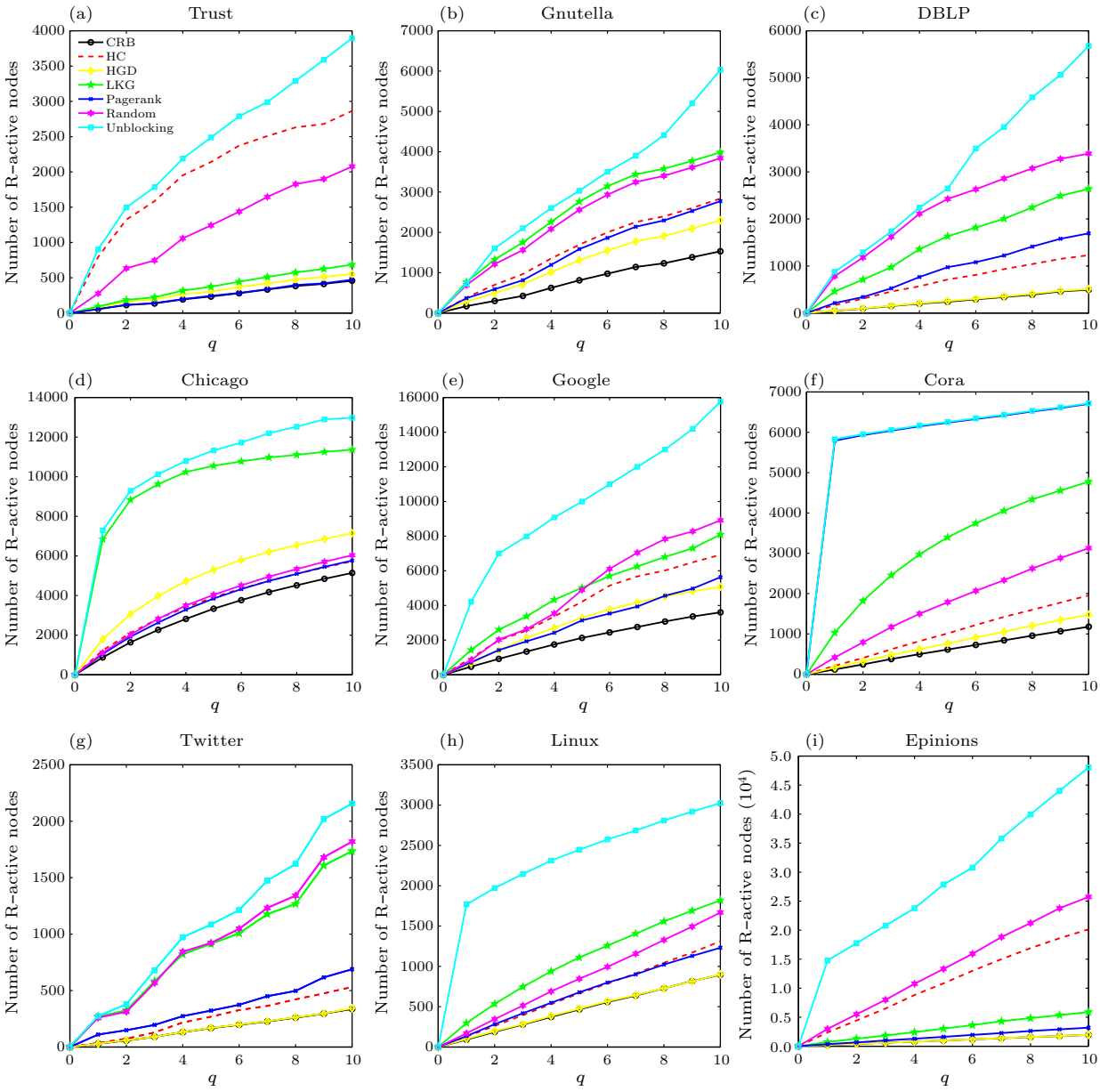


Fig. 6. The stable propagation influence of seven algorithms on nine datasets. The ratio of rumor seeds varies in a certain interval ($|S_R| \in [0.001, 0.01] \times N$) to better depict the performance variation, and the ratio of rumor seeds is fixed as one percent of the size of datasets. The stable propagation influence values are obtained under the CLTST model by averaging 200 independent runs.

6.3. Anti-rumor seeds distribution

To maximize the influence of anti-rumor, the selected seed spreaders should be dispersed separately in the network, and the distances between anti-rumor spreaders are as far as possible.[70] In the third experiment, we pay attention to the average distances between anti-rumor seeds (d_{mean})[71,72] detected by seven different algorithms. The average distance d_{mean} between the anti-rumor seeds is defined as

$$d_{\text{mean}} = \frac{2}{k(k-1)} \sum_{i \neq j \in S_A} d_{ij}, \tag{15}$$

where $k = |S_A|$ is the number of anti-rumor seeds, and d_{ij} is the distance between node i and node j.

As an important evaluation method of the RBM problem, the value of d_{mean} can measure the dispersion degree among seed nodes in the network. Figure 7 shows the d_{mean} of anti-rumor seeds obtained by seven algorithms on nine datasets. As observed in Fig. 7, the performance of the proposed CRB algorithm is better than six compared algorithms on these datasets. In other words, the anti-rumor seeds selected by the proposed CRB algorithm are decentralized separately, which can efficiently minimize the phenomenon of "rich-club". Specifically, in Trust, Chicago, Twitter and Epinions datasets, CRB outperforms other six compared algorithms under different seed ratios. In Gnutella, DBLP and Google datasets, the d_{mean} values of HGD algorithm are approximated to those of the proposed CRB algorithm, and these two algorithms perform better than other five algorithms. In Cora and Linux datasets, the d_{mean} values of CRB, LKG and PageRank algorithms are signifi-

cantly higher than other four algorithms. In summary, the performance of CRB algorithm is superior to other six compared algorithms in terms of anti-rumor seeds distribution, which indicates that the seed spreaders identified by CRB algorithm can spread truth information to majority of nodes, so as to effectively suppress the propagation of rumor.

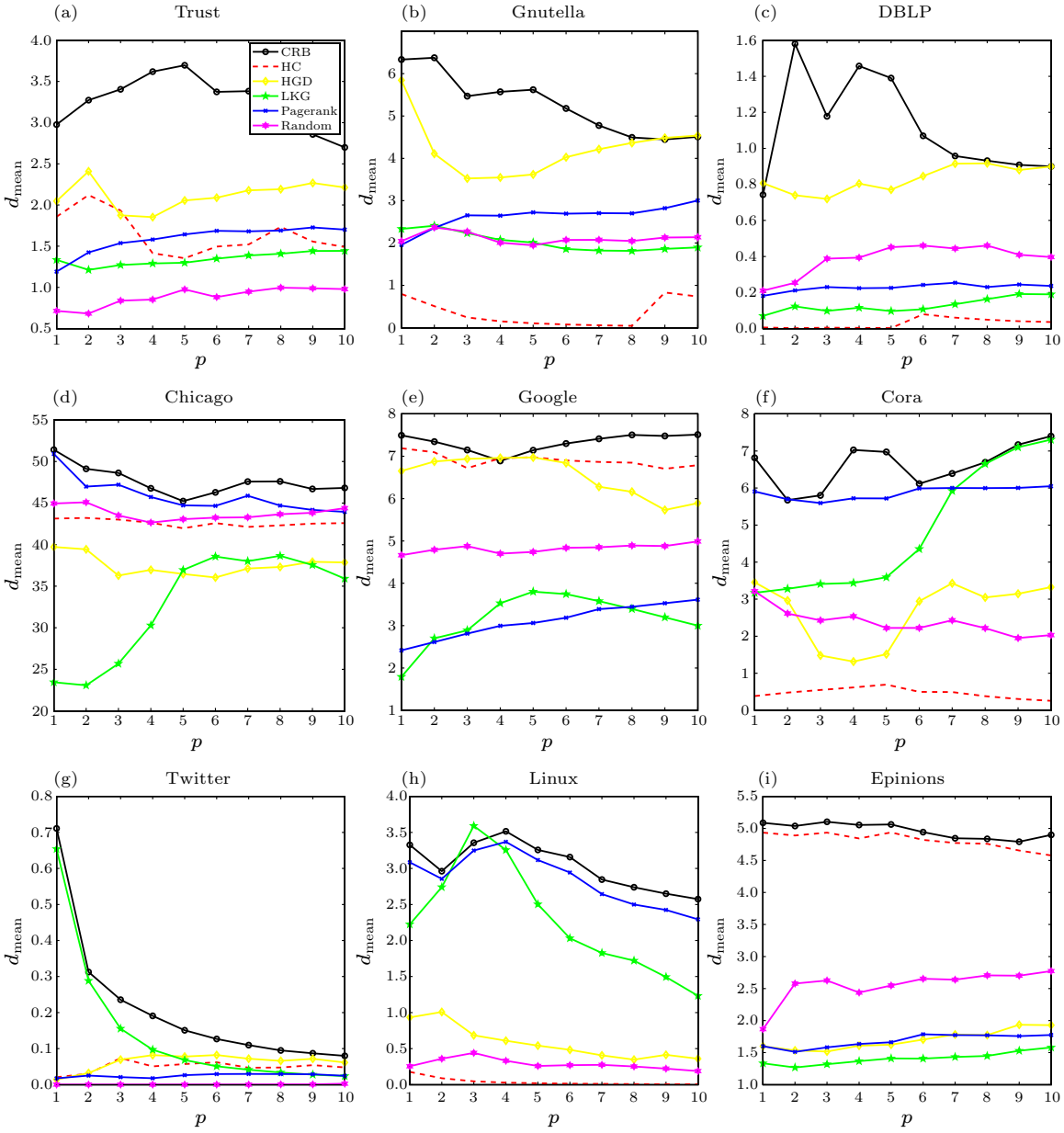


Fig. 7. The average distance d_{mean} (under ten different anti-rumor seed ratios $p \in [0.001, 0.01]$) between each pair of anti-rumor seeds obtained by different algorithms on nine real-world datasets.

Table 3. Average running time (seconds) over 200 runs with fixed rumor seeds and anti-rumor seeds.

Dataset	CRB (proposed)	HC	HGD	LKG	Pagerank	Random
Trust	3454.5647	3678.4648	3621.3135	4055.1564	3455.5465	1556.5649
Gnutella	4124.5456	4455.2678	4373.4621	4875.5156	4092.1215	2090.0856
DBLP	6532.1245	7054.2455	6921.2137	7687.5516	6412.7981	3389.7909
Chicago	6601.2313	7183.7618	6992.1722	7743.1222	6463.1719	3435.4653
Google	7012.1260	7700.2179	7302.1880	8021.7867	6941.2987	3780.9821
Cora	8721.0121	9596.7218	9072.1987	10082.1219	8582.1787	4662.1978
Twitter	8778.7611	9637.4616	9124.9847	10423.1355	8634.9847	4694.5613
Linux	9821.8978	10528.2792	10329.8799	11887.6312	9584.7656	5787.6547
Epinions	19941.5485	21798.4511	20946.5442	25787.1212	19249.8513	11287.5614

6.4. Running time

Computational complexity is one of the most important indicators for evaluating performance of algorithms. In the third group of experiments, we compare the running time of the proposed CRB algorithm and six contrast algorithms. Specifically, the running time of these seven algorithms on the nine OSNs is shown in Table 3. Here, the ratio of rumor seeds and anti-rumor seeds is fixed as $\lceil 0.01 \cdot N \rceil$. Since unblocking strategy requires no measures in rumor propagation, the time cost of this algorithm is negligible. Among the remained six algorithms, the random algorithm takes the shortest running time, and the running time of the proposed CRB algorithm is about twice that of the random algorithm. In addition, the running time of the PageRank algorithm is close to the proposed algorithm. It is obvious that CRB and PageRank outperform the HC, HGD and LKG algorithms. Overall, the proposed algorithm achieves the better performance with a low computational cost.

7. Conclusions

In this study, we propose a new rumor blocking algorithm in online social networks based on a competitive spreading model and influence maximization. First, based on the classic LT model, we propose a new model called CLTST to simulate the situations where rumor and anti-rumor information are competitive spreading in the same network. Most of the existing diffusion models neglect that users' attitudes toward an event change as new information is gathered, the proposed model introduces two thresholds to take effect in infecting inactive nodes or transforming the state of active nodes, and each activated node will transform its state when total influence from the active in-neighbors is changed. Second, we put forward the CRB algorithm to select a set of seeds for spreading anti-rumor information to other nodes, which consists of three phases: (1) community detection, (2) selection of anti-rumor candidate seeds, (3) generation of anti-rumor seed set. The proposed CRB algorithm evaluates the influence of nodes by comprehensively considering the information of community structure, the information about node itself and the mutual interactions with its neighbors. In addition, to avoid premature convergence and yield a satisfactory solution within a reasonable time, an adaptive search strategy is designed to obtain the optimal seed set from the candidates.

Experimental results show that CRB has better performance in weakening the rumor dissemination ability. More specifically, in terms of rumor spread, the proposed CRB algorithm performs better than other comparison algorithms with a margin as high as 0.86% in each dataset. In addition, in terms of the percentage of rumor and anti-rumor seeds, the R-state values of CRB are much lower than those of comparison algorithms with a margin as high as 2.71%. Furthermore, in terms of the anti-rumor seeds dispersion, the proposed CRB algorithm achieves the largest d_{mean} value, and the margin between CRB and six comparison algorithms with a margin as high as 4.59% in each dataset. As the research continues, we will continue to explore in this direction and strive to investigate the performance of algorithm on other spreading dynamics models, or various types of networks such as temporal networks or heterogenous networks.

Acknowledgement

This work was supported by the National Social Science Fund of China (Grant No. 23BGL270).

References

[1] Vosoughi S, Roy D and Aral S 2018 *Science* **359** 1146
[2] Bovet A and Makse H A 2019 *Nat. Commun.* **10** 7
[3] Dong Y F, Huo L A, Xie X X and Li M 2023 *Chin. Phys. B* **32** 070205
[4] Zhan X X, Zhang K Y, Ge L, Huang J M, Zhang Z N, Wei L, Sun G Q, Liu C and Zhang Z K 2023 *IEEE Trans. Netw. Sci. Eng.* **10** 553
[5] Vicario M D, Bessi A, Zollo F, Petroni F, Scala A, Caldarelli G, Stanley, H E and Quattrociocchi W 2016 *Proc. Natl. Acad. Sci. USA* **113** 554
[6] Alkhalifa R, Kochkina E and Zubiaga A 2023 *Inf. Process. Manag.* **60** 103200
[7] Xu G Q and Dong C 2024 *Expert Syst. Appl.* **235** 121154
[8] Luvembe A M, Li W M, Li S H, Liu F F and Wu X 2024 *Inf. Process. Manag.* **61** 103653
[9] Guo J X, Chen T T and Wu W L 2021 *IEEE-ACM Trans. Netw.* **29** 386
[10] Zhang Y H and Zhu J J 2022 *Chin. Phys. B* **31** 060202
[11] Meng F Y, Medo M and Buechel B 2022 *Inf. Sci.* **606** 742
[12] Xu G Q and Meng L 2023 *Chaos Solitons Fractals* **168** 113155
[13] Tong G M, Wu W L, Guo L, Li D Y, Liu C, Liu B and Du D Z 2020 *IEEE Trans. Netw. Sci. Eng.* **7** 845
[14] He Q, Lü Y J, Wang X W, Huang M and Cai Y L 2022 *IEEE Syst. J.* **16** 6457
[15] Meng L, Xu G Q, Yang P L and Tu D Q 2022 *J. Comput. Sci.* **60** 101591
[16] Zhong X J, Yang Y K, Miao Y Q, Peng Y Q and Liu G Y 2022 *Chin. Phys. B* **31** 040205
[17] He X R, Song G J, Chen W and Jiang Q Y 2012 *Proceedings of the 2012 SIAM International Conference on Data Mining* (Philadelphia: SIAM) p. 463
[18] Tan Z H, Wu D K, Gao T H, You I and Sharma V 2019 *Future Generation Comput. Syst.* **94** 293
[19] Sun X L, Wang Y G and Cang L Q 2022 *Chin. Phys. B* **31** 050202
[20] Manouchehri M A, Helfroush M S and Danyali H 2021 *IEEE Trans. Syst. Man. Cybern: Syst.* **52** 4578
[21] Luo X P, Jiang H J, Cheng S S and Li J R 2023 *Chin. Phys. B* **32** 058702
[22] Nguyen N P, Yan G H, Thai M T and Eidenbenz S 2012 *Proceedings of the 4th Annual ACM Web Science Conference* (New York: Association for Computing Machinery) p. 213
[23] Wang S Z, Zhao X J, Chen Y, Li Z J, Zhang K and Xia J L 2013 *Proceedings of the 27th AAAI Conference on Late-Breaking Developments in the Field of Artificial Intelligence* (Singerpore: Association for the Advancement of Artificial Intelligence) p. 134
[24] Kimura M, Saito K and Motoda H 2008 *Proceedings of the 23th AAAI Conference on Artificial Intelligence* (Singerpore: Association for the Advancement of Artificial Intelligence) p. 1175
[25] Nepusz T and Vicsek T 2012 *Nat. Phys.* **8** 568
[26] Hu X, Xiong X, Wu Y, Shi M J, Wei P and Ma C M 2023 *Expert Syst. Appl.* **212** 118638
[27] Zhang Y H and Zhu J J 2022 *Chin. Phys. B* **31** 060202

[28] Wu P and Pan L 2017 *Comput. Netw.* **123** 38
[29] Pham C V, Dinh H M, Nguyen H D, Dang H T and Hoang H X 2017 *Proceedings of the 8th International Symposium on Information and Communication Technology* (New York: Association for Computing Machinery) p. 262
[30] Zheng J G and Pan L 2018 *3rd International Conference on Security of Smart Cities, Industrial Control System and Communications*, October 18–19, 2018 Shanghai, China, p. 1
[31] Yan R D, Li D Y, Wu W L, Du D Z and Wang Y C 2020 *IEEE Trans. Netw. Sci. Eng.* **7** 1067
[32] Ding X J, Li M Y, Tian Y and Jiang M 2021 *IEEE Trans. Eng. Manag.* **11** 1
[33] Yao X P, Gu Y, Gu C L and Huang H J 2022 *Comput. Commun.* **182** 41
[34] Yang L, Ma Z Y, Li Z W and Giua A 2023 *IEEE Trans. Syst. Man. Cybern: Syst.* **53** 3990
[35] Schneider C M, Mihaljev T, Havlin S and Herrmann H J 2011 *Phys. Rev. E* **84** 061911
[36] Tong H H, Prakash B A, Eliassi-Rad T, Faloutsos M and Faloutsos C 2012 *Proceedings of the 21st International Conference on Information and Knowledge Management* (New York: Association for Computing Machinery) p. 245
[37] Yao Q P, Zhou C, Xiang L B, Cao Y N and Li G 2015 *International Conference on Trustworthy Computing and Services* (Berlin: Springer) p. 65
[38] Dey P and Roy S 2017 *International Conference on Advanced Networks and Telecommunications Systems*, December 17–20, 2017, Bhubaneswar, India. p. 1
[39] Yan R D, Li Y, Wu W L, Li D Y and Wang Y C 2019 *ACM Trans. Knowl. Discov. Data* **13** 1
[40] Xiang F S, Wang J H, Wu Y P, Wang X Y, Chen C and Zhang Y 2024 *World Wide Web* **27** 6
[41] Budak C, Agrawal D and Abbadi A 2011 *WWW '11: Proceedings of the 20th international conference on World wide web* (New York: Association for Computing Machinery) p. 665
[42] Tong G M, Wu W L and Du D Z 2018 *IEEE Trans. Comput. Soc. Syst.* **5** 468
[43] Yang L and Li Z W 2019 *Inf. Sci.* **506** 113
[44] Xiao Y P, Yang Q F, Sang C Y and Liu Y B 2020 *IEEE Trans. Netw. Serv. Manag.* **17** 1910
[45] Li Q, Zeng C, Xu W and Xiao Y P 2022 *J. Netw. Comput. Appl.* **201** 103343
[46] Jiang Z Y, Chen X Y, Ma J F and Yu P S 2022 *IEEE Trans. Syst. Man. Cybern: Syst.* **52** 6383
[47] He Q, Du H W and Liang Z W 2023 *IEEE Trans. Comput. Soc. Syst.* **10** 2624
[48] Xie M, Zhan X X, Liu C and Zhang Z K 2023 *Inf. Process. Manag.* **60** 103161
[49] Yang P L, Zhao L J, Lu Z, Zhou L X, Meng F Y and Qian Y 2023 *Chaos Solitons Fractals* **173** 113720
[50] Beni H A, Bouyer A, Azimi S, Rouhi A and Arasteh B 2023 *Inf. Sci.* **640** 119105
[51] Xie X W, Zhan X X, Zhang Z K and Liu C 2023 *Chaos* **33** 013104
[52] Umrawal A K, Quinn C J and Aggarwal V 2023 *IEEE Trans. Emerg. Top. Comput. Intell.* **7** 1253
[53] Guo C, Li W M, Liu F F, Zhong K X, Wu X, Zhang Y G and Jin Q 2024 *Neurocomputing* **564** 126936
[54] Bozorgi A, Samet S, Kwisthout J and Wareham T 2017 *Knowledge-Based Syst.* **134** 149
[55] Wang J and Li K 2021 *Chin. Phys. B* **30** 120518
[56] Li W M, Zhou X K, Yang C, Fan Y T, Wang Z and Liu Y X 2021 *Inf. Fusion* **79** 110
[57] Banerjee S, Jenamani M and Pratihar D K 2023 *Expert Syst. Appl.* **125** 1
[58] Blondel V D, Guillaume J L, Lambiotte R and Lefebvre E 2008 *J. Stat. Mech-Theory Exp.* P10008
[59] Bouyer A, Beni H A, Arasteh B, Aghaee Z and Ghanbarzadeh R 2022 *Expert Syst. Appl.* **213** 118869
[60] Dong C, Xu G Q, Meng L and Yang P L 2022 *Physica A* **603** 127797
[61] Yan Z J, Xia Y X, Guo L J, Zhu L Z, Liang Y Y and Tu H C 2023 *Chin. Phys. B* **32** 068901
[62] Li W M, Guo C, Deng Z B, Liu F F, Wang J J, Guo R Q, Wang C and Jin Q 2023 *Expert Syst. Appl.* **270** 110547
[63] Christakis N A and Fowler J H 2013 *Stat. Med.* **32** 556
[64] Dong C, Xu G Q, Yang P L and Meng L 2023 *Expert Syst. Appl.* **212** 118702
[65] Jiang Q Y, Song G J, Gao C, Wang Y, Si W J and Xie K Q 2011 *Proceedings of the AAAI Conference on Artificial Intelligence* **25** 127
[66] Bao Z K, Liu J G and Zhang H F 2017 *Phys. Lett. A* **381** 976
[67] Jiang L C, Zhao X, Ge B, Xiao W D and Ruan Y R 2019 *Physica A* **381** 58
[68] Samir A M, Rady S and Gharib T F 2021 *Physica A* **582** 126258
[69] Brin S and Page L 1998 *Comput. Netw. ISDN Syst.* **30** 107
[70] Zhang J X, Chen D B, Dong Q and Zhao Z D 2016 *Sci. Rep.* **6** 27823
[71] Liu P F, Li L J, Fang S Y and Yao Y K 2021 *Chaos Solitons Fractals* **152** 111309
[72] Guo C, Li W M, Wang J C, Yu X, Liu X, Luvembe A M, Wang C and Jin Q 2024 *Knowledge-Based Syst.* **291** 111580