International Journal of Automation and Computing

15(4), August 2018, 377-401
DOI: 10.1007/s11633-018-1124-0

Study on Information Diffusion Analysis in Social Networks and Its Applications

Biao Chang Tong Xu Qi Liu En-Hong Chen

Anhui Province Key Laboratory of Big Data Analysis and Application, School of Computer Science,
University of Science and Technology of China, Hefei 230026, China

Abstract: Due to the prevalence of social network services, more and more attentions are paid to explore how information diffuses and users affect each other in these networks, which has a wide range of applications, such as viral marketing, reposting prediction and social recommendation. Therefore, in this paper, we review the recent advances on information diffusion analysis in social networks and its applications. Specifically, we first shed light on several popular models to describe the information diffusion process in social networks, which enables three practical applications, i.e., influence evaluation, influence maximization and information source detection. Then, we discuss how to evaluate the authority and influence based on network structures. After that, current solutions to influence maximization and information source detection are discussed in detail, respectively. Finally, some possible research directions of information diffusion analysis are listed for further study.

Keywords: Information diffusion, influence evaluation, influence maximization, information source detection, social network.

1 Introduction

Recent years have witnessed a rapid development of social network services (SNS), such as Twitter[1], Facebook[2], and Sina Weibo[3]. More and more users are taking them to share information with friends. For example, in Facebook, there are over 2.01 billion monthly active users all over the world during June 2017[1]. These social networks have the characteristics of openness (i.e., every one can join and keep in touch with the outside world), interaction (i.e., users can interact with friends about a movie or an accident by replying or reposting) and timeliness (i.e., a user can update status messages at any time)[2, 3].

Users′ participations generate tremendous data in social networks. In Twitter, on average, 500 million tweets are posted per day[4]. This data contains various informations. For example, people may tweet their opinions on breaking news; or may just update messages to tell friends what have happened in their daily life. Companies may hire influential users to promote new products such as movies and electronic goods. Besides, those information are flowing and can diffuse among users. Once users see something interesting, they can repost or forward these contents to their friends. If their friends also like the contents, they can further share them with their own friends, which thus causes information diffusion in the network, i.e., the so-called effect of word-of-mouth. Those users who adopt the information are called influenced or active.

However, how the information diffuses through networks is usually unknown. Understanding the diffusion mechanism behind massive information is important for a wide range of applications, such as viral marketing[4–6], social behavior prediction[7–9], social recommendation[10–12], and community detection[13–15]. This issue has attracted researchers from various fields including epidemiology, computer science, and sociology. They proposed different kinds of information diffusion models to describe and simulate this process, such as the independent cascade (IC) model[16], linear threshold (LT) model[17, 4] and epidemic models[18]. Most models are contagious and assume that the information starts to diffuse from a source (or seed) node set, and other nodes can access the information only from their neighbors.

The discovered diffusion models have been applied to many practical applications. For example, first, by evaluating users′ influence, we can identify influential spreaders[19, 20] and find experts[21–23]. Second, by choosing seed users and solving the so-called influence maximization problem[4], we can maximize the number of influenced users. This is significant to promote new products through the word-of-mouth effect[4] or place sensors to quickly detect contaminants in the water distribution network in a city[24, 25]. Third, after the information diffuses

Review
Manuscript received November 28, 2017; accepted March 8, 2018; published online June 16, 2018
Recommended by Associate Editor Qing-Long Han

[1] https://www.twitter.com/

[2] https://www.facebook.com/

[3] https://www.weibo.com/

[4] http://www.internetlivestats.com/twitter-statistics/

from a set of source nodes for a period of time, it will influence more nodes. We can infer the source nodes according to these observed influenced nodes, which is called information source detection. It can help to prevent the outbreak of an epidemic[26–28] and trace the rumor source in social networks[29, 30].

Therefore, we will review the recent development of information diffusion analysis in social networks and its applications. Fig. 1 gives an overview of this paper. The rest parts are organized as follows. We start with some preliminaries of social networks in Section 2. Section 3 introduces three basic kinds of information diffusion models. Then we list methods which are used to evaluate the authority and influence in Section 4. Sections 5 and 6 show the solutions to influence maximization and information source detection, respectively. Finally, we conclude some possible directions for further study in Section 7.

2 Preliminaries

A social network can be denoted as $G(V, E, \boldsymbol{W})$, where V is the node set of size n, E is the edge set of size m, and $\boldsymbol{W} = [w_{ij}]$. Edge e_{ij} indicates the direction of information flow from node i to j with a propagation probability $w_{ij} \in [0, 1]$. Undirected networks can be converted into directed ones by $w_{ij} = w_{ji}$. Fig. 2 shows a toy social network with 10 nodes, and the edges indicate possible directions of information flows. Lots of real-world networks can be viewed as instances of social networks, such as

1) Microblogging networks. Nodes represent users or organizations. For example, in Twitter, if user v is a follower of user u, there will be an edge from u to v, and w_{uv} is the probability u affects v and can be learned from historical actions[31].

2) Citation networks. Nodes represent papers and edge e_{uv} indicates paper v has cited u. A simple approach to determine the propagation probability e_{uv} from u to v is sharing u's influence among its neighbors, i.e., $e_{uv} = \dfrac{1}{d_{o(u)}}$, where $d_o(u)$ is the out-degree of u.

3) Collaboration networks. Nodes represent authors and e_{uv} indicates author u and v have collaborated on at least one paper. The propagation probability is proportional to the number of papers that two authors has collaborated on.

4) Email networks. Edge e_{uv} indicates user u has sent at least a email to v. The propagation probability is proportional to the number of emails between two users.

Different kinds of information can spread in a social network, such as innovations, contagion, opinions about specific events. Note that a node is influenced if it adopts the information. An influenced node will further propagate the information to its neighbors, i.e., word-of-mouth, which causes information diffusion in the network. Thus, except for specific explanation, every node has two states: active (i.e., infected/influenced) and inactive. For example, in Twitter, users reposting a funny tweet are active, while others are inactive.

Datasets. There are many websites providing open

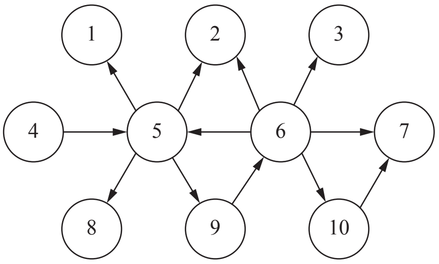


Fig. 2 A toy social network where edges indicate the directions of information flows

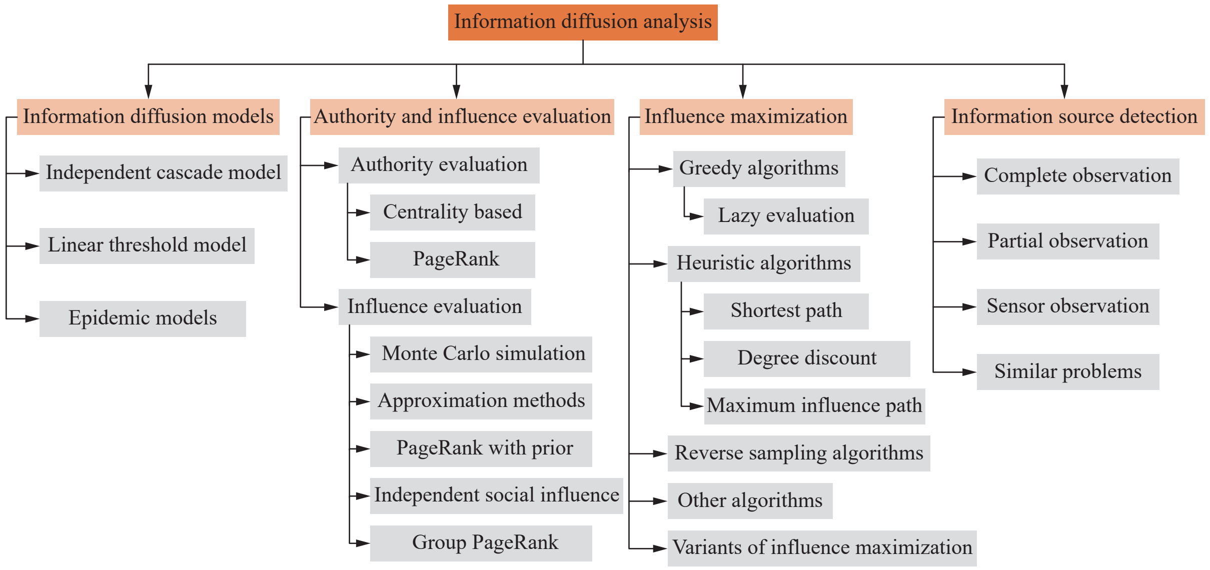


Fig. 1 Overview of recent advances on information diffusion analysis included in this paper

datasets of social networks for research. Here we list some of them for easy reference.

1) Stanford large network dataset collection[5]. It is a collection of more than 50 large network datasets from tens of thousands of nodes and edges to tens of millions of nodes and edges, including social networks, web graphs, road networks, Internet networks, citation networks, collaboration networks, and communication networks.

2) Aminer[6]. It provides a repository of external datasets for social network analysis, including microblogging networks, patent dataset from Patentminer.org, knowledge linking dataset, mobile dataset and other online social networks.

3) Social computing data repository[7]. It hosts datasets from many different social media sites, most of which have blogging capacity, such as BlogCatalog, Twitter, MyBlogLog, Digg, StumbleUpon, del.icio.us, MySpace, LiveJournal, The Unofficial Apple Weblog (TUAW), Reddit, etc.

4) KONECT[8]. Koblenz Network Collection (KONECT) is a project to collect large network datasets for researching in network science and related fields. It includes several hundred network datasets of various types, including directed, undirected, bipartite, weighted, unweighted, signed and rating networks.

3 Information diffusion models

How the information diffuses through networks is unknown and has been studied by researchers from various fields including epidemiology[18], computer science[16], and sociology[32]. They proposed different kinds of information diffusion models to describe and simulate this process. Most of them are contagious and follow two rules below:

Rule 1. Every piece of information diffusion starts from several source nodes.

For example, John is a movie star and posts a tweet on Twitter to promote his new movie. This may cause a hot discussion among his fans, and thus he is the source node initializing this diffusion.

Rule 2. Every disseminator can access the information only from its neighbors.

In the above example, Alice is a fan of John, and she can only read the tweet from John or other fans' retweeting.

All information diffusion models are consistent with Rule 2, but achieve Rule 2 in different ways. They can be divided into two categories: 1) progressive models where nodes can switch from being inactive to being active, but do not switch in the other direction and 2) non progressive models where nodes can switch in both directions and allow to be activated for many times. In the next part, we will introduce three basic information diffusion models, namely independent cascade (IC) model, linear threshold (LT) model and Epidemic models, which are widely used and are fundamental for personal influence evaluation, influence maximization, etc. More diffusion models can be found here[33].

3.1 Independent cascade model

Independent cascade (IC) model was proposed by Goldenberg et al.[16] in 2001. It describes a diffusion like Domino and assumes the information starts from a set of active seed nodes A_0, which follows Rule 1. For viral marketing, A_0 are the users who have discounts and would like to promote the products among their friends. Every active node cannot switch back to being inactive.

As time goes by, inactive nodes can receive information from active ones. Specifically, at time t, A_t are the set of current active nodes. For node $u \in A_t$, it have only one chance to affect its inactive adjacent node v with the probability w_{uv}. If successful, v becomes active and will try to affect its own neighbors in the next time-stamp $t+1$, otherwise v keeps inactive and u has no chance to affect v any more. If node v has more than two active in-neighbors, they will affect v independently. The above explains how the IC model interprets Rule 2. This process continues to unfold until no more nodes become active. Note that the independent cascade model is progressive and stochastic, thus the final active nodes A_∞ may change when the information starts from different seed nodes A_0.

3.2 Linear threshold model

Linear threshold (LT) model was proposed by Granovetter[17] in 1978. It assumes that each node v has a specific threshold θ_v uniformly sampled from the interval [0, 1], and $\sum_{u \in V} w_{uv} \leq 1$. Given the initial seed nodes A_0, the diffusion process will unfold deterministically in discrete steps. Specifically, in step t, nodes which were active in previous step will remain active, and an inactive node v becomes active if

$$\sum_{u \in N_{in}(v)} w_{uv} \geq \theta_v \qquad (1)$$

where $N_{in}(v)$ is a set of v's active in-neighbors. This process continues to unfold until no more nodes become active. We can see the probability that an inactive node becomes active, increases monotonically as more of its neighbors become active. What's more, v's threshold can be considered as a weighted fraction of v's neighbors that must become active in order to successfully affect v.

In summary, there are two main differences between

[5] https://snap.stanford.edu/data/

[6] https://cn.aminer.org/data-sna

[7] http://socialcomputing.asu.edu/pages/home

[8] http://konect.uni-koblenz.de/

LT and IC models. First, active nodes in the LT model have more than one chance to affect their inactive neighbors. Second, node v's active neighbors will affect v together in the LT model, while v's active neighbors only have one chance to independently affect v in the IC model. However, Kempe and McKendrick[18] proposed a general threshold model and a general cascade model, and have proven their equivalence. Besides, both LT and IC models are special cases of the triggering model[18], where each node v independently chooses a random triggering set T_v according to some distribution over subsets of its neighbors, and v is active if it has a neighbor in its chosen T_v.

3.3 Epidemic models

Some researchers adopt the epidemic models to simulate the infection and recovery processes of nodes in networks[34, 35], which are originally describing how a disease spreads within a population in epidemiology[18]. The information or disease also starts from a set of infected seed nodes A_0. The simplest is the susceptible-infected (SI) model[35, 36], which assumes each node has two possible states: susceptible and infected. When a node is in the susceptible state, it can potentially get infected by the information. Once a node u is infected, it will remain infected forever and spreads the information to its susceptible adjacent node v with a probability of w_{uv}. Note that diffusions along edges are supposed to be independent. The susceptible-infected-susceptible (SIS) model[36] is similar to the SI model, except that an infected node u can become susceptible again with a probability of γ_u.

Susceptible-infected-recovered (SIR) model[18] generalizes the SI model, and assumes a node has three states: susceptible, infected and recovered. When a node u is infected, it has a probability of γ_u to recover and becomes immune to the disease, which means u will not get infected any more. Its other settings are similar to the SI model. Another epidemic model, i.e., susceptible-infected-recovered-susceptible (SIRS)[36] extends the SIR model and also assumes a node has the above three states. But after node u recovers from being infected, it can become susceptible again with a probability of λ_u, Fig. 3 shows the possible state changes of a node in the above four epidemic models.

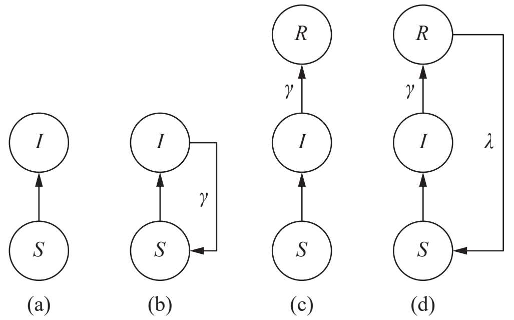


Fig. 3 Node state transition diagrams for four epidemic models: (a) SI, (b) SIS, (c) SIR, (d) SIRS, where γ and λ is the transition probability

Indeed, there are other epidemic models, such as susceptible exposed infected recovered (SEIR)[37], maternal susceptible infected recovered (MSIR)[38], susceptible exposed infected recovered susceptible (SEIRS)[39]. Readers can refer to the work[40] for more details. How to exploit these models for information diffusion analysis is underexplored.

4 Authority and influence evaluation

Based on the above diffusion models, we can evaluate the influence or authority of an individual in social networks, which is important for influential spreader identification[19, 20, 41] and expert finding[42, 21, 22]. A user's influence and authority seem to be different at a first glance, because "influence" measures the impact that it has on others through out-links (e.g., persuading them to buy a product) while "authority" is the endorsement received from its followers through in-links. However, some works[43, 44] have realized that they have a close relationship because an individual earns its authority by influencing others. In the next, we will show how to evaluate the influence and authority.

4.1 Authority evaluation

In this subsection, we focus on the solutions to authority evaluation which only exploit the network structure, such as centrality based and PageRank. Readers can find other methods by referring to the work[45].

4.1.1 Centrality based

There are many ways to compute the centrality of a node and its larger value means more influential. The first and simplest way is degree centrality, which equals to the number of links upon a node. In a directed network, we can use outdegree and indegree to measure the centrality, respectively. For a node u, outdegree can evaluate its importance as information senders, while indegree measures its gregariousness. That's to say, the larger u's outdegree is, the more users u will affect. While the larger u's indegree is, the closer u is to others.

For degree centrality, it considers nodes with more connections to be more influential. In fact, the influence of a node should be determined by its neighbors. Eigenvector centrality provides another way to measure individual influence with this fact. Let $\boldsymbol{A}$ be the adjacency matrix, i.e., $a_{uv} = 1$ if node u is linked to node v, otherwise $a_{uv} = 0$. Formally, u's eigenvector centrality, $c_e(u)$, can be computed by

$$c_e(u) = \frac{1}{\lambda} \sum_{v \in V} a_{v,u} \times c_e(v) \tag{2}$$

where λ is a fixed constant. This equation can be rewritten in vector notation as

$$\lambda \boldsymbol{c}_e = \boldsymbol{A}^{\mathrm{T}} \mathbf{c}_e \tag{3}$$

where $\boldsymbol{c}_e = (c_e(v_1), c_e(v_2), \cdots, c_e(v_n))^{\mathrm{T}}$. Thus, we can see $\boldsymbol{c}_e$ is an eigenvector of $\boldsymbol{A}$, which corresponds to the largest eigenvalue according to Perron-Frobenius theorem.

The third way to compute centralities is based on the distance between nodes. 1) Node u's closeness centrality[46], $c_c(u)$, is defined as the reciprocal of the average shortest distance between u and others. Formally,

$$c_c(u) = \frac{1}{\sum\limits_{v \in V} d(u, v)} \tag{4}$$

where $d(u, v)$ is the shortest distance between node u and v, computed by the topological distance or weights along the path. 2) u's betweenness centrality[47], $c_b(u)$, counts the number of shortest paths among others which pass through u. Formally,

$$c_b(u) = \sum_{s \neq u \neq t \in V} \frac{\sigma_{st}(u)}{\sigma_{st}} \tag{5}$$

where σ_{st} is the number of shortest paths between s and t, and $\sigma_{st}(u)$ is the number of shortest paths between s and t passing through u. 3) u's Jordan centrality[48], $c_j(u)$, is defined as the reciprocal of the maximum distance between u and other nodes. Formally,

$$c_j(u) = \frac{1}{\max\{d(u, v) | v \in V\}}. \tag{6}$$

Note that the closeness and Jordan centralities assume authoritative nodes can send information to others as fast as possible, while betweenness centrality shows how important a node is in connecting others as a pivot.

When comparing nodes of graphs with different sizes, we can normalize the aforementioned centralities by things like the number of nodes. Readers can find more details in the work[36].

4.1.2 PageRank

PageRank[49] is originally used for evaluating authorities of Web pages and as the cornerstone of Google's search engine[9]. It is also an extension of the normal eigenvector centrality discussed above. The general PageRank values $\boldsymbol{x} = (x(v_1), x(v_2), \cdots, x(v_n))^{\mathrm{T}}$ of nodes in G can be defined as

$$\boldsymbol{x} = d\boldsymbol{W}\boldsymbol{x} + \frac{1-d}{n}\boldsymbol{e} \tag{7}$$

where $d \in [0, 1]$ is a decay factor, n is the number of nodes, and $\boldsymbol{e}$ is a vector fully filled with ones. This equation could be solved by the power iteration. Please refer to the work[50] for more details.

The random surfer model[49] can explain PageRank vividly. A user starts surfing on a web page and then clicks current links randomly. He will continue clicking until stopping at a desired page. PageRank assumes that the surfer is more likely to stop at important pages. When $d = 1$, $\boldsymbol{x} = \boldsymbol{W}\boldsymbol{x}$ shows that $\boldsymbol{x}$ is the stationary distribution of a random walk with $\boldsymbol{W}$ as the transition matrix. But in real scenarios, many pages have no outlink or are in a small loop of web pages, and thus the surfer will be stuck. To overcome this problem, the surfer can randomly open a new page and keeps surfing. The second term on the right side of (7) tells this strategy: If the surfer is stuck, he will click a page with a probability of $\frac{1}{n}$.

Haveliwala[51] considered more personalized knowledge and proposed a topic-sensitive PageRank. The uniform personalization vector $\boldsymbol{e}$ in (7) is replaced by a nonuniform $\boldsymbol{q}$ whose i-th element equals to 1 if it belongs to the target topic, otherwise it would be 0. Kleinberg[52] designed a similar algorithm, called HITS. It computes the authority weight and hub weight in a subgraph simultaneously. Besides, Weng et al.[53] proposed TwitterRank, an extension of PageRank, to measure the influence of users in Twitter. It takes both the topical similarity between users and the link structure into account to measure the influence.

Due to its simplicity and effectiveness, PageRank has been applied to complete many tasks, such as influential spreaders identification[19] and link prediction[11] in social networks, item recommendation[54] and expert finding[22].

4.2 Influence evaluation

Someone's influence can be considered as the ability to affect others. Kempe et al.[4] defined the influence of a node set A to be the expected number of active nodes at the end of the process, which is also named influence spread, given that A is the initial active set. Many methods have been designed to compute this value efficiently and effectively.

4.2.1 Monte Carlo simulation

Kempe et al.[4] proposed to run Monte Carlo (MC) simulations to estimate the influence spread under the IC model or LT model. The MC simulation is a process: under the IC or LT model, we diffuse a piece of information from a node set A in the network, and can get the number of active nodes at the end of this diffusion. Therefore, the influence spread of a node set A, denoted as $f(A)$ can be estimated by

$$f(A) = \frac{1}{R} \sum_{v \in V} \delta(v) \tag{8}$$

where R is the count of MC simulations, and $\delta(v)$ is an

[9] https://www.google.com/

indicator. $\delta(v)=1$ if node v is active at the end, otherwise it would be 0. Each MC simulation is independent, and thus the law of large numbers ensures that (8) converges to the real value as long as R is large enough. However, this method is time-consuming, especially for large-scale networks. The authors left it as an open question to compute the influence spread.

4.2.2 Approximation methods

Chen et al.[55] further studied this problem, and got the following depressing result. Given a node set A, computing its influence spread $f(A)$ is #P-hard under the IC or LT model.[55, 56] Thus some researchers try to design approximation methods to estimate the influence spread. Aggarwal et al.[57] proposed a method, SteadyState-Spread, to determine the expected information spread of a given starting set of nodes A. They first computed the steady-state probability $\pi(i)$ that node i assimilates the information by solving the following system of nonlinear equations.

$$\pi(i)=\begin{cases}1, & \text{if } i\in A\\ 1-\prod\limits_{j\in V}(1-w_{ji}\pi(j)), & \text{otherwise.}\end{cases}\qquad(9)$$

That means in order to let node i assimilate the information, it must receive the information from at least one of its neighbors. Then, the sum of steady-state assimilation probabilities of all nodes can reach the desired influence spread.

Yang et al.[58] noted that (9) is not strictly applied to some situations. For example, it is invalid when the network has structural-defect node pairs. More importantly, there are some difficulties in solving systems of nonlinear equations, such as convergence and multiple solutions. They illustrated an observation that influence propagation probabilities in real-world social networks are usually quite small. Then, they represented the steady-state probability approximation by a linear system defined as

$$\pi(i)=\sum_{j\in V}w_{ji}\pi(j).\qquad(10)$$

They also proposed a simple iterative algorithm to solve the linear system problem. We can see (10) is similar to (2). This indicates that the influence and authority should have a latent relationship, which we will discuss in the next subsection.

However, in many scenarios, the network where diffusions take place is in fact implicit or even unknown. For example, in viral marketing settings, we only observe people purchasing products without explicitly knowing who was the influencer that caused the purchases. Thus, Yang and Leskovec[59] studied modeling information diffusion in implicit networks. They focused on modeling the global influence of a node on the rate of diffusion through the (implicit) network over time. Every node u has a particular non-negative influence function $I_u(l)$ which can be considered as the number of followup mentions l time units after u adopted the information. Then the volume, $V(t)$, the number of nodes that mention the information at time t, is the sum of properly aligned influence functions of nodes.

$$V(t)=\sum_{u\in A(t)}I_u(t-t_u)\qquad(11)$$

where $A(t)$ denotes the set of already active nodes that got activated prior to time t, i.e., $t_u\leq t$. They proposed a non-parametric approach to implement the influence function.

More methods to estimate the influence spread when dealing with the influence maximization problem will be introduced in Section 5.2.

4.2.3 PageRank with prior

Xiang et al.[44, 60] further understood PageRank from the perspective of influence propagation to explore the relationship between authority and influence. Specifically, they first proposed a linear social influence computation model as follows.

Definition 1. Denote the influence from node i to j by $f_{i\rightarrow j}$, then

$$f_{i\rightarrow i}=\alpha_i,\quad \alpha_i>0\qquad(12)$$

$$f_{i\rightarrow j}=\frac{1}{1+\lambda_j}\sum_{1\leq k\leq n}w_{kj}f_{i\rightarrow k},\quad \text{for } j\neq i\qquad(13)$$

where α_i is a prior probability value and $\lambda_j\in[0,+\infty)$ is a damping factor[44].

Equestion (13) shows that the influence from node i to j is proportional to the linear combination of its influence on j's neighbors. That's to say, if i wants to affect j, he can successfully affect k and then k will affect j with a probability of w_{kj}. α_i can be considered as the prior probability for node i to propagate the information. For example, in viral marketing, $\alpha_i=1$ means node i is a seed node and agrees to promote the product. λ_j indicates how much the influence will be blocked by node j. For simplicity, the authors set $\lambda_j=\lambda$ for each node j. When $\lambda_j=0$ and $\alpha_i=1$, (13) degrades into a linear approximation method for the IC model[58]. Besides, the authors said the above model can also approximate the non-linear stochastic influence model[57] by setting $\lambda_j\in[0,1]$ and $\alpha_i=1$ carefully.

The authors noticed that PageRank is actually a special case of the linear social influence model in Definition 1 with an appropriate priority. This shows the reasonableness of taking PageRank as a baseline in social influence related applications[57, 54, 61]. Moreover, the authors found individual influence $f_{i\rightarrow V}=\sum_{j\in V}f_{i\rightarrow j}$ has an upper bound under their model, which can be exploited to accelerate the selection of top-K influential nodes, even for the topic-sensitive task[44]. Based on the influence computation model in Definition 1, they further proposed independent social influence and group PageRank, which will be shown in the next two subsection.

4.2.4 Independent social influence

Actually, influences of different nodes may have overlaps that affect the same part of other nodes. For instance, in a social network, users u and v are adjacent, and u is one of the most influential users. If affecting u successfully, v can affect more others with the help of u, and thus its observed influence is much larger than the real value. Liu et al.[62] noted this scenario and tried to compute the independent social influence based on the linear model in Definition 1. They introduced the following definition of independent social influence.

Definition 2. Denote the influence from node $i \in S$ to j (independent from other nodes in S) by $f^{S\backslash i}_{i\to j}$, then

$$f^{S\backslash i}_{i\to i} = 1 \tag{14}$$

$$f^{S\backslash i}_{i\to j} = 0, \quad j \in S\backslash i \tag{15}$$

$$f^{S\backslash i}_{i\to j} = d \sum_{1\leq k\leq n} w_{kj} f_{i\to k}, \quad j \notin S \tag{16}$$

where $d \in (0, 1]$ is a damping factor[62].

From the difference with Definition 1, we can see that $f^{S\backslash i}_{i\to j}$ is essentially the influence of i on the network when other nodes in S are "removed" from the diffusion. Thus, the "removed" nodes will stop receiving and forwarding the information from i. The authors found the proposed independent influence has two interesting properties: 1) The influence of a set of nodes is actually the sum of each node′s independent influences. This is consistent with our intuition. 2) Someone′s independent influence has an upper bound. Based on these two properties, they also demonstrated two practical applications: rank the seeds according to their independent influence to figure out the contribution of each selected seed, and quickly find the top-K influential nodes from the seed nodes S.

4.2.5 Group PageRank

Liu et al.[63] provided a bounded linear approach for influence computation, called Group PageRank. They first extended Definition 1 of influence to a set on another node.

Definition 3. Denote the influence from a node set S to node j by $f_{S\to j}$, then

$$f_{S\to j} = 1, \quad \text{if}\, j \in S \tag{17}$$

$$f_{S\to j} = d \sum_{1\leq k\leq n} w_{kj} f_{S\to k}, \quad \text{otherwise} \tag{18}$$

where $d \in (0, 1]$ is a damping factor[63].

Then they found that the influence from S to T, $f_{S\to T} = \sum_{i\in T} f_{S\to i}$ has a upper bound $GPR(S, T)$, which is called Group PageRank.

$$f_{S\to T} \leq \frac{|T|}{1-d} \sum_{i\in S} (1 - d \sum_{k\in S} t_{ki}) fPR_i \triangleq GPR(S, T) \tag{19}$$

where fPR_i is the PageRank value of node i and can be computed by (7). They have several interesting conclusions. First, Group PageRank is also a generalization of PageRank because when $|S| = 1$, $GPR(S, T)$ is proportional to fPR_i. Second, $GPR(S, T)$ is essentially the sum of each single PageRank of nodes in S with a "discount". That means the mutual influences between the nodes in S are removed when estimating the influence spread of S. Third, $GPR(S, T)$ only depends on fPR_i. If computing fPR_i for each node in advance, we can quickly get Group PageRank for every node set in $O(|S|^2)$.

In summary, getting the exact value of influence is hard, and thus many approximate methods have been proposed to simplify the computation process and improve the efficiency.

5 Influence maximization

In this section, we will show how to solve the influence maximization problem based on information diffusion models and influence evaluation.

In a social network such as Twitter, which users should be selected to offer discounts and then let them promote a new product through the word-of-mouth effect? Given the water distribution network in a city, where should we place sensors to quickly detect contaminants? Both of them can be formalized as the influence maximization (IM) problem that selects a set of seed nodes to maximize the expected number of active nodes at the end of diffusion process. It was first noted by Richardson and Domingoes[6] when they mined knowledge-sharing sites for viral marketing. Then Kempe et al.[4] formulated it as the following discrete optimization problem.

Problem 1 (**Influence manimization**).

In a social network $G(V, E)$, influence maximization is to select a seed node set S of size K such that

$$S = \arg\max_{S\subset V} f(S), \qquad \text{s.t.} \quad |S| = K$$

where $f(S)$ is the influence spread of S in this network[4].

There are two intuitive solutions: one is enumerating and selecting the subset with the maximal influence spread. This will lead to combinatorial explosion and is not applicable to large-scale networks. The other is selecting top-K nodes with maximal influences, but different individual influences may overlap with each other so that their collective influence is not the maximal. Kempe et al.[4] claimed that influence maximization is NP-hard under the independent cascade (IC) model and linear threshold (LT) model. Therefore, many researchers focus on this problem due to its wide applications and propose various approximation methods to speed up the solutions, which can be divided into four categories: greedy, heuristic, reverse sampling and other algorithms.

5.1 Greedy algorithms

Kempe et al.[4] noted that the influence spread function f under the IC and LT model is monotone and submodular.

Definition 4 (**Monotonicity**). A set function $f : 2^V \to \mathbf{R}$ is monotone if $f(S) \leq f(T)$ such that $S \subset T \subset V$[4].

Definition 5 (**Submodularity**). A set function $f : 2^V \to \mathbf{R}$ is submodular if it satisfies

$$f(S \cup \{u\}) - f(S) \geq f(T \cup \{u\}) - f(T)$$

for any u and $S \subset T$[4].

For a non-negative, monotone and submodular function f, let S be a set of size k obtained by selecting elements one at a time which provides the maximal marginal increase of the function value. Let S^* be the optimal set that maximizes the value of f over all k-element sets. Nemhauser et al.[64] have shown that

$$f(S) \geq (1 - \frac{1}{e}) f(S^*) \tag{20}$$

i.e., S provides a $(1 - \frac{1}{e})$-approximation.

Algorithm 1. Framework of the greedy algorithm

```
Input:
    $G(V, E, W)$ – the network with $W = [w_{ij}]$
    $K$ – the number of seed nodes
Output:
    $S$ – the seed set
1) $S = \varnothing$
2) while $|S| \neq K$ do
3)     $u = \arg\max_{v \in V \setminus S}(f(S \cup \{v\}) - f(S))$
4)     $S = S \cup \{u\}$
5) end while
6) return $S$
```

Therefore, Kempe et al.[4] proposed a framework of the greedy algorithm to select seed nodes one by one and its pseudo codes are listed in Algorithm 1. It starts with an empty seed set. In each iteration, we compute the marginal influence gain for each node, and then the node which provides the largest marginal influence gain is selected into the set (i.e., Lines 3 and 4).

Definition 6 (**Marginal influence gain**). Given a node set, the marginal influence gain of node u, $\Delta_u(S)$, is the increase value of influence spread of S if u is added into S, i.e., $\Delta_u(S) = f(S \cup \{u\}) - f(S)$[4].

This simple algorithm provides an amazing performance guarantee that it can approximate the problem with a ratio of $(1 - \frac{1}{e})$, as long as the influence spread function f has the two properties (i.e., monotonicity and submodularity) at the same time. That's to say, the influence spread of the outputted seed set of Algorithm 1 is provably within $(1 - \frac{1}{e})$ of the optimal value.

We can see the bottleneck of Algorithm 1 is to evaluate the influence spread of a seed set (i.e., the value of f). Kempe et al. ran Monte Carlo (MC) simulations for R times to estimate its value under the IC or LT model, as described in Section 4.2.1. Thus, the Monte Carlo simulation will be executed $KR|V|$ times in total so that the above greedy algorithm is time-consuming and prohibitive for large-scale networks.

5.1.1 Lazy evaluation

Leskovec et al.[24] exploited the submodularity to avoid unnecessary recalculations of the marginal influence gains in each iteration, and developed an efficient algorithm, namely cost-effective lazy forward (CELF) selection. It is based on the diminishing returns property that the earlier a node is selected into the seed set, the larger marginal influence gain it can achieve. That means for a node $u \in V$ and $S_k \subseteq S_{k+1} \subseteq V$:

$$\Delta_u(S_k) \geq \Delta_u(S_{k+1}) \tag{21}$$

where S_k is the seed set after the k-th iteration.

Algorithm 2. Cost-effective lazy forward (CELF)[33]

```
Input:
    $G(V, E, W)$ – the network with $W = [w_{ij}]$
    $K$ – the number of seed nodes
Output:
    $S$–the seed set
1) $S = Q = \varnothing$
2) for $u \in V$ do
3)     $u.mig = \Delta_u(S) = f(S \cup \{u\}) - f(S)$
4)     $u.round = 0$
5)     add $u$ to $Q$ in the decreasing order of $*.mig$
6) end for
7) while $|S| \neq K$ do
8)     $u$ = the first element in $Q$
9)     $Q = Q - \{u\}$
10)    if $u.round == |S|$ then
11)        $S = S \cup \{u\}$
12)    else
13)        $u.mig = \Delta_u(S) = f(S \cup \{u\}) - f(S)$
14)        $u.round = |S|$
15)        add $u$ to $Q$ again in the decreasing order of
           $*.mig$
16)    end if
17) end while
18) return $S$
```

Its pseudo codes are shown in Algorithm 2. Specifically, it initializes each node's marginal influence gain (i.e.,

$*.mig$ in Algorithm 1), which are added into a list Q in the decreasing order of $*.mig$. In each iteration, assuming node u has the largest $.mig$ in Q, we recompute its real marginal influence gain relative to the current seed set S by Monte Carlo simulations. Then, we can adopt the lazy evaluation strategy: If u's new gain is still larger than other nodes' in Q, we can select u into S and jump into the next iteration, which thus avoids unnecessary recalculations of other nodes' marginal influence gains. Otherwise, we update $u.mig$ with the new gain and repeat the above steps.

Eventually, CELF not only keeps the performance with a ratio of $(1-\frac{1}{e})$, but also achieves 700 times speedup, compared with the basic one in Algorithm 1. Goyal et al.[65] further optimized Algorithm 2 based on the submodularity property of the spread function in influence propagation models, and introduced CELF++ which has an improvement of CELF by 17%–61%.

Besides, Liu et al.[63] explored Group PageRank in Section 4.2.5 as the influence spread and adopted the greedy framework to solve this problem. In the (k+1)-st iteration, node u's marginal influence gain is $\Delta_u(S_k) = GPR(S_k \cup \{u\}, V) - GPR(S_k, V)$, where $GPR(S, V)$ is defined by (19) and only depends on fPR_i. After we get fPR_i for each node in the initialization, $GPR(S, V)$ can be quickly obtained by looking up the buffer. The authors have shown that this gain also follows (21) and has the diminishing returns property. Therefore, they applied the above lazy evaluation into practice and the experimental results show their method is more efficient than two heuristic algorithms, namely influence ranking and influence estimation (IRIE)[66] and prefix excluding maximum influence arborescence (PMIA)[55] which will be discussed in the next subsection.

5.2 Heuristic algorithms

Although aforementioned methods exploit the lazy evaluation to speed up the greedy algorithm, their running time on large-scale networks is still very high. Therefore, many researchers start to develop heuristic algorithms to further improve the efficiency of influence spread evaluation according to properties of specific diffusion models.

5.2.1 Shortest path

Due to the hardness of getting an exact calculation or a good estimate of influence spread, Kimura and Saito[67] proposed two models, shortest-path model (SPM) and shortest-path-1 model (SP1M), to simplify the IC model and to efficiently obtain good approximate solutions to the influence maximization problem, when the propagation probabilities through links are small. In SPM, each node v has the chance to become active only at step $t = d(S, v)$, where $d(S, v)$ is the topological distance from S to v. That means each node is activated only through the shortest paths from an initial active set. Namely, SPM is a special type of the IC model where only the most efficient information can spread. While in SP1M, each node v has a chance to become active only at steps $t = d(S, v)$ and $t = d(S, v) + 1$. They showed that the exact value of influence spread in SPM and SP1M can be computed efficiently.

More importantly, adopting the greedy framework in Algorithm 1 can guarantee the output with a ratio of $(1-\frac{1}{e})$ in the SPM and SP1M diffusion models. But a critical drawback of SPM and SP1M is that they ignore the influence probabilities among users and only consider the topological structure.

5.2.2 Degree discount

When selecting the seed nodes one by one, Chen et al.[68] explored the effect of the selected seed nodes on the rest nodes. They adopted the node degree to estimate its influence and proposed two degree discount heuristics to diminish that effect.

1) SingleDiscount: Each neighbor of a newly selected seed will discount its degree by one. This heuristic can be applied to all information diffusion models.

2) DegreeDiscountIC: This is a more accurate degree discount heuristic for the IC model with a small propagation probability p. When selecting v into the seed set, the increase of the expected number of active nodes is

$$1 + (d_v - 2t_v - (d_v - t_v)t_v p + o(t_v)) \times p \qquad (22)$$

where d_v is the degree of v, and t_v is the number of v's neighbors that are already selected as seeds. The larger t_v is, the more discount d_v will get.

They have shown that the above degree discount heuristics achieve much larger influence spread than classic degree and centrality-based heuristics. What's more, DegreeDiscountIC achieves almost equal influence thread with the greedy algorithm when tuned for a specific influence cascade model. However, they have no performance guarantee for general graphs.

5.2.3 Maximum influence path

Chen et al.[55] extended SPM and SP1M by considering maximum influence paths (MIP) instead of shortest paths, to approximate the actual expected influence within the social network. Its main idea is to use local arborescence structures of each node to approximate the influence propagation.

Specifically, a maximum influence path between node u and v is the path with the maximum propagation probability from u to v. They first computed maximum influence paths between every pair of nodes in the network via a Dijkstra shortest-path algorithm, and ignore MIPs with probabilities smaller than an influence threshold θ, which can effectively restrict the influence computation into a local region. Then they aggregated MIPs starting or ending at each node into the arborescence structures, which represent the local influence regions of each node. Different values of θ control the size of these local influence regions. Thus this heuristic method is able to achieve tunable balance between efficiency (in terms of running time) and effectiveness (in term of influence spread). The authors only considered the influence propagated through these local arborescences, and referred to this model as the maximum influence arborescence (MIA) model[55].

When the graph is sparse and the propagation probabilities on edges are small, to improve the efficiency, they provided a variant of MIA, called prefix excluding MIA (PMIA) with a batch update[55]. When selecting the next seed, for every node v, PMIA recomputes its in-arborescence so that every seed candidate $w \in V \backslash S$ has a path to v while not passing through any seed in S. As a result, all selected seeds form a sequence S according to the selection order, so that any seed s in S has alternative paths to all nodes v that do not pass through any seed in the prefix of S proceeding s.

Moreover, they have proved that the influence spreads in the MIA and PMIA models are submodular and monotone[55]. Therefore, adopting the greedy algorithm in previous subsection under the MIA and PMIA model can also approximate the problem with a ratio of $(1-\frac{1}{e})$. Results from extensive simulations on several real-world and synthetic networks demonstrate that their algorithm was the best scalable solution to the influence maximization problem at that time.

After that, many works try to further extend the above algorithm, e.g., IRIE[66], local directed acyclic graph (LDAG)[56] and simple path (SIMPATH)[69]. IRIE integrates a new message passing based influence ranking (IR), and influence estimation (IE) methods for influence maximization in both the independent cascade (IC) model and its extension IC-N that involves negative opinion propagations. In each round of selecting a seed node, the greedy algorithm uses Monte Carlo simulations while PMIA uses more efficient local arborescence based heuristics to estimate the influence spread of every possible candidate. This is especially slow for the first round where the influence spread of every node needs to be estimated. Therefore, Jung et al.[22] proposed a novel global influence ranking (IR) method derived from a belief propagation approach, which uses a small number of iterations to generate a global influence ranking for the nodes and then selects the highest ranked node as the first seed. To avoid the overlapping influence, they integrated IR with a simple influence estimation (IE) method, so that after one seed is selected, they can estimate additional influence impact of this seed to other nodes in the network, and then use the results to adjust next round computation of influence ranking. IE is much faster than directly estimating marginal influence gain of many seed candidates. When combining IR and IE together, we obtain the fast IRIE algorithm.

On the other hand, LDAG[56] and SIMPATH[69] are tailored for the LT model. LDAG exploits the fact that computing influence spread in directed acyclic graphs (DAGs) can be done in linear time. It constructs a local DAG surrounding every node v in the network, and restricts the influence to v within the local DAG structure. This makes influence computation tractable and fast on a small DAG. Then the authors combine the greedy algorithm with a fast scheme that updates the incremental influence spread of every node. While SIMPATH builds on the fact that the spread of a set of nodes can be calculated as the sum of spreads of each node in the set on appropriate induced subgraphs under the LT model. It iteratively selects seeds in a lazy forward manner like CELF. Instead of using expensive MC simulations to estimate the spread, it can be computed by enumerating the simple paths starting with the seed nodes within a small range of neighborhood, where the majority of the influence flows since probabilities of paths diminish rapidly as they get longer.

In general, these heuristic algorithms are more efficient for large-scale networks through properties of specific diffusion models, but few of them can keep the performance guarantees under the standard IC and LT models described in Section 3.

5.3 Reverse sampling algorithms

Recently, Borgs et al.[70] made a theoretical breakthrough and inspired many researchers to solve the influence maximization problem from a quite different perspective of reverse sampling, which has approximation guarantees and is even more efficient than the above heuristic algorithms.

We first introduce two concepts for better explanation.

Definition 7 (**Reversa reachable set**). Let v be a node in G, and g be a graph obtained by removing each edge e in G with $1-w_e$ probability. The reverse reachable (RR) set for v in g is the set of nodes in g that can reach v (That is, for each node u in the RR set, there is a directed path from u to v in g.)[71].

Definition 8 (**Random RR set**). Let G be the distribution of g induced by the randomness in edge removals from G. A random RR set is an RR set generated on an instance of g randomly sampled from G, for a node selected uniformly at random from g[71].

Borgs et al.[70] proposed a reverse influence sampling (RIS) method under the IC model. It runs in two steps:

1) Generate a certain number of random RR sets from G.

2) Use the standard greedy algorithm for the maximum coverage problem[72] to select k nodes to cover the maximum number of RR sets generated.

Its main idea is if a node u appears in a large number of RR sets, then it should have a high probability to activate many other nodes under the IC model; in that case, u's influence spread should be large. More importantly, RIS can return a $(1-\frac{1}{e}-\epsilon)$-approximate solution with at least $1-n^{-l}$ probability in $O(kl^2(m+n)\log^2 \frac{n}{\epsilon^3})$ time. They also shown it is near-optimal since any other algorithm guarantees the same approximation rate and succeeds with at least a constant probability must run in $\Omega(m+n)$ time (i.e., the lower bound).

However, RIS has a large hidden constant factor in its time complexity so that its practical efficiency is rather unsatisfactory. Tang et al.[71] borrowed ideas from RIS and proposed a two-phase influence maximization (TIM) algorithm. It first computes a lower-bound of the maximum expected influence spread among all size-k node sets and then uses the lower-bound to derive a parameter θ. Then it samples θ random RR sets from G, and derives a size-k node set that covers a large number of RR sets like RIS. It can return a $(1-\frac{1}{e}-\epsilon)$-approximate solution with at least $1-n^{-l}$ probability in $O((k+l)(m+n)\log\frac{n}{\epsilon^2})$ expected time. TIM$^+$ improves TIM by adding an intermediate step that heuristically refines θ into a tighter lower bound and leads to higher efficiency. After that, Tang et al.[73] designed another method, influence maximization via martingales (IMM) to further improve the efficiency. It has the same performance guarantees with TIM and TIM$^+$, but offers significantly improved empirical efficiency and can be extended to a larger class of diffusion models. The experimental results show IMM is often faster in orders of magnitude than the states of the art in terms of computation efficiency, including heuristic algorithms such as IRIE[66] and SIMPATH[69]. Meanwhile, Cohen et al.[74] designed a sketch-based influence maximization (SKIM) algorithm.

Nguyen et al.[75] designed two novel sampling algorithms SSA and D-SSA, aiming to achieve minimum number of RIS samples. However, Huang et al.[76] revised their work and discovered inaccuracies in previously reported technical results on the accuracy and efficiency of SSA and D-SSA, which was set right by then. They presented a revised version of SSA, dubbed SSA-Fix that restores $(1-\frac{1}{e})$-approximation at the cost of increased computation overheads. The experimental results show that SSA and D-SSA are more efficient than IMM when k is large under the IC and LT models. They suggested that there exists opportunities for further scaling up influence maximization with approximation guarantees.

5.4 Other algorithms

There are various other algorithms for influence maximization, and here we demonstrate four typical methods which may bring us new perspectives.

First, now that evaluating influence spread on the whole network is time-consuming, can we just deal with it on the community-level? A community is a densely connected subset of nodes that are only sparsely linked with the remaining network[15]. Wang et al.[77] noted this idea and proposed a community-based greedy algorithm (CGA), for mining top-K influential nodes in mobile social networks, following the divide-and-conquer principle. Specifically, they first extended a community detection method so that it can divide the network into communities based on information diffusion models. Then they proposed a dynamic programming method to incrementally choose the communities to be processed. Within a community, we can adopt any existing algorithm to detect influential nodes, such as PageRank and CELF. Besides, they have proved that CGA obtains a $(1-\mathrm{e}^{-\frac{1}{1+\theta\Delta d}})$-approximation, where θ is the threshold used in the community detection and Δd is the maximal difference between the number of nodes affected by a node in the network and that in a community. When $\theta=0$, the number of generated communities will be 1, which means all communities will be combined into one, CGA is the same as the original greedy algorithm with $(1-\frac{1}{e})$-approximation.

Second, Wang et al.[78] noticed that influence maximization finds some influential nodes whose influences can cover the whole network, which is similar to selecting some informative rows to reconstruct a matrix. Thus, they proposed a novel framework, named data reconstruction for influence maximization (DRIM), from the perspective of data reconstruction. They first constructed an influence matrix, each row of which is the influence of a node on other nodes. Instead of using time-consuming Monte Carlo simulations to estimate the influence spread, they turn to the linear social influence model in Definition 1, which gives us a closed-form solution to the influence of each node. Then, they selected the most informative k rows to reconstruct the matrix and their corresponding nodes are the seed nodes which could maximize the influence spread. The experimental results show that the proposed framework is at least as effective as the traditional greedy algorithm[4]. However, this framework has no performance guarantee and its time complexity is too high.

Third, Jiang et al.[79] proposed a totally different approach based on simulated annealing (SA) to the influence maximization problem. Simulated annealing simulates the process of metal annealing and optimizes the solutions to a number of NP-hard problems. The proposed SA based algorithm for influence maximization problem will converge towards optimum as the iteration number grows larger. SA can escape the local optimum and is able to learn to improve the influence spread of solution set automatically. They also designed two heuristic methods to accelerate the convergence process of SA, and a new method of computing influence to speed up the proposed algorithm.

Finally, indeed, users' influences and the network structure are dynamic over time. The previous works are done only in static networks. Rodriguez and Schölkopf[80] focused on influence maximization in continuous time diffusion networks. They described how continuous time Markov chains allow us to analytically compute the average total number of nodes reached by a diffusion process starting in a set of seed nodes. They also showed that selecting a set of most influential source nodes in the continuous time influence maximization problem is NP-hard, and developed an efficient approximation algorithm with provable near-optimal performance. Wang et al.[81] studied the incremental influence maximization for dynamic social networks. They designed an incremental algorithm, dynamic influence maximization (DIM), for the linear threshold model. It consists of two phases: initial seeding and seeds updating. They also proposed two pruning strategies for the seeds updating phase to further reduce the running time. While Wang et al.[81] tried to track the influential nodes in dynamic networks. They modeled a dynamic network as a stream of edge weight updates, which embraces many practical scenarios as special cases, such as edge and node insertions, deletions as well as evolving weighted graphs. Their key idea is to use the polling-based methods and maintain a sample of random RR sets so that we can approximate the influence of nodes with provable quality guarantees.

5.5 Variants of influence maximization

There have been many variants of the classical influence maximization for different applications. Here we will briefly discuss some of them and hope to attract more readers for further study.

First, try to generalize the influence maximization problem or add more constrains to the original formulation in Problem 1. For example, budgeted influence maximization (BIM) is identifying a small set of influential individuals who can influence the maximum number of members within a limited budget. It was formally described by Kempe et al.[4] and attracted much attention later[82, 83]. While Yang et al.[84] took a step further and proposed the continuous influence maximization (CIM) problem. It deals with a real-world scenario: Imagine we are introducing a new product through a social network, in which we can get the purchase probability curve with respect to discount for each user in the network. Based on that, it can be decided what discount should be offered to those social network users so as to maximize purchases under a predefined budget. We can see CIM is a generalization of influence maximization (IM) and BIM. Besides, Aslay et al.[85] studied the revenue maximization problem in incentivized social advertising. It is to allocate advertisements to influential users with the rational goal of maximizing its own revenue. They consider the propensity of advertisements for viral propagation, and carefully apportion the monetary budget of each of the advertisers between incentives to influential users and ad-engagement costs.

Second, in many real-world cases, marketers usually target certain products at particular groups of customers. For example, a cosmetic company would want its products to attract more women than men. Li et al.[86] formulated the above as a labeled influence maximization problem, which aims to find a set of seed nodes to trigger the maximum spread of influence on the target customers in a labeled social network. The label information is widely available in current social networks, by which users describe their personal interests, graduated colleges, hometown, age, skills, etc. Tang et al.[87] considered the magnitude of influence and the diversity of the influenced crowd at the same time, and formulated it as the diversified influence maximization problem. An obvious case is that this could reduce the risk of marketing campaigns, as the proverb goes: “Don′t put all your eggs in one basket”. Besides, Liu et al.[88] combined targeted marketing with viral marketing to build a better and stronger marketing business. Targeted marketing identifies typical customers and concentrates marketing efforts on these customers, which could make the promotion of items much easier and more cost-effective. They studied the problem of maximizing information awareness in viral marketing with constrained targets.

Third, Wang et al.[89] considered both active nodes and informed nodes that are aware of the information when they study the coverage of information propagation in a network. They proposed a new problem called information coverage maximization that aims to maximize the expected number of both active nodes and informed ones, and showed this problem is NP-hard and submodular in the IC model. After that, they further studied the activity maximization problem[90] which selects a set of seed users to maximize the expected total amount of excitements for a piece of new information. It is substantially different from the renowned influence maximization problem and cannot be tackled with the existing approaches. In a social network, the excitements among different users even at the same information are different. They aim to find an optimal set of seed users under a given budget, and start information propagation from the seed users so as to gather the maximum sum of activity strengths among the influenced users.

Finally, sometimes, more than one type of information such as different information about competitive products is spreading in social networks. He et al.[91] focused on the blocking maximization problem under the competitive linear threshold (CLT) model, which states that one entity would try to block the influence propagation of its competing entity as much as possible by strategically selecting a number of seed nodes that could initiate the propagation by themselves. They extended LDAG[56] and designed an efficient algorithm competitive local directed acyclic graph (CLDAG) which utilizes the properties of the CLT model, to address this issue. Besides, it is supposed that one of the competitors could enhance its influence by creating new links. A natural question is, when the number of new links is limited due to limited resource, how to add these links so as to maximize the influence of the given competitor over the others (called competitiveness). Zhao et al.[92] formulated it as a competitiveness maximization problem on complex networks. They take two cases into consideration: maximize the number of supporters of the competitor and maximize the total supporting degree of normal agents toward the competitor. Besides, many individuals also care about the influence of themselves and want to enhance the influence. Thus, Ma et al.[93] considered an individual influence maximization problem that maximizing the target individual influence by recommending new links.

6 Information source detection

The purpose of influence maximization is to find a small set of seed nodes to maximize the expected number of activated users. But when observing which nodes are active after a piece of information has diffused in the network, can we infer the source or seed nodes triggering this observed diffusion result? For example, after a rumor has spread among the network, we want to find the rumor source nodes to stop its dissemination. This problem is called information source detection (or patient-zero), which can be considered as the reverse process of information diffusion. It also has attracted many researchers to study, due to its wide range of applications such as epidemic outbreak prevention[26–28] and rumor source tracing in social networks[30, 29].

After the information starts to spread in the network G from an unknown source node set S^* at time t_0, there will be many nodes being infected till time $t(t \geq t_0)$. Note that we assume every node usually has three possible states: infected (i.e., active), susceptible (i.e., inactive) and recovered, like epidemic models. Let G_I denote the infected subgraph $G_I(V_I, E_I)$ which consists of infected nodes V_I and their inter-edges E_I. $P(G_I|S)$ represents the likelihood to observe G_I if the information starts to diffuse from the node set S. Information source detection aims to identify the source nodes initiating the diffusion process based on the observed node states and the network structure, which can be formally defined as follows:

Problem 2 (**Information source detection**). Given graph $G(V,E)$ and the infected subgraph $G_I(V_I, E_I)$ observed at time $t(t \geq t_0)$, information source detection is to select set of source nodes $\hat{S}$ such that $\hat{S} = \arg\max P(G_I|S)$, i.e., the largest likelihood to observe the infected subgraph. t_0 is the unknown time the information started to spread.

For example, in Fig. 4, we observe seven infected nodes and want to identify the source node. This problem is challenging due to many aspects. First, we often observe only one snapshot of the network and get the states of some nodes, which is just a part of the whole diffusion process. That means we just know which nodes are infected, but cannot distinguish the propagation paths that indicate who infects who and when they are infected. Second, the actual information diffusion laws are unknown, which cannot be comprehensively described by the models in Section 3. Third, information diffusion is highly dynamic and has a great variety of patterns when initiated from different sources. For instance, a photo will be shared for many times if it is posted by a celebrity in social networks. Forth, there are usually multiple source nodes in real-world scenarios, while the number is unknown. Finally, the time-stamp t_0 when the information started to diffuse and how long it has lasted, are also unavailable.

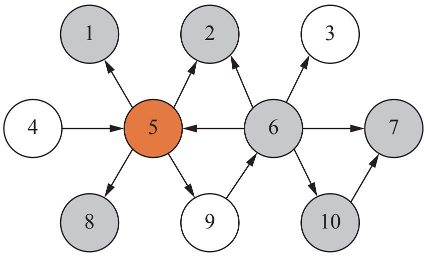


Fig. 4 Information diffusion in a toy social network, where the orange node 5 is the source node and gray nodes are infected, while others are susceptible. Edges indicate the directions of information flows. Color versions of the figures in this paper are available online.

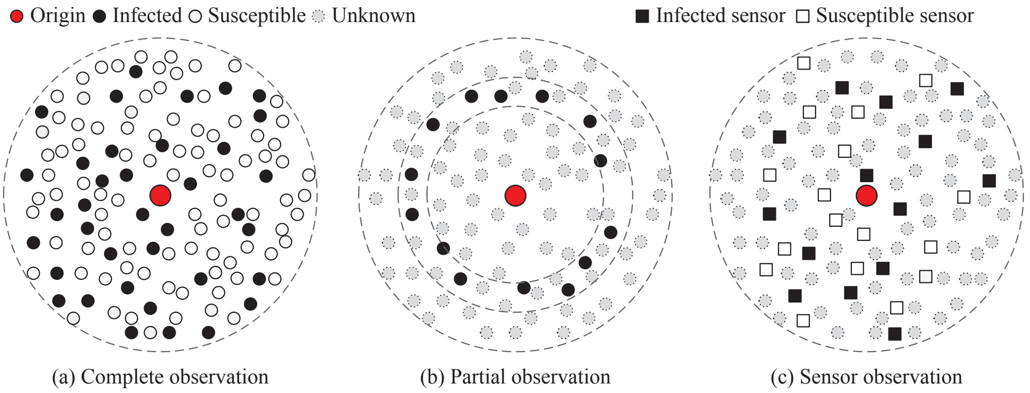
Shah and Zaman[35] are among the first to consider this problem. After that, many efforts have been devoted to different cases, which can be divided into three categories[94] according to the observed node states: complete observation partial observation sensor observation. Fig. 5 shows three examples of observed diffusion results for each category. In the next part, we will briefly describe the corresponding solutions to detect the source nodes of the observed three categories in recent years.

6.1 Detection with complete observation

In this subsection, we introduce some detection methods with the complete observation. It means when observing the diffusion at time t after the information has spread, we will get the complete states of all nodes in the whole network. That's to say that we can identify which nodes are infected, and which are recovered or still susceptible.

6.1.1 Rumor center

When noticing the source detection problem in their seminal work, Shah and Zaman[35] provided a systematic study on finding the source of a computer virus in a network. They assumed there is only one source node and described the virus spreading in a network with the SI model, a variant of the popular SIR model. Then they constructed the following maximum likelihood estimator


● Origin ● Infected ○ Susceptible ○ Unknown
■ Infected sensor □ Susceptible sensor

(a) Complete observation (b) Partial observation (c) Sensor observation

Fig. 5 Three examples[94] of observed diffusion results corresponding to three categories of observation for information source detection in a network, whose edges between nodes are hidden for brevity

for the virus source.

$$\hat{v} = \arg\max_{v\in V_I} P(G_I|v^* = v) \tag{23}$$

where $\hat{v}$ and v^* are the detected source node and actual source node respectively. They showed that in a regular tree, the above estimator equals to select a node with the maximal rumor centrality, $R(v, G_I)$, which is the number of permitted permutations of nodes that result in a spanning tree T of G_I and begin with node $v \in V_I$. A permitted permutation is a permutation of the nodes in V_I subjected to the ordering constraints set by the network structure. Thus, the detected source node $\hat{v}$ is called a rumor center.

Luckily, they found the rumor centrality $R(v, G_I)$ of node v has a simple expression for trees:

$$R(v, G_I) = |V_I| \prod_{u\in V_I} \frac{1}{T_u^v} \tag{24}$$

where T_u^v is the number of nodes in the subtree rooted at node u with node v as the source. They also designed an efficient message-passing algorithm to compute the rumor centrality for each node, running in $O(|V_I|)$ time. However, they further noted that permitted permutation is also known as a linear extension of the poset, while counting its number falls in the complexity class #P-complete in general graphs[30]. To extend their method to general graphs, they assumed the virus spreads from node v along a breadth first search (BFS) tree rooted at v, $T_{bfs}(v)$, and detected the rumor center with the maximal rumor centrality $R(v, T_{bfs}(v))$. Besides, they proved that the rumor center is equivalent to the distance center on a tree. What's more, on trees which grow faster than a line, the estimator in (23) always has non-trivial detection probability, whereas on trees that grow like a line, the detection probability will go to 0 as the network grows.

Their method has several limitations in some aspects. First, it is only applicable for the case when there is one source node. Second, it only considers the infected subgraph and neglects other uninfected nodes which are also important for detecting the source. Third, rumor centrality assumes that the probabilities of all permitted permutation are equal for general graphs.

After that, some researchers tried to improve this method. Dong et al.[95] constructed a maximum a posteriori (MAP) estimator to detect a single source from many suspect nodes under the SI model. A priori knowledge will indicate the set of suspect nodes V_s in different cases. For example, V_s with cardinality k forms a connected subgraph of G, or V_s contains only two suspect nodes separated by their shortest path distance. Then they proposed a local rumor center, which is a generalized rumor centrality, to identify the source from suspects. For regular tree-type networks with node degree, they also characterized the correct detection probability of the source estimator upon observing some infected nodes, in both the finite and asymptotic regimes. Inspired by the derivation of rumor center, Chang et al.[26] proposed a greedy method to estimate the likelihood $P(G_I|S)$. Its basic idea is to find the upper bound of the probability of permitted permutations that start with the same node.

Besides, Wang et al.[96] addressed the problem of single rumor source detection with multiple independent observations under the SI model by joint rumor center. Suppose k different rumors originate from a common node in the network, which can be regarded as k times independent rumor spreading with the same rumor source. For each rumor spreading, we can observe a corresponding infected subgrpah. For regular tree graphs, they showed the detected source is a node that maximizes the product of its rumor centralities in each infected subgrpah. They found even with two observations, the detection probability at least doubles that of a single observation. Luo et al.[97] considered the problem of estimating the multiple infection sources and the infection regions (subsets of nodes infected by each source) in a network. They exploited the Voronoi partition to estimate the infection regions and combined two regions to find two source nodes with their source estimation method, an extension of rumor centrality. They proved that if there are at most two infection sources in a geometric tree, their estimator identifies the true source nodes with a probability going to one as the number of infected nodes increases. However, this method can hardly be used in the real world, especially on large-scale networks due to its high time complexity.

6.1.2 Eigenvector center

The second kind of source detection methods is based on the eigenvector center, which exploits the adjacent matrix analysis. For example, Fioriti and Chinnici[27] predicted the multiple sources of an outbreak with a spectral technique. They proposed to use the node dynamical importance (DI) which is the reduction of the largest eigenvalue of the adjacency matrix after a node has been removed, to assess the most prominent nodes of a network. They noted that a large reduction after the elimination of a node implies the node is relevant to the aging of an infection network. Dynamical importance (i.e., dynamic age) of node v is defined by

$$DI_v = \frac{|\lambda_m^{new} - \lambda_m|}{\lambda_m} \tag{25}$$

where λ_m and λ_m^{new} are the maximum eigenvalues of the adjacency matrix and the one after v is removed, respectively. The detected source nodes are those with the highest DI values. Results show that the spectral technique is able to identify the source nodes if the graph approximates a tree sufficiently.

Besides, based on the minimum description length (MDL) principle, Prakash et al.[28] proposed a novel method, NETSLEUTH, under the SI model. The total description length of a diffusion consists of two parts: cost of the model to identify the source nodes S and cost of describing the infected subgraph G_I given a source set S. NETSLEUTH can identify the set of seed nodes and virus propagation ripple which starts with those nodes and best describes the given snapshot. They showed we can easily optimize the description length of the virus propagation ripple for a given seed set by greedily maximizing the likelihood. For single source node, the likelihood has an upper bound, which can be maximized by finding the smallest eigenvalue of the Laplacian submatrix corresponding to the infected graph G_I. To find the next source node, they first remove the previous selected source node from the infected subgraph. Then, they repeat the above steps on the remaining subgraph until the MDL cost function stops decreasing. As a result, it can identify the best set of seed nodes in a principled manner, without choosing k, the number of seed nodes in advance. However, the computation of eigenvalues at each step makes this method not applicable for large-scale networks.

6.1.3 Sampling methods

The third kind of source detection methods is based on sampling to estimate the likelihood of observing the infected subgraph for each node. Different with previous methods, they focus on the stochastic diffusion models, such as the independent cascade (IC) model and linear threshold (LT) model. For example, Zhai et al.[98] designed a Markov chain Monte Carlo (MCMC) algorithm to find the single source of a cascade given the snapshot under the IC model. They formulated the detection as a source inference problem with maximum likelihood estimation like Problem 2, and proved its #P-completeness. Note that the generation of infected subgraphs corresponds to a specific distribution G_I. Because calculating the exact value of likelihood is #P-hard, they proposed to use the Metropolis algorithm to sample G_I in a Markov chain. When the MCMC chain converges, the stationary distribution will be G_I. After that, they counted the infected subgraphs which equals to the observed one of each node, and selected a node with the maximal value as the source node. However, this method is time-consuming when the number of infected nodes is large, and it is hard to judge the convergence of MCMC to stop the sampling. Zhang et al.[99] further extended this method for source detection under the LT model.

Besides, Nguyen et al.[100] proposed a new approach to identify multiple infection sources by searching for a seed set S that minimizes the symmetric difference between the cascade from S and V_I, a set of observed infected nodes. They designed an approximation algorithm, sampling-based infection sources identification (SISI), to identify infection sources without the prior knowledge on the number of source nodes. Inspired by other works[70, 71], SISI contains two key components: an efficient truncated reverse infection sampling (TRIS) to compute the objective with high accuracy and confidentiality, and an innovative transformation of the studied problem into a submodular cost covering problem to provide high quality solutions with performance guarantees. Note that SISI works for most progressive diffusion models, and has provable guarantee for the problem in general graphs.

6.1.4 Diffusion kernel

A diffusion kernel can represent diffusion processes in a given network, but computing this kernel is computationally challenging in general. Feizi et al.[101] proposed a path-based network diffusion kernel which considers edge-disjoint shortest paths among pairs of nodes in the network, and can be computed efficiently for both homogeneous and heterogeneous continuous-time diffusion models. They used this network diffusion kernel to solve the inverse diffusion problem, named network infusion (NI) with both likelihood maximization and error minimization. They applied this framework to both single-source and multi-source diffusion, and single-snapshot and multi-snapshot observations, using both uninformative and informative prior probabilities for candidate source nodes.

Pena et al.[102] casted the problem of source localization on graphs as the simultaneous problem of sparse recovery and diffusion kernel learning (SR-DKL). A l_1 regularization term enforces the sparsity constraint while they recover the sources of diffusion from a single snapshot of the diffusion process. The diffusion kernel is estimated by assuming the process to be as generic as the standard heat diffusion.

6.1.5 Others

Zhu and Ying[103] presented a new source localization algorithm under the independent cascade (IC) model, called the short-fat tree (SFT). Loosely speaking, the algorithm selects a node as the source such that the breadth-first search (BFS) tree from the node has the minimum depth but the maximum number of leaf nodes. They also established the performance guarantees of SFT for both tree networks and the Erdos-Renyi (ER) random graph. On tree networks, SFT is the maximum a posterior (MAP) estimator.

6.2 Detection with partial observation

In some scenarios, we can only observe the states of partial nodes at a given time t. Jiang et al.[94] summarized them as four cases.

1) Nodes reveal their states with probability μ if they have been infected.

2) We can identify all infected nodes, but cannot distinguish susceptible or recovered nodes, because some infected nodes may recover from the disease with a probability such as in the SIR model.

3) Only the nodes infected at time t are observed, while the states of other nodes infected before time t are missing. For example, the observed black nodes in the ring in Fig. 5 are infected at time t.

4) We only observe a part of nodes at time t due to some limitations such as financial and human resources. Note that some observed nodes may be infected before time t.

In the next part, we will introduce some typical solutions to different cases.

6.2.1 Jordan center

This kind of methods selects a Jordan center as the detected source node, which has the maximal Jordan centrality defined in (6). That means Jordan center is a node minimizing the maximum distance with other nodes. Zhu and Ying[104] studied the source detection problem under the popular Susceptible-Infected-Recovered (SIR) model. Given a snapshot of the network, we know all infected nodes but cannot distinguish the susceptible nodes and recovered nodes. The network is assumed to be an undirected graph and each node in the network has three possible states: susceptible (S), infected (I), and recovered (R). Nodes in state S can be infected and change to state I, and nodes in state I can recover and change to state R.

They formalized this problem with maximum likelihood estimation (MLE). To solve it, we need to consider all possible infection sample paths, which is impossible for large-scale networks with unknown initial infection time t_0. To overcome this difficulty, they proposed to find the sample path which most likely leads to the observed snapshot, and viewed the first node associated with that sample path as the information source. They proved that for infinite-trees, the estimator is a node that minimizes the maximum distance to the infected nodes, i.e., the Jordan center. A reverse-infection algorithm was proposed to find such estimator in general graphs. In the algorithm, each infected node broadcasts its identity in the network, and then the node who is the first to collect all identities of infected nodes declares itself as the information source. Ties are broken based on the sum of distances to the infected nodes. They showed it can output a node within a constant distance from the actual source with a high probability, independent of the number of infected nodes and the time the snapshot is taken.

Zhu and Ying[105] further extended this method for source detection under the heterogeneous SIR model with sparse observations. They assumed that a small subset of infected nodes are reported. The heterogeneous SIR model allows different infection probabilities along edges and different recovery probabilities at different nodes. Besides, Luo et al.[106, 107] explored the sample path based approach for source detection under SI and SIS models. They obtained the same conclusion as under the SIR model: the detected source is a Jordan center. However, the Jordan center method is designed for tree-like networks, which are very different from real-world networks.

6.2.2 Message passing methods

The second kind of methods is based on message passing. Lokhov et al.[108] took the infected and uninfected nodes to detect the source node under the SIR model. They introduced an effective inference algorithm based on dynamic message passing (DMP) equations. Let $P_S^i(t)$, $P_I^i(t)$ and $P_R^i(t)$ denote the marginal probabilities that node i is susceptible (S), infected (I), and recovered (R) at time t, respectively. They first used the following DMP equations to estimate the marginal probabilities of a given node.

$$\begin{aligned} &P_S^i(t+1) = P_S^i(0)\prod_k \theta^{k\to i}(t+1)\\ &P_R^i(t+1) = P_R^i(t) + \mu_i P_I^i(t)\\ &P_I^i(t+1) = 1 - P_S^i(t+1) - P_R^i(t+1) \end{aligned} \qquad (26)$$

where $\theta^{k\to i}(t+1)$ is the probability that the infection signal has not been passed from node k to i up to time t+1, and μ_i is the recovery probability of node i. Then they exploited a mean-field-type approach to approximate the likelihood of the observed states as a product of the marginal probabilities. A node maximizing the likelihood is selected to be the source. Importantly, DMP remains efficient in the case where the snapshot sees only a part of the network. Hu et al.[109] extended DMP to the susceptible-infected-recovered-infected (SIRI) model and proposed an algorithm known as the heterogeneous infection spreading source (HISS) estimator to infer the infection source. It is able to incorporate side information (if any) of the observed states of a subset of nodes at different times, and of the prior probability of each infected or recovered node to be the infection source.

As noted by the authors, DMP has two drawbacks. First, the space of initial conditions considered must be explored exhaustively. Second, DMP relies on a further assumption of single-site factorization of the likelihood function, which is not necessarily consistent with the more accurate underlying approximation. Altarelli et al.[110] realized these and conducted Bayesian inference for this problem on a factor graph under the SIR model. They derived belief propagation (BP) equations for the probability distribution of system states conditioned on some observations, which is more accurate than DMP. Besides, BP can be used to identify the origin of an epidemic outbreak in the SIR, SI, and similar models, even with multiple infection seeds and incomplete or heterogeneous information. They further generalized the analysis to more realistic cases in which observations are imperfect[111]. They said it also can give accurate predictions about the future evolution of an outbreak from which only a partial observation (noisy and/or incomplete) of the current state is available.

DMP and BP have been shown to perform better than centrality based methods, such as Jordan center and rumor center in previous sections. However, DMP and BP are too time-consuming for large-scale networks, because they need to run on the whole network which may have large number of nodes.

6.2.3 Diffusion reconstruction

Some methods are based on diffusion reconstruction which recover the states or propagation paths of unknown nodes. For example, Zang et al.[112] presented a multi-source locating method based on a given snapshot of partially and sparsely observed infected nodes in the network. They first introduced a reverse propagation method to detect recovered and unobserved infected nodes in the network, and then used community cluster algorithms to change the multi-source locating problem into a bunch of single source locating problems. At the last step, they identified the nodes with the largest likelihood as the source nodes in the infected clusters.

Gundecha et al.[29] tried to seek the provenance (i.e., sources or originators) of information for a few known recipients by recovering the information propagation paths in social media. The proposed method exploits easily computable node centralities of a large social media network. Feng et al.[113] studied the problem of recovering other unknown recipients and seeking the provenance of information based on a few known recipients. They exploited the property of frequent pattern and node centrality measures to find important nodes.

6.2.4 Others

Karamchandani and Franceschetti[114] extended the rumor centrality for source detection under probabilistic sampling, i.e., each node reports its status with probability p. They computed the centralities of each node in the reported rumor subgraph which is the minimum subgraph that connects all infected nodes. Besides, Brockmann and Helbing[115] proposed a novel concept of effective distance, which can be used to reconstruct the origin of outbreaks. Shi et al.[116] proposed a two-stage method under the SI model that first locates a set of suspected source nodes and then identifies the infection source from the candidate source nodes by the Markov random field method. Zhang et al.[117] studied the diffusion sources locating problem by learning from information diffusion data collected only from a small subset of network nodes. They presented a new regression learning model that can detect anomalous diffusion sources by jointly solving five challenges: unknown number of source nodes, few activated detectors, unknown initial propagation time, uncertain propagation path and uncertain propagation time delay.

6.3 Detection with sensor observation

Sometimes, due to the large quantity of nodes in a network, we have to select some specific nodes as sensors to monitor the information diffusion, such as choosing some users in a social network and many computers in the Internet to stop the spread of wrong information. That means at time t, we can observe the states of these selected sensors. More importantly, sensor nodes also provide the state transition time (i.e., when they are infected) and infection directions (i.e., which adjacent nodes the information comes from). In the next part, we will show how to use those data to detect the source nodes.

6.3.1 Delay distance estimator

Pinto et al.[118] estimated the location of the source from measurements collected by k sparsely placed sensors. Every edge has a deterministic propagation time, which is independent and identically distributed with a Gaussian distribution. The information diffusion follows the continuous SI model, where an infected node will retransmit the information to all of its other neighbors in propagation delays. To minimize the scale of seeking sources, they first determined a unique sub-tree T_a according to the direction from which information arrived at the sensors. Given a sensor node o_1, calculate the observed delay, $\boldsymbol{d}$, between o_1 and the other sensors in T_a. Then they assumed an arbitrary node $s \in T$ as the source node, and got the diffusion time from s to o_k, denoted as $P(s, o_k)$. After that, the deterministic delay, $\boldsymbol{\mu}_s$, is calculated for every sensor node relative to o_1, where $\mu_{s,k} = |P(s, o_k) - P(s, o_1)|$. For a general propagation tree, the optimal estimator is given by

$$\hat{s} = \arg\max_{s \in T_a} \boldsymbol{\mu}_s^{\mathrm{T}} \boldsymbol{\Lambda}^{-1}\left(\boldsymbol{d} - \frac{1}{2}\boldsymbol{\mu}_s\right) \tag{27}$$

where $\boldsymbol{\Lambda}$ is the delay covariance. That means the detected source is a node minimizing the distance between the observed delays and deterministic delays of sensor nodes relative to o_1. It is optimal for arbitrary trees, and achieves the maximum probability of correct localization in $O(n)$. However, it runs in $O(n^3)$ for general graphs with the BFS heuristic method, which is too unsatisfactory to large-scale graphs.

Agaskar and Lu[119] described an alternate representation for the susceptible-infected (SI) infection model based on geodesic distances on a randomly-weighted version of the graph. This representation allows us to exploit fast Monte Carlo (MC) algorithms to compute geodesic distances and to estimate the marginal distributions for each observer, and then compute a pseudo-likelihood function that is maximized to find the source.

Besides, Shen et al.[120] developed a time-reversal backward spreading (TRBS) algorithm to locate the source of a diffusion-like process efficiently. Sensors $o_1, o_2, \cdots, o_k$ receive the information at time $t_{o_1}, t_{o_2}, \cdots, t_{o_k}$. TRBS consists of two steps. First, for arbitrary node i, it computes the reversed arrival time for each sensor and leads to a vector

$$\boldsymbol{T}_i = [t_{o_1} - \hat{t}(i, o_1), t_{o_2} - \hat{t}(i, o_2), \cdots, t_{o_k} - \hat{t}(i, o_k)]^{\mathrm{T}} \quad (28)$$

where $\hat{t}(i, o_k)$ is the shortest time delay from node o_k to i, and follows a certain distribution, such as Gaussian or uniform distributions. Second, calculate the variance of the elements in $\boldsymbol{T}_1, \boldsymbol{T}_2, \cdots, \boldsymbol{T}_n$ and select a node with the minimum variance as the source. It runs in $O(kn \log n)$ and $O(n^2 \log n)$ in the worse case. Fu et al.[121] further extended this method for multiple sources detection with limited observers.

6.3.2 Others

Seo et al.[122] followed the intuition that the source node must be close to the infected sensors but far from the negative monitors, and proposed four metrics (FM): 1) Reachability to all positive monitors. It calculates how many positive monitors are reachable from each node. 2) Distance to positive monitors. They sorted the suspected sources by increasing total distances from positive monitors. 3) Reachability to negative monitors. For each sorted node v, they counted the number of negative monitors that are not reachable from v and preferred larger counts. 4) Distance to negative monitors. It is more natural that negative monitors are far from rumor sources, so nodes with larger distance to negative monitors are preferred.

Offline learning models do not meet the needs of early warning, real-time awareness, and real-time response of malicious information spreading in social networks. Therefore, Wang et al.[123] combined online learning and regression-based detection (OLRD) methods for real-time diffusion source detection with sensors. They proposed a new l_1 non-convex regression model as the learning function, and an online stochastic sub-gradient algorithm (OSS) to optimize the objective.

Sometimes, some sensors may fail to report their states. Louni et al.[124] noted this and addressed the problem of locating the source of a rumor in large social networks where some of these sensor nodes have failed. They estimated the missing information about the sensors by doubly non-negative matrix completion and compressed sensing (DNMC-CS) techniques. It first used the compressed sensing method to recover sporadically missing measurements and the doubly non-negative (DN) completion to recover measurements missing in bursts. Then it detects the rumor source based on the recovered measurements with a maximum likelihood (ML) estimator.

In conclusion, we compare some typical methods for different scenarios in Table 1 and have several findings. First, most current methods are detecting the information sources under epidemic models, such as SI and SIR. While other models such as IC and LT are more widely used for information diffusion analysis in social networks. Under these models, it is promising to solve the source detection problem, like [100]. For example, we can extend current methods to more diffusion models according to the difference among them. Second, current methods have no performance guarantee with respect to generic graphs, except for SISI[100]. Many of them can only provide guarantees in some specific cases. For example, the basic rumor center ensures that the probability of correct detection is bounded uniformly away from 0 under the SI model for regular expander trees and geometric trees[35]. Zhu and Ying[104] proved that Jordan center can output a node within a constant distance from the actual source with a high probability for regular trees. Therefore, we can follow SISI and propose more effective methods with performance guarantees based on reverse sampling algorithms for influence maximization in Section 5.3. Third, detection methods with both partial and sensor observations are running on the whole graph, which is unsatisfactory for large-scale graphs. Therefore, reducing their time complexities is necessary.

6.4 Similar problems

In literature, there are some problems similar to information source detection defined in Problem 2. For example, Lappas et al.[34] defined the problem of k-effectors, which selects a set of k active nodes that can best explain the observed activation states, $\boldsymbol{a}$, in social networks. Formally, k-effectors is a set S of active nodes (effectors), of cardinality at most k such that

$$C(S) = \sum_{v \in V} |\boldsymbol{a}(v) - \alpha(v, S)| \quad (29)$$

is minimized, where $\alpha(v, S)$ is the probability that node v is active at the end of the diffusion process if S is the

Table 1 Comparison of different methods for information source detection, where h is the number of independent observations[96], r is the iteration times, l is the number of edge-disjoint shortest paths among pairs of nodes[101], t_0 is the time how long the information has diffused, and t_r is the maximal recovery time allowed[110]. The performance guarantees are with respect to generic graphs.

Observation	Method		Graph	Diffusion models	# of sources	Performance guarantee	Time complexity
Complete	Rumor center (RC)	Basic RC[35]	tree	SI	single	None	$O(\|V_I\|^2)$
		Local RC[95]	Tree	SI	Single	None	$O(\|V_I\|^2)$
		Joint RC[96]	Tree	SI	Single	None	$h \times O(\|V_I\|^2)$
		Multi RC[97]	Tree	SI	Multiple	None	$O(\|V_I\|^k)$
	Eigenvector center	DI[27]	Generic	SI	Multiple	None	$O(\|V_I\|^3)$
		NETSLEUTH[28]	Generic	SI	Multiple	None	$O(\|V_I\|^3)$
	Sampling methods	MCMC[98]	Generic	IC	Single	None	$r \times O(\|V_I\|^2)$
		SISI[100]	Generic	SI/IC/LT	Multiple	$\frac{2}{(1-\epsilon)^2}\Delta$	$O(\frac{m\Delta\Lambda}{\|E_s\|+\Delta^2})$
	Diffusion kernel	NI[101]	Generic	SI	Multiple	None	$O(\|V_I\|l(m+n\log(n)))$
		SR-DKL	Generic	Heat	Multiple	None	$r \times O(n^2)$
Partial	Jordan center[104]		Tree	SIR	Single	None	$O(n^3)$
	Message passing methods	DMP[108]	Generic	SIR	Single	None	$O(t_0dn2)$
		BP[110]	Generic	SIR	Multiple	None	$O(rmt_0t_r^2)$
	Diffusion reconstruction[112]		Generic	SIR	Multiple	None	$O(n^3)$
Complete	Delay distance estimator	Gaussian[118]	Tree	SI	Single	None	$O(n^3)$
		MC[119]	Generic	SI	Single	None	$O(\frac{n\log n}{\epsilon})$
		TRBS[120]	Generic	SI	Single	None	$O(n^2\log n)$
	Others	FM[122]	Generic	SI	Single	None	$O(n^3)$
		OLRD[123]	Generic	SI	Multiple	None	$O(rn^2)$

source set. They proved that the k-effectors (0) problem is NP-complete under the IC model and it is NP-hard to approximate for general graphs. For tree graphs, the problem can be solved optimally in polynomial time using a dynamic programming algorithm. From (29), we find k-effectors is a generalization of influence maximization, which can be considered as selecting a node set S with $|S| \leq K$ such that $\sum_{v\in V}\alpha(v,S)$ is maximized. What's more, k-effectors is a relaxation of source detection defined in Problem 2, which tries to infer a node set S with $|S| \leq K$ such that $\sum_{v\in V}|\boldsymbol{a}(v) - \alpha(v,S)| = 0$. This also reflects the relationship between source detection and influence maximization. Bulteau et al.[125] provided a more thorough computational complexity analysis of k-effectors. They exploit a parameterization measuring the "degree of randomness" which might be proven useful for analyzing other probabilistic network diffusion problems as well. Besides, Nguyen et al.[126] studied the k-suspector problem which aims to find the top k most suspected sources of wrong information such that the number of original attackers included is maximized, and claimed NP-hardness of the problem under the IC model.

7 Conclusions and future research topics

To conclude, we have reviewed recent advances on information diffusion analysis in social networks and its applications in this paper. Specifically, we first introduced three typical information diffusion models, namely independent cascade (IC) model, linear threshold (LT) model and epidemic models, which can be used to describe how the information diffuses in a network. Then, we showed three practical problems: authority and influence evaluation, influence maximization, and information source detection. Authority and influence evaluation in social networks is important for influential spreaders identification and expert finding, while influence maximization contributes to viral marketing and sensor placement. Information source detection has a wide range of applications such as epidemic outbreak prevention and rumor source tracing in social networks. Although many efforts have been devoted to these problems, there are still some rooms for improvement. Here we will list several possible directions for further study.

First, current information diffusion models have perfect theoretical properties for further analysis, but simplify real-world scenarios which are actually very complicated. Users can access the information from external sources such as TV, newspaper and other websites, not only from its neighbors in a social network. Besides, there may be multiple types of information spreading in the network at the same time, such as information of competitive products. Therefore, it is promising to model multiple types of information diffusion in heterogeneous social networks with external influence. For example, Myers et al.[127] presented a model in which information can reach a node via the links of the social network or through the influence of external sources. Besides, Zhan et al.[128] studied the influence maximization problem in multiple partially aligned heterogeneous in online social networks.

Second, large scalability is one of the biggest challenges to apply influence maximization and information source detection in real-world applications, especially for large-scale networks. Solutions to influence maximization have achieved a great improvement after reverse sampling algorithms are proposed by Borgs et al.[70], and thus we can bring in the experience to accelerate solutions to information source detection, like Nguyen et al.[100] Besides, implementing these solutions in distributed programming is another practical direction.

Third, most current solutions are applicable for static networks, and they neglect that networks are dynamic and evolving. For example, a user may unfollow some of his friends in some time and his personal interests may change on different topics. That's to say, the tie strengths among different users are varying over time. We should take this fact into account so as to analyze the information diffusion in social networks better.

Forth, deep learning has been applied to many tasks of social network analysis recently, such as network embedding[129, 130] and link prediction[131]. The real process of information diffusion in social networks is complicated and sometimes unobserved. Can we design deep learning methods for analyzing the information diffusion? For example, when we input the network structure and a user's attributes such as the age, gender, posts, into a deep learning based model, we can output this user's influence. Bourigault et al.[132] proposed a representation learning approach for information source detection in social networks. It relies neither on a known diffusion graph nor on a hypothetical diffusion law, but directly infers the source from diffusion records.

Finally, it is attractive to incorporate the information diffusion analysis with other practical problems, such as behavior prediction for social users[8, 133, 134]. For example, social users are usually affected by multiple companies at the same time, and not only the user interests but also these social influences will contribute to the user consumption behaviors. Ma et al.[135] proposed a general approach to figure out the targeted users for social marketing, taking both user interests and multiple social influences into consideration. Valuable users should have the best balanced influence entropy (being "Hesitant") and utility scores (being "Interested"). Wu et al.[133] took the underlying social theories to explain and model the evolution of users' two kinds of behaviors: users' preferences (reflected in user-item interaction behavior) and the social network structure (reflected in user-user interaction behavior). Xu et al.[8] tried to reveal how the social propagation affects the prediction of cab drivers' future behaviors.

Acknowledgements

This research was supported by National Natural Science Foundation of China (Nos. 61703386, U1605251 and 91546103), the Anhui Provincial Natural Science Foundation (No. 1708085QF140), the Fundamental Research Funds for the Central Universities (No. WK2150110006), and the Youth Innovation Promotion Association of Chinese Academy of Sciences (No. 2014299).

References

[1] Zephoria. The top 20 valuable Facebook statistics-updated April 2018. [Online], Available: https://zephoria.com/top-15-valuable-facebook-statistics/,October1,2017.

[2] H. Kwak, C. Lee, H. Park, S. Moon. What is Twitter, a social network or a news media? In *Proceedings of the 19th International Conference on World Wide Web*, New York, USA, pp. 591–600, 2010.

[3] H. P. Zhang, R. Q. Zhang, Y. P. Zhao, B. J. Ma. Big data modeling and analysis of microblog ecosystem. *International Journal of Automation and Computing*, vol. 11, no. 2, pp. 119–127, 2014. DOI: 10.1007/s11633-014-0774-9.

[4] D. Kempe, J. Kleinberg, É. Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, Washington DC, USA, pp. 137–146, 2003. DOI: 10.1145/956750.956769.

[5] J. Leskovec, L. A. Adamic, B. A. Huberman. The dynamics of viral marketing. *ACM Transactions on the Web*, vol. 1, no. 1, pp. 5, 2007. DOI: 10.1145/1232722.1232727.

[6] M. Richardson, P. Domingos. Mining knowledge-sharing sites for viral marketing. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, Edmonton, Canada, pp. 61–70, 2002. DOI: 10.1145/775047.775057.

[7] C. Ma, C. Zhu, Y. J. Fu, H. S. Zhu, G. Q. Liu, E. H. Chen. Social user profiling: A social-aware topic modeling perspective. In *Proceedings of the 22nd International Conference on Database Systems for Advanced Applications*, Springer, Suzhou, China, pp. 610–622, 2017. DOI: 10.1007/978-3-319-55699-4_38.

[8] T. Xu, H. S. Zhu, X. Y. Zhao, Q. Liu, H. Zhong, E. H. Chen, H. Xiong. Taxi driving behavior analysis in latent vehicle-to-vehicle networks: A social influence perspective. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, San Francisco, USA, pp. 1285–1294, 2016. DOI: 10.1145/2939672.2939799.

[9] X. Y. Zhao, T. Xu, Q. Liu, H. Guo. Exploring the choice

under conflict for social event participation. In *Proceedings of the 21st International Conference on Database Systems for Advanced Applications*, Springer, Dallas, USA, pp. 396–411, 2016. DOI: 10.1007/978-3-319-32025-0_25.

[10] L. Backstrom, J. Leskovec. Supervised random walks: Predicting and recommending links in social networks. In *Proceedings of the 4th ACM International Conference on Web Search and Data Mining*, ACM, Hong Kong, China, pp. 635–644, 2011. DOI: 10.1145/1935826.1935914.

[11] D. Liben-Nowell, J. Kleinberg. The link-prediction problem for social networks. *Journal of the American Society for Information Science and Technology*, vol. 58, no. 7, pp. 1019–1031, 2007. DOI: 10.1002/asi.v58:7.

[12] T. Xu, H. S. Zhu, E. H. Chen, B. X. Huai, H. Xiong, J. L. Tian. Learning to annotate via social interaction analytics. *Knowledge and Information Systems*, vol. 41, no. 2, pp. 251–276, 2014. DOI: 10.1007/s10115-013-0717-8.

[13] S. Fortunato. Community detection in graphs. *Physics Reports*, vol. 486, no. 3, pp. 75–174, 2010. DOI: 10.1016/j.physrep.2009.11.002.

[14] S. Fortunato, M. Barthlemy. Resolution limit in community detection. *Proceedings of the National Academy of Sciences of the United States of America*, vol. 104, no. 1, pp. 36–41, 2007. DOI: 10.1073/pnas.0605965104.

[15] M. Girvan, M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, no. 12, pp. 7821–7826, 2002.

[16] J. Goldenberg, B. Libai, E. Muller. Talk of the network: A complex systems look at the underlying process of word-of-mouth. *Marketing Letters*, vol. 12, no. 3, pp. 211–223, 2001. DOI: 10.1023/A:1011122126881.

[17] M. Granovetter. Threshold models of collective behavior. *American Journal of Sociology*, vol. 83, no. 6, pp. 1420–1443, 1978. DOI: 10.1086/226707.

[18] W. O. Kermack, A. G. McKendrick. Contributions to the mathematical theory of epidemics – II. The problem of endemicity. *Bulletin of Mathematical Biology*, vol. 53, no. 1–2, pp. 57–87, 1991. DOI: 10.1007/BF02464424.

[19] M. Cataldi, L. Di Caro, C. Schifanella. Emerging topic detection on Twitter based on temporal and social terms evaluation. In *Proceedings of the 10th International Workshop on Multimedia Data Mining*, ACM, Washington DC, USA, 2010. DOI: 10.1145/1814245.1814249.

[20] M. Kitsak, L. K. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H. E. Stanley, H. A. Makse. Identification of influential spreaders in complex networks. *Nature Physics*, vol. 6, no. 11, pp. 888–893, 2010. DOI: 10.1038/nphys1746.

[21] J. Zhang, J. Tang, J. Z. Li. Expert finding in a social network. In *Proceedings of the 12th International Conference on Database Systems for Advanced Applications*, Springer, Bangkok, Thailand, pp. 1066–1069, 2007. DOI: 10.1007/978-3-540-71703-4_106.

[22] H. S. Zhu, E. H. Chen, H. Xiong, H. H. Cao, J. L. Tian. Ranking user authority with relevant knowledge categories for expert finding. *World Wide Web*, vol. 17, no. 5, pp. 1081–1107, 2014. DOI: 10.1007/s11280-013-0217-5.

[23] H. S. Zhu, E. H. Chen, H. H. Cao. Finding experts in tag based knowledge sharing communities. In *Proceedings of the 5th International Conference on Knowledge Science, Engineering and Management*, Springer, Irvine, USA, pp. 183–195, 2011. DOI: 10.1007/978-3-642-25975-3_17.

[24] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance. Cost-effective outbreak detection in networks. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, San Jose, USA, pp. 420–429, 2007. DOI: 10.1145/1281192.1281239.

[25] L. Zou, Z. D. Wang, D. H. Zhou. Event-based control and filtering of networked systems: A survey. *International Journal of Automation and Computing*, vol. 14, no. 3, pp. 239–253, 2017. DOI: 10.1007/s11633-017-1077-8.

[26] B. Chang, F. D. Zhu, E. H. Chen, Q. Liu. Information source detection via maximum a posteriori estimation. In *Proceedings of IEEE International Conference on Data Mining*, IEEE, Atlantic City, USA, pp. 21–30, 2015. DOI: 10.1109/ICDM.2015.116.

[27] V. Fioriti, M. Chinnici. Predicting the sources of an outbreak with a spectral technique. *Applied Mathematical Sciences*, vol. 8, pp. 6775–6782, 2014. DOI: 10.12988/ams.2014.49693.

[28] B. A. Prakash, J. Vreeken, C. Faloutsos. Spotting culprits in epidemics: How many and which ones? In *Proceedings of the 12th International Conference on Data Mining*, IEEE, Brussels, Belgium, pp. 11–20, 2012. DOI: 10.1109/ICDM.2012.136.

[29] P. Gundecha, Z. Feng, H. Liu. Seeking provenance of information using social media. In *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*, ACM, San Francisco, USA, pp. 1691–1696, 2013. DOI: 10.1145/2505515.2505633.

[30] D. Shah, T. Zaman. Rumors in a network: Who's the culprit? *IEEE Transactions on Information Theory*, vol. 57, no. 8,pp. 5163–5181,2011.DOI:10.1109/TIT.2011.2158885.

[31] A. Goyal, F. Bonchi, L. V. S. Lakshmanan. Learning influence probabilities in social networks. In *Proceedings of the 3rd ACM International Conference on Web Search and Web Data Mining*, ACM, New York, USA, pp. 241–250, 2010. DOI: 10.1145/1718487.1718518.

[32] B. Ryan, N. C. Gross. The diffusion of hybrid seed corn in two Iowa communities. *Rural Sociology*, vol. 8, no. 1, pp. 15–24, 1943.

[33] H. Y. Zhang, S. Mishra, M. T. Thai. Recent advances in information diffusion and influence maximization in complex social networks. *Opportunistic Mobile Social Networks*, J. Wu, Y. S. Wang, Eds., USA: CRC Press, 2014.

[34] T. Lappas, E. Terzi, D. Gunopulos, H. Mannila. Finding effectors in social networks. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, Washington DC, USA, pp. 1059–1068, 2010. DOI: 10.1145/1835804.1835937.

[35] D. Shah, T. Zaman. Detecting sources of computer viruses in networks: Theory and experiment. In *Proceedings of the ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, ACM, New York, USA, pp. 203–214, 2010. DOI: 10.1145/1811039.1811063.

[36] R. Zafarani, M. A. Abbasi, H. Liu. *Social Media Mining: An Introduction*, Cambridge, UK: Cambridge University Press, 2014.

[37] Y. Yao, X. R. Luo, F. X. Gao, S. L. Ai. Research of a potential worm propagation model based on pure P2P principle. In *Proceedings of International Conference on Communication Technology*, IEEE, Guilin, China, 2006. DOI: 10.1109/ICCT.2006.342006.

[38] H. W. Hethcote. The mathematics of infectious diseases. *SIAM Review*, vol. 42, no. 4, pp. 599–653, 2000. DOI: 10.1137/S0036144500371907.

[39] K. L. Cooke, P. van den Driessche. Analysis of an SEIRS epidemic model with two delays. *Journal of Mathematical Biology*, vol. 35, no. 2, pp. 240–260, 1996. DOI: 10.1007/s002850050051.

[40] Y. Xiang, X. Fan, W. T. Zhu. Propagation of active worms: A survey. *International Journal of Computer Sys-*

tems Science and Engineering, vol. 24, no. 3, pp. 157–172, 2009.

[41] H. S. Zhu, E. H. Chen, H. H. Cao, J. L. Tian. Context-aware expert finding in tag based knowledge sharing communities. *International Journal of Knowledge and Systems Science*, vol. 3, no. 1, pp. 48–63, 2012. DOI: 10.4018/jkss.2012010104.

[42] H. S. Zhu, H. H. Cao, H. Xiong, E. H. Chen, J. L. Tian. Towards expert finding by leveraging relevant categories in authority ranking. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, ACM, Glasgow, UK, pp. 2221–2224, 2011. DOI: 10.1145/2063576.2063931.

[43] B. A. Prakash, C. Faloutsos. Understanding and managing cascades on large graphs. *Proceedings of the VLDB Endowment*, vol. 5, no. 12, pp. 2024–2025, 2012. DOI: 10.14778/2367502.2367567.

[44] B. Xiang, Q. Liu, E. H. Chen, H. Xiong, Y. Zheng, Y. Yang. PageRank with priors: An influence propagation perspective. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, Beijing, China, pp. 2740–2746, 2013.

[45] S. Y. Lin, W. X. Hong, D. D. Wang, T. Li. A survey on expert finding techniques. *Journal of Intelligent Information Systems*, vol. 49, no. 2, pp. 255–279, 2017. DOI: 10.1007/s10844-016-0440-5.

[46] A. Bavelas. Communication patterns in task-oriented groups. *The Journal of the Acoustical Society of America*, vol. 22, no. 6, pp. 725–730, 1950. DOI: 10.1121/1.1906679.

[47] L. C. Freeman. A set of measures of centrality based on betweenness. *Sociometry*, vol. 40, no. 1, pp. 35–41, 1977. DOI: 10.2307/3033543.

[48] C. Jordan. Sur les assemblages de lignes. *Journal fr die reine und angewandte Mathematik*, vol. 1869, no. 70, pp. 185–190, 1869. DOI: 10.1515/crll.1869.70.185.

[49] L. Page, S. Brin, R. Motwani, T. Winograd. The PageRank citation ranking: Bringing order to the web, Technical Report SIDL-WP-1999-0120, Stanford University, USA, 1999.

[50] P. Berkhin. A survey on PageRank computing. *Internet Mathematics*, vol. 2, no. 1, pp. 73–120, 2005. DOI: 10.1080/15427951.2005.10129098.

[51] T. H. Haveliwala. Topic-sensitive PageRank. In *Proceedings of the 11th International Conference on World Wide Web*, ACM, New York, USA, pp. 517–526, 2002.

[52] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, vol. 46, no. 5, pp. 604–632, 1999. DOI: 10.1145/324133.324140.

[53] J. S. Weng, E. P. Lim, J. Jiang, Q. He. TwitterRank: Finding topic-sensitive influential twitterers. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining*, ACM, New York, USA, pp. 261–270, 2010. DOI: 10.1145/1718487.1718520.

[54] Q. Liu, B. Xiang, E. H. Chen, Y. Ge, H. Xiong, T. F. Bao, Y. Zheng. Influential seed items recommendation. In *Proceedings of the 6th ACM Conference on Recommender Systems*, ACM, Dublin, Ireland, pp. 245–248, 2012. DOI: 10.1145/2365952.2366005.

[55] W. Chen, C. Wang, Y. J. 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, Washington DC, USA, pp. 1029–1038, 2010. DOI: 10.1145/1835804.1835934.

[56] W. Chen, Y. F. Yuan, L. Zhang. Scalable influence maximization in social networks under the linear threshold model. In *Proceedings of the 10th International Conference on Data Mining*, IEEE, Sydney, Australia, pp. 88–97, 2010. DOI: 10.1109/ICDM.2010.118.

[57] C. C. Aggarwal, A. Khan, X. F. Yan. On flow authority discovery in social networks. In *Proceedings of SIAM International Conference on Data Mining*, SIAM, Mesa, USA,pp. 522–533,2011.DOI:10.1137/1.9781611972818.45.

[58] Y. Yang, E. H. Chen, Q. Liu, B. Xiang, T. Xu, S. A. Shad. On approximation of real-world influence spread. In *Proceedings of European Conference on Machine Learning and Knowledge Discovery in Databases*, Springer, Bristol, UK, pp. 548–564, 2012. DOI: 10.1007/978-3-642-33486-3_35.

[59] J. Yang, J. Leskovec. Modeling information diffusion in implicit networks. In *Proceedings of the10th International Conference on Data Mining*, IEEE, Sydney, Australia, pp. 599–608, 2010. DOI: 10.1109/ICDM.2010.22.

[60] Q. Liu, B. Xiang, N. J. Yuan, E. H. Chen, H. Xiong, Y. Zheng, Y. Yang. An influence propagation view of PageRank. *ACM Transactions on Knowledge Discovery from Data*, vol. 11, no. 3, pp. 30, 2017. DOI: 10.1145/3046941.

[61] J. Tang, J. M. Sun, C. Wang, Z. Yang. Social influence analysis in large-scale networks. In *Proceedings of the 15th ACM International Conference on Knowledge Discovery and Data Mining*, ACM, Paris, France, pp. 807–816, 2009. DOI: 10.1145/1557019.1557108.

[62] Q. Liu, B. Xiang, L. Zhang, E. H. Chen, C. Tan, J. Chen. Linear computation for independent social influence. In *Proceedings of the 13th International Conference on Data Mining*, IEEE, Dallas, USA, pp. 468–477, 2013. DOI: 10.1109/ICDM.2013.48.

[63] Q. Liu, B. Xiang, E. H. Chen, H. Xiong, F. S. Tang, J. X. Yu. Influence maximization over large-scale social networks: A bounded linear approach. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, ACM, Shanghai, China, pp. 171–180, 2014. DOI: 10.1145/2661829.2662009.

[64] G. L. Nemhauser, L. A. Wolsey, M. L. Fisher. An analysis of approximations for maximizing submodular set functions-I. *Mathematical Programming*, vol. 14, pp. 265–294, 1978.

[65] A. Goyal, W. Lu, L. V. S. Lakshmanan. CELF++: Optimizing the greedy algorithm for influence maximization in social networks. In *Proceedings of the 20th International Conference Companion on World Wide Web*, ACM, Hyderabad, India, pp. 47–48, 2011. DOI: 10.1145/1963192.1963217.

[66] K. Jung, W. Heo, W. Chen. IRIE: Scalable and robust influence maximization in social networks. In *Proceedings of the 12th International Conference on Data Mining*, IEEE, Brussels, Belgium, pp. 918–923, 2012. DOI: 10.1109/ICDM.2012.79.

[67] M. Kimura, K. Saito. Tractable models for information diffusion in social networks. In *Proceedings of the 10th European Conference on Principles and Practice of Knowledge Discovery in Databases*, Springer, Berlin, Germany, pp. 259–271, 2006. DOI: 10.1007/11871637_27.

[68] W. Chen, Y. J. Wang, S. Y. Yang. Efficient influence maximization in social networks. In *Proceedings of the 15th ACM International Conference on Knowledge Discovery and Data Mining*, ACM, Paris, France, pp. 199–208, 2009. DOI: 10.1145/1557019.1557047.

[69] A. Goyal, W. Lu, L. V. S. Lakshmanan. SIMPATH: An efficient algorithm for influence maximization under the linear threshold model. In *Proceedings of the 11th International Conference on Data Mining*, IEEE, Vancouver, Canada, pp. 211–220, 2011. DOI: 10.1109/ICDM.2011.132.

[70] C. Borgs, M. Brautbar, J. Chayes, B. Lucier. Maximizing social influence in nearly optimal time. In *Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms*, Illinois, USA, pp. 946–957, 2014. DOI: 10.1137/1.9781611973402.70.

[71] Y. Z. Tang, X. K. Xiao, Y. C. Shi. Influence maximization: Near-optimal time complexity meets practical efficiency. In *Proceedings of ACM SIGMOD International Conference on Management of Data*, ACM, Snowbird, USA, pp. 75–86, 2014. DOI: 10.1145/2588555.2593670.

[72] V. V. Vazirani. *Approximation Algorithms*, Berlin Heidelberg, Germany: Springer, 2013. DOI: 10.1007/978-3-662-04565-7.

[73] Y. Z. Tang, Y. C. Shi, X. K. Xiao. Influence maximization in near-linear time: A martingale approach. In *Proceedings of ACM SIGMOD International Conference on Management of Data*, ACM, Melbourne, Australia, pp. 1539–1554, 2015. DOI: 10.1145/2723372.2723734.

[74] E. Cohen, D. Delling, T. Pajor, R. F. Werneck. Sketch-based influence maximization and computation: Scaling up with guarantees. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, ACM, Shanghai, China, pp. 629–638, 2014. DOI: 10.1145/2661829.2662077.

[75] 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 International Conference on Management of Data*, ACM, San Francisco, USA, pp. 695–710, 2016. DOI: 10.1145/2882903.2915207.

[76] K. K. Huang, S. B. Wang, G. Bevilacqua, X. K. Xiao, L. V. S. Lakshmanan. Revisiting the stop-and-stare algorithms for influence maximization. *Proceedings of the VLDB Endowment*, vol. 10, no. 9, pp. 913–924, 2017. DOI: 10.14778/3099622.3099623.

[77] Y. Wang, G. Cong, G. J. Song, K. Q. Xie. Community-based greedy algorithm for mining top-K influential nodes in mobile social networks. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, Washington DC, USA, pp. 1039–1048, 2010. DOI: 10.1145/1835804.1835935.

[78] Z. F. Wang, H. Wang, Q. Liu, E. H. Chen. Influential nodes selection: A data reconstruction perspective. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, ACM, Gold Coast, Australia, pp. 879–882, 2014. DOI: 10.1145/2600428.2609464.

[79] Q. Y. Jiang, G. J. Song, G. Cong, Y. Wang, W. J. Si, K. Q. Xie. Simulated annealing based influence maximization in social networks. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence*, AAAI, San Francisco, USA, pp. 127–132, 2011.

[80] M. G. Rodriguez, B. Schölkopf. Influence maximization in continuous time diffusion networks. https://arxiv.org/abs/1205.1682, 2012.

[81] Y. K. Wang, J. H. Zhu, Q. Ming. Incremental influence maximization for dynamic social networks. In *Proceedings of the 3rd International Conference of Pioneering Computer Scientists, Engineers and Educators*, Springer, Changsha, China, pp. 13–27, 2017. DOI: 10.1007/978-981-10-6388-6_2.

[82] E. Güney. On the optimal solution of budgeted influence maximization problem in social networks. *Operational Research*, to be published. DOI: 10.1007/s12351-017-0305-x.

[83] H. Nguyen, R. Zheng. On budgeted influence maximization in social networks. *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 6, pp. 1084–1094, 2013. DOI: 10.1109/JSAC.2013.130610.

[84] Y. Yang, X. B. Mao, J. Pei, X. F. He. Continuous influence maximization: What discounts should we offer to social network users? In *Proceedings of International Conference on Management of Data*, ACM, San Francisco, USA, pp. 727–741, 2016. DOI: 10.1145/2882903.2882961.

[85] C. Aslay, F. Bonchi, L. V. S. Lakshmanan, W. Lu. Revenue maximization in incentivized social advertising. *Proceedings of the VLDB Endowment*, vol. 10, no. 11, pp. 1238–1249, 2017. DOI: 10.14778/3137628.3137635.

[86] F. H. Li, C. T. Li, M. K. Shan. Labeled influence maximization in social networks for target marketing. In *Proceedings of the 3rd International Conference on Privacy, Security, Risk and Trust and the IEEE 3rd International Conference on Social Computing*, IEEE, Boston, USA, pp. 560–563, 2011. DOI: 10.1109/PASSAT/SocialCom.2011.152.

[87] F. S. Tang, Q. Liu, H. S. Zhu, E. H. Chen, F. D. Zhu. Diversified social influence maximization. In *Proceedings of IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, IEEE, Beijing, China, pp. 455–459, 2014. DOI: 10.1109/ASONAM.2014.6921625.

[88] Q. Liu, Z. Dong, C. R. Liu, X. Xie, E. H. Chen, H. Xiong. Social marketing meets targeted customers: A typical user selection and coverage perspective. In *Proceedings of IEEE International Conference on Data Mining*, IEEE, Shenzhen, China, pp. 350–359, 2014. DOI: 10.1109/ICDM.2014.93.

[89] Z. F. Wang, E. H. Chen, Q. Liu, Y. Yang, Y. Ge, B. Chang. Maximizing the coverage of information propagation in social networks. In *Proceedings of the 24th International Conference on Artificial Intelligence*, AAAI, Buenos Aires, Argentina, pp. 2104–2110, 2015.

[90] Z. F. Wang, Y. Yang, J. Pei, L. Y. Chu, E. H. Chen. Activity maximization by effective information diffusion in social networks. *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 11, pp. 2374–2387, 2017. DOI: 10.1109/TKDE.2017.2740284.

[91] X. R. He, G. J. Song, W. Chen, Q. Y. Jiang. Influence blocking maximization in social networks under the competitive linear threshold model. In *Proceedings of SIAM International Conference on Data Mining*, SIAM, Anaheim, USA, pp. 463–474, 2012. DOI: 10.1137/1.9781611972825.40.

[92] J. H. Zhao, Q. P. Liu, L. Wang, X. F. Wang. Competitiveness maximization on complex networks. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, to be published. DOI: 10.1109/TSMC.2016.2636240.

[93] G. W. Ma, Q. Liu, E. H. Chen, B. Xiang. Individual influence maximization via link recommendation. In *Proceedings of the 16th International Conference on Web-age Information Management*, Springer, Qingdao, China, pp. 42–56, 2015. DOI: 10.1007/978-3-319-21042-1_4.

[94] J. J. Jiang, S. Wen, S. Yu, Y. Xiang, W. L. Zhou. Identifying propagation sources in networks: State-of-the-art and comparative studies. *IEEE Communications Surveys & Tutorials*, vol. 19, no. 1, pp. 465–481, 2017. DOI: 10.1109/COMST.2016.2615098.

[95] W. X. Dong, W. Y. Zhang, C. W. Tan. Rooting out the rumor culprit from suspects. In *Proceedings of IEEE International Symposium on Information Theory*, IEEE, Istanbul, Turkey, pp. 2671–2675, 2013. DOI: 10.1109/ISIT.2013.6620711.

[96] Z. X. Wang, W. X. Dong, W. Y. Zhang, C. W. Tan. Rooting our rumor sources in online social networks: The value of diversity from multiple observations. *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 4, pp. 663–677, 2015. DOI: 10.1109/JSTSP.2015.2389191.

[97] W. Q. Luo, W. P. Tay, M. Leng. Identifying infection sources and regions in large networks. *IEEE Transactions on Signal Processing*, vol. 61, no. 11, pp. 2850–2865, 2013. DOI: 10.1109/TSP.2013.2256902.

[98] X. M. Zhai, W. L. Wu, W. Xu. Cascade source inference in networks: A Markov chain monte Carlo approach. *Computational Social Networks*, vol. 2, no. 1, pp. 17, 2015. DOI: 10.1186/s40649-015-0017-4.

[99] L. Zhang, T. Y. Jin, T. Xu, B. Chang, Z. F. Wang, E. H. Chen. A Markov chain monte Carlo approach for source detection in networks. In *Proceedings of the 6th National Conference on Social Media Processing*, Springer, Beijing, China, pp. 77–88, 2017. DOI: 10.1007/978-981-10-6805-8_7.

[100] H. T. Nguyen, P. Ghosh, M. L. Mayo, T. N. Dinh. Multiple infection sources identification with provable guarantees. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, ACM, Indianapolis, USA, pp. 1663–1672, 2016. DOI: 10.1145/2983323.2983817.

[101] S. Feizi, K. Duffy, M. Kellis, M. Médard. Network infusion to infer information sources in networks, Technical Report MIT-CSAIL-TR-2014-028, Computer Science and Artificial Intelligence Laboratory, USA, 2014.

[102] R. Pena, X. Bresson, P. Vandergheynst. Source localization on graphs via L_1 recovery and spectral graph theory. In *Proceedings of the 12th Image, Video, and Multidimensional Signal Processing Workshop*, IEEE, Bordeaux, France, 2016. DOI: 10.1109/IVMSPW.2016.7528230.

[103] K. Zhu, L. Ying. Information source detection in networks: Possibility and impossibility results. In *Proceedings of the 35th Annual IEEE International Conference on Computer Communications*, IEEE, San Francisco, USA, 2016. DOI: 10.1109/INFOCOM.2016.7524363.

[104] K. Zhu, L. Ying. Information source detection in the SIR model: A sample-path-based approach. *IEEE/ACM Transactions on Networking*, vol. 24, no. 1, pp. 408–421, 2016. DOI: 10.1109/TNET.2014.2364972.

[105] K. Zhu, L. Ying. A robust information source estimator with sparse observations. *Computational Social Networks*, vol. 1, no. 1, pp. 3, 2014. DOI: 10.1186/s40649-014-0003-2.

[106] W. Q. Luo, W. P. Tay. Finding an infection source under the sis model. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, IEEE, Vancouver, Canada, pp. 2930–2934, 2013. DOI: 10.1109/ICASSP.2013.6638194.

[107] W. Q. Luo, W. P. Tay, M. Leng. How to identify an infection source with limited observations. *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 4, pp. 586–597, 2014. DOI: 10.1109/JSTSP.2014.2315533.

[108] A. Y. Lokhov, M. Mézard, H. Ohta, L. Zdeborová. Inferring the origin of an epidemic with a dynamic message-passing algorithm. *Physical Review E*, vol 90, no. 1, Article number 012801, 2014. DOI: 10.1103/PhysRevE.90.012801.

[109] W. H. Hu, W. P. Tay, A. Harilal, G. X. Xiao. Network infection source identification under the SIRI model. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, IEEE, Brisbane, Australia, pp. 1712–1716, 2015. DOI: 10.1109/ICASSP.2015.7178263.

[110] F. Altarelli, A. Braunstein, L. Dall′Asta, A. Lage-Castellanos, R. Zecchina. Bayesian inference of epidemics on networks via belief propagation. *Physical Review Letters*, vol. 112, no. 11, Article number 118701, 2014. DOI: 10.1103/PhysRevLett.112.118701. DOI: 10.1103/PhysRevLett.112.118701.

[111] F. Altarelli, A. Braunstein, L. Dall′Asta, A. Ingrosso, R. Zecchina. The patient-zero problem with noisy observations. *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2014, no. 10, Article number 10016, 2014. DOI: 10.1088/1742-5468/2014/10/P10016. DOI: 10.1088/1742-5468/2014/10/P10016.

[112] W. Y. Zang, P. Zhang, C. Zhou, L. Guo. Discovering multiple diffusion source nodes in social networks. *Procedia Computer Science*, vol. 29, pp. 443–452, 2014. DOI: 10.1016/j.procs.2014.05.040.

[113] Z. Feng, P. Gundecha, H. Liu. Recovering information recipients in social media via provenance. In *Proceedings of IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, IEEE, Niagara Falls, Canada, pp. 706–711, 2013. DOI: 10.1109/ASONAM.2013.6785780.

[114] N. Karamchandani, M. Franceschetti. Rumor source detection under probabilistic sampling. In *Proceedings of IEEE International Symposium on Information Theory*, IEEE, Istanbul, Turkey, pp. 2184–2188, 2013. DOI: 10.1109/ISIT.2013.6620613.

[115] D. Brockmann, D. Helbing. The hidden geometry of complex, network-driven contagion phenomena. *Science*, vol. 342, no. 6164, pp. 1337–1342, 2013. DOI: 10.1126/science.1245200.

[116] C. Y. Shi, Q. Zhang, T. G. Chu. Source identification of network diffusion processes with partial observations. In *Proceedings of the 36th Chinese Control Conference*, IEEE, Dalian, China, pp. 11296–11300, 2017. DOI: 10.23919/ChiCC.2017.8029159.

[117] P. Zhang, J. He, G. D. Long, G. Y. Huang, C. Q. Zhang. Towards anomalous diffusion sources detection in a large network. *ACM Transactions on Internet Technology*, vol. 16, no. 1, pp. 24, 2016. DOI: 10.1145/2806889.

[118] P. C. Pinto, P. Thiran, M. Vetterli. Locating the source of diffusion in large-scale networks. *Physical Review Letters*, vol. 109, no. 6, Article number 068702, 2012. DOI: 10.1103/PhysRevLett.109.068702. DOI: 10.1103/PhysRevLett.109.068702.

[119] A. Agaskar, Y. M. Lu. A fast Monte Carlo algorithm for source localization on graphs. In *Proceedings of SPIE 8858, Wavelets and Sparsity XV*, SPIE, San Diego, USA, 2013. DOI: 10.1117/12.2023039.

[120] Z. S. Shen, S. N. Cao, W. X. Wang, Z. R. Di, H. E. Stanley. Locating the source of diffusion in complex networks by time-reversal backward spreading. *Physical Review E*, vol. 93, no. 3, Article number 032301, 2016. DOI: 10.1103/PhysRevE.93.032301. DOI: 10.1103/PhysRevE.93.032301.

[121] L. Fu, Z. S. Shen, W. X. Wang, Y. Fan, Z. R. Di. Multi-source localization on complex networks with limited observers. *Europhysics Letters*, vol. 113, no. 1, Article number 18006, 2016. DOI: 10.1209/0295-5075/113/18006.

[122] E. Seo, P. Mohapatra, T. Abdelzaher. Identifying rumors and their sources in social networks. In *Proceedings of Volume 8389, Ground/Air Multisensor Interoperability, Integration, and Networking for Persistent ISR III*, SPIE, Baltimore, USA, 2012. DOI: 10.1117/12.919823.

[123] H. S. Wang, P. Zhang, L. Chen, H. Liu, C. Q. Zhang. Online diffusion source detection in social networks. In *Proceedings of International Joint Conference on Neural Networks*, IEEE, Killarney, Ireland, pp. 1–8, 2015. DOI: 10.1109/IJCNN.2015.7280455.

[124] A. Louni, S. Anand, K. P. Subbalakshmi. Identification of source of rumors in social networks with incomplete information. https://arxiv.org/abs/1509.00557, 2015.

[125] L. Bulteau, S. Fafianie, V. Froese, R. Niedermeier, N. Talmon. The complexity of finding effectors. *Theory of Computing Systems*, vol. 60, no. 2, pp. 253–279, 2017.

DOI: 10.1007/s00224-016-9670-8.

[126] D. T Nguyen, N. P. Nguyen, M. T. Thai. Sources of misinformation in online social networks: Who to suspect? In *Proceedings of IEEE Military Communications Conference*, IEEE, Orlando, USA, 2012. DOI: 10.1109/MILCOM.2012.6415780.

[127] S. A. Myers, C. G. Zhu, J. Leskovec. Information diffusion and external influence in networks. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, Beijing, China, pp. 33–41, 2012. DOI: 10.1145/2339530.2339540.

[128] Q. Y. Zhan, J. W. Zhang, S. Z. Wang, P. S. Yu, J. Y. Xie. Influence maximization across partially aligned heterogenous social networks. In *Proceedings of the 19th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining*, Springer, Ho Chi Minh City, Vietnam, pp. 58–69, 2015. DOI: 10.1007/978-3-319-18038-0_5.

[129] D. F. Du, H. Wang, T. Xu, Y. N. Lu, Q. Liu, E. H. Chen. Solving link-oriented tasks in signed network via an embedding approach. In *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*, IEEE, Banff, Canada, pp. 75–80, 2017. DOI: 10.1109/SMC.2017.8122581.

[130] D. X. Wang, P. Cui, W. W. Zhu. Structural deep network embedding. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, San Francisco, USA, pp. 1225–1234, 2016. DOI: 10.1145/2939672.2939753.

[131] F. Liu, B. Q. Liu, C. J. Sun, M. Liu, X. L. Wang. Deep learning approaches for link prediction in social network services. In *Proceedings of the 20th International Conference on Neural Information Processing*, Springer, Daegu, Korea, pp. 425–432, 2013. DOI: 10.1007/978-3-642-42042-9_53.

[132] S. Bourigault, S. Lamprier, P. Gallinari. Learning distributed representations of users for source detection in online social networks. In *Proceedings of European Conference on Machine Learning and Knowledge Discovery in Databases*, Springer, Riva del Garda, Italy, pp. 265–281, 2016. DOI: 10.1007/978-3-319-46227-1_17.

[133] L. Wu, Y. Ge, Q. Liu, E. H. Chen, R. C. Hong, J. P. Du, M. Wang. Modeling the evolution of users′ preferences and social links in social networking services. *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 6, pp. 1240–1253, 2017. DOI: 10.1109/TKDE.2017.2663422.

[134] T. Zhang, R. Z. Qin, Q. L. Dong, W. Gao, H. R. Xu, Z. Y. Hu. Physiognomy: Personality traits prediction by learning. *International Journal of Automation and Computing*, vol. 14, no. 4, pp. 386–395, 2017. DOI: 10.1007/s11633-017-1085-8.

[135] G. W. Ma, Q. Liu, L. Wu, E. H. Chen. Identifying hesitant and interested customers for targeted social marketing. In *Proceedings of the 19th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining*, Springer, Ho Chi Minh City, Vietnam, pp. 576–590, 2015. DOI: 10.1007/978-3-319-18038-0_45.

Biao Chang received the B. Sc. degree in computer science from University of Science and Technology of China (USTC), China in 2012. He is now a Ph. D. degree candidate at School of Computer Science and Technology of USTC, China under the supervision of professor En-Hong Chen. He also visited Singapore Management Univercity as a research assistant under the supervision of professor Fei-Da Zhu from March 2015 to March 2016. His work has been published in conference proceedings including IJCAI, ICDM, CIKM.

His research interests include social network analysis and recommender systems.

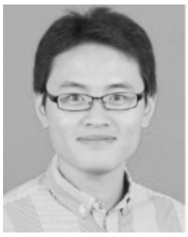
E-mail: changb110@gmail.com

ORCID iD: 0000-0002-5181-2397

Tong Xu received the Ph. D. degree in University of Science and Technology of China (USTC), China in 2016. He is currently working as a postdoctoral researcher of the Anhui Province Key Laboratory of Big Data Analysis and Application, USTC. He has authored nearly 20 journal and conference papers in the fields of social network and social media analysis, including KDD, AAAI, ICDM, SDM, etc. He was a recipient of the ACM (Hefei) Doctoral Dissertation Award, 2016.

His research interests include social network analysis and data mining.

E-mail: tongxu@ustc.edu.cn



Qi Liu received the Ph. D. degree in computer science from University of Science and Technology of China, China. He has published prolifically in refereed journals and conference proceedings, e.g., *IEEE Transactions on Knowledge and Data Engineering*, *ACM Transactions on Information Systems*, *ACM Transactions on Knowledge Discovery from Data*, *ACM Transactions on Intelligent Systems and Technology*, KDD, IJCAI, AAAI, ICDM, SDM, and CIKM. He has served regularly on the program committees of a number of conferences, and is a reviewer for the leading academic journals in his fields. He received the ICDM 2011 Best Research Paper Award and the Best of SDM 2015 Award. He is a member of ACM and the IEEE.

His research interests include data mining and knowledge discovery.

E-mail: qiliuql@ustc.edu.cn

En-Hong Chen received the Ph. D. degree from University of Science and Technology of China. He is a professor and vice dean of School of Computer Science, USTC. He has published more than 150 papers in refereed conferences and journals, including *IEEE Transactions on Knowledge and Data Engineering*, *IEEE Transactions on Industrial Electronics*, KDD, ICDM, NIPS, and CIKM. He was on program committees of numerous conferences including KDD, ICDM, and SDM. He received the Best Application Paper Award on KDD 2008, the Best Research Paper Award on ICDM 2011, and the Best of SDM 2015. His research is supported by the US National Science Foundation for Distinguished Young Scholars of China. He is a senior member of the IEEE.

His research interests includes data mining and machine learning, social network analysis, and recommender systems.

E-mail: cheneh@ustc.edu.cn (Corresponding author)