Front. Eng. Manag. 2022, 9(4): 520–549
https://doi.org/10.1007/s42524-022-0190-8

REVIEW ARTICLE

Yang OU, Qiang GUO, Jianguo LIU

Identifying spreading influence nodes for social networks

Abstract The identification of spreading influence nodes in social networks, which studies how to detect important individuals in human society, has attracted increasing attention from physical and computer science, social science and economics communities. The identification algorithms of spreading influence nodes can be used to evaluate the spreading influence, describe the node's position, and identify interaction centralities. This review summarizes the recent progress about the identification algorithms of spreading influence nodes from the viewpoint of social networks, emphasizing the contributions from physical perspectives and approaches, including the microstructure-based algorithms, community structure-based algorithms, macrostructure-based algorithms, and machine learning-based algorithms. We introduce diffusion models and performance evaluation metrics, and outline future challenges of the identification of spreading influence nodes.

Keywords complex network, network science, spreading influence, machine learning

1 Introduction

Complicated interactions between individuals can be well described by social networks, where nodes represent online users and offline individuals, and edges denote relations between nodes (Yan et al., 2013; Buyalskaya et al., 2021). Therefore, the social network has been studied by many branches of science in solving a wide range of problems, including information diffusion (Watts and Dodds, 2007; Muthukrishna and Schaller, 2020), spread of infectious disease (Jia et al., 2020; Bertozzi et al., 2020), formation of social relationships (Liben-Nowell and Kleinberg, 2007), and identification of online user reputation (Liu et al., 2017b; Dai et al., 2018). One of the key academic questions of social networks is the so-called identification of spreading influence nodes, which aims to find nodes that can maximize the scale of information diffusion. For instance, in online social platforms, such as Weibo, Facebook, and Twitter, a group of users serves as the influencer who can spread information widely and rapidly and arouse widespread concern and discussions of a topic in a short period (Lou and Tang, 2013). Identifying these spreading influence nodes is of importance for a great number of applications (Hou et al., 2014), such as viral marketing (Huang et al., 2019), controlling rumor (Borge-Holthoefer and Moreno, 2012), and fake news verification (Campan et al., 2017). Specifically, knowing the spreading influence of each node, marketing managers can accurately identify which influencers can help to promote their new products to target customers more effectively, thereby maximizing the use of advertising budgets. Official departments will be able to detect the rumor spreading sources and take corresponding measures before the rumor causes huge influence on social order. Online users can judge whether a news is fake by verifying the importance of news sources.

Identifying the spreading influence nodes of social networks is a long-standing challenge in modern social science, which has attracted considerable research effort over the past decades. Measure the influence of each node by conducting real-world experiments in social networks with more than millions of nodes is unfeasible because resources and time are limited. The mainstream ideology of this field is to estimate the spreading influence of nodes based on nodes' attributes and structural characteristics because it can sharply reduce the costs if the identification algorithms of spreading influence nodes are accurate (Lü et al., 2016; Liu et al., 2021). Existing literature reviews summarized the identification algorithms of

Received October 25, 2021; accepted February 14, 2022

Yang OU, Qiang GUO
Research Center of Complex Systems Science, University of Shanghai for Science and Technology, Shanghai 200093, China

Jianguo LIU (✉)
Institute of Accounting and Finance, Shanghai University of Finance and Economics, Shanghai 200433, China
E-mail: liujg004@ustc.edu.cn

This work was partially supported by the National Natural Science Foundation of China (Grant Nos. 72171150, 71771152, 61773248, and 71901144), and the Major Program of National Fund of Philosophy and Social Science of China (Grant Nos. 18ZDA088 and 20ZDA060).

spreading influence nodes with different focuses. Liu et al. (2013b) gave an overview of the identification algorithms of spreading influence nodes from the network topology and diffusion models' viewpoints and systematically analyzed the advantages and disadvantages of different algorithms. Ren and Lü (2013) introduced more than 30 different algorithms before 2014. Lü et al. (2016) presented a survey on the identification algorithms of spreading influence nodes and performance evaluation metrics, and compared the performance of representative methods in different types of networks. Liu et al. (2021) reviewed the algorithms developed on the basis of centralities (Freeman, 1977; 1978), PageRank (Brin and Page, 1998), and Hyperlink-Induced Topic Search (HITS) (Kleinberg, 1999). In addition to algorithms designed for the static network, the temporal network-based method has received increasing attention in recent years (Yang et al., 2018a; Yin et al., 2018; Guo et al., 2019) because it can well describe the dynamic characteristics of social systems. Chen et al. (2020) summarized three different types of algorithms proposed for the temporal network: The network topology-based, the random walk-based, and machine learning-based algorithms. Ren (2020) pointed out the challenges when applying the temporal network-based algorithms in the growing networks, time-varying networks, and perturbed temporal network.

A number of novel algorithms based on new techniques and ideas have emerged in recent years due to the confluence of improved computational capabilities, the explosive growth of new datasets, the increasing trend of interdisciplinary collaboration, and fast-changing demands. The existing algorithms can be classified into four categories in accordance with the type of structural attributes used to design identification algorithms: The microstructure-based (MSB), community structure-based (CSB), macrostructure-based (MASB), and machine learning-based (MLB) algorithms. Specifically, the MSB algorithms are developed to meet the efficient identification of spreading influence nodes in large-scale networks. Recent studies have given more attention to the relations and attributes of high-order neighbors rather than simply aggregating the structure information of nearest neighbors to enhance the accuracy of the MSB algorithms (Dai et al., 2019; Sun et al., 2019). Community structure information, which can help researchers to obtain a more comprehensive understanding of the social network, has been increasingly used to identify spreading influence nodes (Galvão et al., 2010; Ghalmane et al., 2019a; Zhang et al., 2019b). Combining the macro and micro structure information to enhance the generalizability of algorithms has become a new trend (Zareie et al., 2019; Namtirtha et al., 2021). The MLB algorithms have begun to appear in this field and still lack a systematic review (Yu et al., 2020; Fan et al., 2020). To catch up with the recent progress of the identification of spreading influence nodes, we present a review on new developments of the MSB, CSB, MASB, and MLB algorithms. We introduce diffusion models and performance evaluation metrics commonly used in studies of the identification of spreading influence nodes. We attempt to summarize the current challenges of this field. Details, including the methodology, study problems, data, and main findings of representative MSB, CSB, MASB, and MLB algorithms are summarized in Table 1.

The rest of this review is organized as follows. Section 2 presents the basic definitions of the social network and the description of the identification problem of spreading influence nodes. Section 3 introduces the MSB algorithms. Section 4 summarizes the CSB algorithms. Sections 5 and 6 discuss the MASB algorithms and the MLB algorithms, respectively. Sections 7 and 8 describe the diffusion models and performance evaluation metrics, respectively. Section 9 summarizes the future study trends and unsolved problems of this field.

2 Related definition and problem description

Let $G(V, E)$ be an unweighted network consisting of $|V| = n$ nodes and $|E| = m$ edges, where $V = \{v_1, v_2, ..., v_n\}$ and $E = \{e_{ij} \mid i, j = 1, 2, ..., n\}$ denote the set of nodes and the set of edges, respectively. In social networks, online users or offline individuals can be regarded as nodes, and edges describe the relations between these nodes. Considering whether the edge has weights and directions, social networks can be classified into the undirected and weighted network, the undirected and unweighted network, the directed and unweighted network, and the directed and weighted network, as shown in Fig. 1. The social network can also be represented by its adjacency matrix $\boldsymbol{A} = \{a_{ij}\}_{n\times n}$, where $a_{ij} = 1$ if node i is connected to node j, $a_{ij} = 0$ otherwise.

The identification task of spreading influence nodes in social networks is to find nodes that can cause a great influence on the structure of social networks or maximize the scale and speed of information spreading. Specifically, the identification of spreading influence nodes can be further divided into the task of node ranking and the task of influence maximization. The node ranking task refers to ranking nodes in descending order in accordance with their spreading influence scores obtained by applying an evaluation function $f(\cdot)$ of spreading influence nodes. The influence maximization problem aims to find a set of seed nodes S with a fixed-size k to achieve the maximum influence.

3 MSB algorithms

In the era of big data, social networks, such as Weibo, are characterized by large scale and intricate connections

Table 1 Approach names, categories, study problems, data, diffusion models, and main analysis findings of the representative MSB, CSB, MASB, and MLB algorithms

Approach	Category	Study problem	Data	Diffusion model	Main analysis findings
Percolation-based greedy algorithm (PBGA) (Hu et al., 2018)	MSB	Influence maximization	GrQc (Leskovec et al., 2007), HepTh (Leskovec et al., 2007), Enron (Leskovec et al., 2009), NoLA Facebook, DBLP (Yang and Leskovec, 2012), QQ (Ren et al., 2015), LiveJournal (Yang and Leskovec, 2012), Weibo, Delicious (Lü et al., 2011)	Susceptible-infected-recovered (SIR) (Hethcote, 2000)	The spreading influence of nodes can be approximately estimated by using the local structural information of nodes
Spreading strength (SS) (Yu et al., 2019)		Node ranking	Facebook (McAuley and Leskovec, 2012), PGP (Boguñá et al., 2004), Protein (Jeong et al., 2001), Guntella08 (Leskovec et al., 2007), GrQc (Leskovec et al., 2007), CondMat (Leskovec et al., 2007), HepTh (Leskovec et al., 2007), US Air, PowerGrid (Watts and Strogatz, 1998)	SIR (Hethcote, 2000)	The indirect influence of a node on its neighborhood is important for measuring the spreading influence of the node
Local centrality (LC) (Chen et al., 2012)		Node ranking	Blog (Xie, 2006), Netscience (Newman, 2006), Router (Spring et al., 2002), Email (Guimerà et al., 2003)	SIR (Hethcote, 2000)	The degree information of high-order neighbors can improve the accuracy and resolution of the degree centrality (DC)
Neighborhood centrality (NC) (Liu et al., 2016b)		Node ranking	Email (Guimerà et al., 2003), HepTh (Leskovec et al., 2007), Hamster (Kunegis, 2016), PGP (Boguñá et al., 2004), Astro Physics (Newman, 2001), Router (Spring et al., 2002)	SIR (Hethcote, 2000)	Considering the structural information of a node's neighbors within two steps is a good choice to balance accuracy and efficiency
Local structure similarity (LSS) (Liu et al., 2017a)		Influence maximization	GrQc (Leskovec et al., 2007), Router (Spring et al., 2002), Hamster (Kunegis, 2016), Polblogs	SIR (Hethcote, 2000), Susceptible-infected (SI) (Barabási and Albert, 1999)	The local structural property of nodes can help identify multiple spreading influence nodes more accurately than by using the distance
VoteRank (Zhang et al., 2016)		Influence maximization	YouTube (Yang and Leskovec, 2012), CondMat (Leskovec et al., 2007), Berkstan (Leskovec et al., 2009), Notre DAME (Albert et al., 1999)	SIR (Hethcote, 2000), SI (Barabási and Albert, 1999)	The performance of VoteRank is highly correlated with the number of ranked nodes
ClusterRank (CR) (Chen et al., 2013)		Node ranking	Delicious (Lü et al., 2011), SM	SIR (Hethcote, 2000)	Nodes with small clustering coefficients are likely to connect with more nodes in the future
Local structure centrality (LSC) (Gao et al., 2014)		Node ranking	Email (Guimerà et al., 2003), Blog, PGP (Boguñá et al., 2004), Twitter	SIR (Hethcote, 2000)	The positive effect of the clustering coefficient of a node's second-order neighbors has a significant influence on the spreading influence of the node
V-communities (Vc) (Zhao et al., 2014b)	CSB	Node ranking	Facebook (McAuley and Leskovec, 2012), GrQc (Leskovec et al., 2007), Netscience (Newman, 2006), Protein (Jeong et al., 2001)	SIR (Hethcote, 2000)	The number of communities connected to a node can help detect the spreading influence nodes that a single centrality may ignore
Community-based centrality (CbC) (Zhao et al., 2015)		Node ranking	Facebook (McAuley and Leskovec, 2012), Metabolic, Email, PowerGrid (Watts and Strogatz, 1998), Router (Spring et al., 2002), Blogcatalog	SIR (Hethcote, 2000)	The size of the community and the distribution of a node's neighbors in each community play important roles in measuring the node's spreading influence
Community-based mediator (CbM) (Tulu et al., 2018)		Node ranking	Karate (Zachary, 1977), American football network (Girvan and Newman, 2002), Dolphin (Lusseau et al., 2003), Airport, Internet	SIR (Hethcote, 2000)	The edge density within each community and the edge density between communities can be used to identify spreading influence nodes accurately with low computational complexity
Community-hole index (CHR) (Wang et al., 2018)		Node ranking	GrQc (Leskovec et al., 2007), Weibo (Tang and Liu, 2009), arXiv (Pan and Saramäki, 2012), Amazon	SIR (Hethcote, 2000)	The importance of communities connected to a node is related to the node's spreading influence
Modular centrality (MC) (Ghalmane et al., 2019a)		Node ranking	Facebook (McAuley and Leskovec, 2012), Netscience (Newman, 2006), GrQc (Leskovec et al., 2007)	SIR (Hethcote, 2000)	Dividing the network with an overlapping community structure into local and global networks can help identify spreading influence nodes more accurately
Network global structure-based centrality (NGSC) (Namtirtha et al., 2021)	MASB	Influence maximization	Odlis, Netscience (Newman, 2006), Advogato (Massa et al., 2009)	SIR (Hethcote, 2000)	The network components and network density have a significant influence on the performance of identification algorithms
Gravity centrality (GC) (Ma et al., 2016)		Node ranking	Facebook, Netscience (Newman, 2006), Email (Guimerà et al., 2003), TAP (Zeng and Zhang, 2013), Y2H (Kumar and Snyder, 2002), Blogs (Xie, 2006), Router (Spring et al., 2002), HepTh (Leskovec et al., 2007), PGP (Boguñá et al., 2004)	SIR (Hethcote, 2000)	The gravity model can be applied to identify the spreading influence of nodes with relatively high accuracy

(*Continued*)

Approach	Category	Study problem	Data	Diffusion model	Main analysis findings
C_{EffG} (Shang et al., 2021)		Node ranking	Jazz, Netscience (Newman, 2006), GrQc (Leskovec et al., 2007), EEC, Email, PB, Facebook, US Air, Physicians, PDZBase, Haggle, Infectious	SI (Barabási and Albert, 1999)	Replacing the Euclidean distance of the gravity model by the effective distance can achieve higher accuracy
Dynamic-sensitive (DS) (Liu et al., 2016a)		Influence maximization	Erdos, Email contact (Kitsak et al., 2010), Router (Spring et al., 2002), Protein (Jeong et al., 2001)	SIR (Hethcote, 2000), SI (Barabási and Albert, 1999)	The spreading influence of a node is determined by topological structures and the spreading dynamics
Influence capacity (Wang et al., 2016)		Node ranking	Karate (Zachary, 1977), Netscience (Newman, 2006), Dolphin (Lusseau et al., 2003), Email (Guimerà et al., 2003), Jazz, PGP (Boguñá et al., 2004), Blog (Xie, 2006), Facebook, Enron (Leskovec et al., 2009), Twitter	SIR (Hethcote, 2000)	The order in which nodes within the same layer are removed can distinguish the spreading influence of nodes located in the same shell
Link entropy (Liu et al., 2015a)		Node ranking	Router (Spring et al., 2002), Email contact (Kitsak et al., 2010), AS, Email (Guimerà et al., 2003), HepTh (Leskovec et al., 2007), Hamster (Kunegis, 2016), PGP (Boguñá et al., 2004), Netscience (Newman, 2006), Astro Physics (Newman, 2001)	SIR (Hethcote, 2000)	The node with high coreness is a spreading influence node with a strong edge diversity to nodes located in other shells of the network
θ (Liu et al., 2013a)		Node ranking	Email (Guimerà et al., 2003), PGP (Boguñá et al., 2004), AS, P2P (Leskovec et al., 2007)	SIR (Hethcote, 2000)	The distance from the node to nodes in the core–shell layer can effectively distinguish the spreading influence of nodes in the same shell layer
Multicentrality predictors (Bucur, 2020)	MLB	Node ranking	Adolescent, Advogato (Massa et al., 2009), Astro Physics (Newman, 2001), CondMat (Leskovec et al., 2007), GrQc (Leskovec et al., 2007), HepTh (Leskovec et al., 2007), AS, Brightkite, Email, Epinions, Euroroad, Facebook, Github, Guntella, Googleplus, Hamster (Kunegis, 2016), IMDB, OpenFlights, PGP (Boguñá et al., 2004), Twitch, Twitter Stanford, US Airports, PowerGrid (Watts and Strogatz, 1998), WikiTalk	SIR (Hethcote, 2000)	The spreading influence nodes identified by the MSB algorithms may be located in the peripheral regions of the network, and the MASB algorithms can help to rectify this shortcoming
Perturb and combine (P&C) (Tixier et al., 2019)		Node ranking	Email (Klimt and Yang, 2004), Epinions, WikiVote	SIR (Hethcote, 2000)	The idea of ensemble learning can improve the robustness of the k-core and PageRank algorithm
Influence deep learning (IDL) (Wang et al., 2019)		Node ranking	Sina Weibo, Epinions, WikiVote, NetHEPT (Pal et al., 2014)	Independent cascading (IC) (Kempe et al., 2003)	The graph convolutional networking (GCN) model has a great potential for the identification of spreading influence nodes

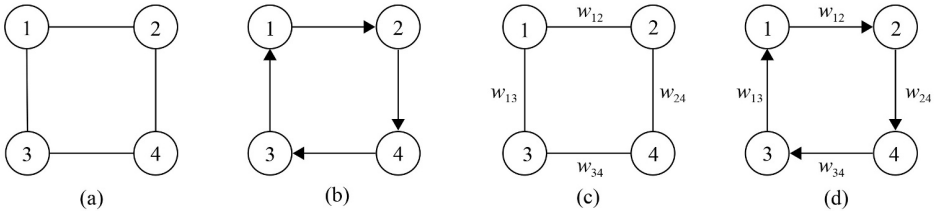

Fig. 1 Diagram of four types of networks: (a) undirected and unweighted network; (b) directed and unweighted network; (c) undirected and weighted network; and (d) directed and weighted network.

between users. For such networks, directly using the structure information of the entire network to identify spreading influence nodes will be costly and inefficient. To develop efficient identification algorithms that can be applied to large-scale social networks, an increasing number of researchers attempt to identify spreading influence nodes by considering only the micro-level structural information. Hu et al. (2018) proved the feasibility of identifying the spreading influence nodes via the micro-level structure information on the basis of percolation theory (Dorogovtsev et al., 2008). They found the nucleation behavior of the spreading process, that is, if the number of individuals influenced by spreading sources is larger than a small characteristic number, then the information will rapidly reach the percolation cluster regardless of the global structure of the network, and it will be contained within a local area otherwise. Over the past decades, a great number of the MSB algorithms that can achieve relatively high accuracy while keeping low complexity have been proposed (Liu et al., 2017a; Bao et al., 2017).

The simplest MSB algorithm is the degree centrality (DC) (Freeman, 1978), which defines the number of a node's first-order neighbors as its spreading influence,

which is given as

$$DC(i) = \frac{k(i)}{n-1}, \tag{1}$$

where $k(i)$ denotes node i's degree, and n is the total number of nodes in the network. In common sense, information shared by online users may influence their followers and indirectly affect followers' friends. Having users with the same number of followers on online social platforms is common. In such cases, only considering the number of directly connected nodes may be extremely naive. An example is shown in Fig. 2 that although the degree centralities of nodes 1 and 2 are the same, the difference in the number of second-order neighbors will be ignored by the DC method.

In spite of the above limitations, many MSB algorithms borrowed the idea of the DC method because it requires the least information. The local centrality (LC) (Chen et al., 2012) improved the performance of the DC method by simply aggregating the degree of high-order neighbors. However, for large-scale social networks, the more inclusion of information of higher-order neighbors, the better the performance of the identification algorithm is not always the case. Therefore, deciding how many steps of neighbor nodes to consider is a key point in balancing the accuracy and computational complexity of identification algorithms. Liu et al. (2016b) compared the changes in accuracy and complexity of their proposed algorithm called neighborhood centrality (NC) when using the structural information of different order neighbors. They found that once the structure information of more than three-order neighbors are used, the accuracy will not obtain remarkably improvement while the complexity increases intensely. The NC algorithm is given as

$$NC_i^l = r_i + a\sum_{j\in\Gamma_i} r_j + a^2\sum_{z\in\Gamma_j\setminus i} r_z + a^3\sum_{o\in\Gamma_z\setminus j} r_o + \ldots + a^l\sum_{g\in\Gamma_{g-1}\setminus x} r_g, \tag{2}$$

where r denotes a benchmark identification algorithm of spreading influence nodes, Γ_i is the set of first-order neighbors of node i, $a \in [0, 1]$ is a free parameter, and l represents the neighbor's order. As shown in Eq. (2), the NC algorithm assuming a node's spreading influence will be largely affected by neighbors close to it and will be slightly influenced by distant neighbors. In other words, Liu et al. (2016b) thought that the neighbors' contribution to the node's spreading influence is related to distances from the neighbor nodes to the target node.

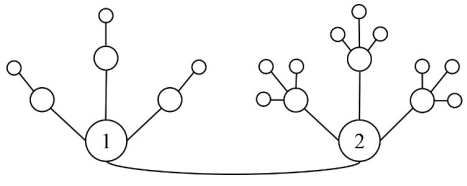

Fig. 2 Diagram of nodes with the same number of first-order neighbors but a different number of second-order neighbors.

The local neighbor contribution (LNC) algorithm (Dai et al., 2019) considers the node's self-influence and its neighbors' contribution to the spreading influence, which is given as

$$LNC(i) = \left(D(i)\sum_{j=1}^{h} P(j)DC(j)\right) \times \left(k(i)\sum_{j=1}^{k(i)} C_1^{k(i)} P(j)(1-P(j))^{(k(i)-1)}\right), \tag{3}$$

where $D(i)$ is the sum of neighbors' degree of node i and h denotes number of first- and second-order neighbors of node i. $P(j) = 1/k(j)$ represents node j's contribution to the spreading influence of node i. $C_1^{k(i)}$ represents the combination in mathematics.

Except for the difference in neighbors' contributions on the spreading influence of the node, two connected nodes, i and j, have different influence on each other in directed networks. Yu et al. (2019) thought that this difference should be considered in undirected networks and defined the spreading strength (SS) of node i on node j, that is, c_{ij}, as the combination of the direct influence of node i on node j and the indirect influence of node i's neighbors on node j, which is given as

$$c_{ij} = 1 + k_j^{\text{out}}\left(1 + \frac{|d_{ij,2}|}{4}\right)^a, \tag{4}$$

where k_j^{out} denotes the number of node j's neighbors that are not neighbors of node i, $|d_{ij,2}|$ represents the number of paths between i and j with length 2, and a is a free parameter.

The algorithms based on the degree information of high-order neighbors, that is, high-order-neighbor-degree-based algorithms, are efficient and easy to understand. However, in addition to the degree information of neighbor nodes, the topological connection of a node's neighbors, which indicates the potential for the node to spread information to other parts of networks, plays an important role in measuring its spreading influence (Soffer and Vázquez, 2005; Ren et al., 2013b). The local clustering coefficient c_i is used to design the MSB algorithms for quantifying the interactions between neighbor nodes. Mathematically, the local clustering coefficient in undirected networks is defined as

$$c_i = \frac{2\left|e_{jv} \mid j,\ v \in \Gamma_i\right|}{k(i)(k(i)-1)}. \tag{5}$$

Soffer and Vázquez (2005) uncovered the negative correlation between the clustering coefficient and degree value in undirected networks. On this basis, Chen et al. (2013) further discovered that nodes with small

clustering coefficient are likely to connect with more nodes in the future and proposed ClusterRank (CR) method, represented by s_i, which is defined as

$$s_i = f(c_i)\sum_{j\in\Gamma_i}\left(k_j^{\text{out}}+1\right), \tag{6}$$

where $f(c_i)$ represents the effect of node i's local clustering coefficient on its spreading influence. Specifically, $f(c_i)$ is a decreasing function concerning the local clustering coefficient because a higher local clustering coefficient of a node indicates that its neighbors interact with each other closer than with nodes in other parts of networks, resulting in information shared by the node being easily contained only in a local area. In addition to the local clustering coefficient of first-order neighbors, Gao et al. (2014) proposed local structure centrality (LSC) by further considering the positive effect of the local clustering coefficient of a node's second-order neighbors on its spreading influence, which is given as

$$LSC(i) = \sum_{j\in\Gamma_i}\left(aN(u)+(1-a)\sum_{v\in\Gamma_j^2}c_v\right), \tag{7}$$

where $N(u)=\left|\Gamma_j^2\right|$ is the total number of a node's first- and second-order neighbors, and $a\in[0,1]$ is a free parameter that can be adjusted in accordance with structural attributes of networks to guarantee a stable performance. However, the adjustment of the free parameter leads to high extra computational complexity. Berahmand et al. (2018) presented a parameter-free centrality algorithm that considers the degree and the local clustering coefficients of a node's first- and second-order neighbors, which is defined as

$$Centrality(i) = \frac{k(i)}{c_i+1/k(i)}+\sum_{j\in\Gamma_i^2}c_j. \tag{8}$$

Inspired by the positive effect of the local clustering coefficient of second-order neighbors on the spreading influence of the node, Yang et al. (2020) introduced entropy technology to calculate the weights assigned to degree and local clustering coefficient, and proposed a novel centrality (DCC) algorithm, which is given as

$$DCC(i) = aI_{\text{D}}(i)+bI_{\text{C}}(i), \tag{9}$$

where $I_{\text{D}}(i)=k(i)+\sum_{j\in\Gamma_i}k(j)$ accounts for the effect of degree and neighbors' degree of node i, $I_{\text{C}}(i)=e^{-c_i}\sum_{j\in\Gamma_i^2}c_j$ measures the effect of first- and second-order neighbors' local clustering coefficients, and $a+b=1$.

The introduction of the local clustering coefficient enables the identification algorithms to have a higher resolution in distinguishing the spreading influence of nodes with the same number of neighbors, and local clustering coefficient-based algorithms can identify nodes located in the dense parts of networks. However, nodes located in locally dense but globally peripheral regions of the network are easily misclassified as spreading influence nodes because the local-clustering-coefficient-based algorithms give less attention to the position of nodes. The main challenge of designing an effective MSB algorithm for identifying multiple spreading influence nodes is how to reduce the overlapping spreading influence of selected nodes. Taking DC algorithm as an example, social networks often have the heterogeneous property, that is, nodes with a small degree are more likely to connect with nodes with a large degree. Seed nodes selected by the DC method are easy to gather within a local area of social networks (Barabási and Bonabeau, 2003; Zhou et al., 2018). One of the main ideas to alleviate this problem is to avoid selecting seed nodes with similar topological attributes within a local area to ensure that they are distributed in different parts of the network. Sheikhahmadi et al. (2015) considered the distance and the number of common neighbors between seed nodes to ensure that selected nodes are evenly distributed in the network. However, calculating the distances between each pair of nodes in large-scale networks is time consuming. Instead of using distance, Liu et al. (2017a) proposed local structure similarity (LSS), which sets the structure similarity between seed nodes as a constraint when identifying seed nodes. Specifically, LSS first selects the node with the largest degree as the initial seed and then find other seeds in an iterative manner. In each iteration, the next seed node will be chosen from the first- and second-order common neighbors of all selected seed nodes in accordance with the structure similarity score s, which is given as

$$s_{ij} = \frac{\left|P_i\cap\Gamma_j\right|}{k(i)}, \tag{10}$$

where P_i denotes the set of first-order neighbors of node i, Γ_j is the first-order neighbors of node j if node i is connected to node j, and Γ_j is the set of first- and second-order neighbors of node j if node j is the second-order neighbor of node i. If the structure similarity scores between the target node and all seed nodes are smaller than a given threshold γ, then the target node will be selected as the next seed node. To ensure that seed nodes are detected from different parts of the network, Bao et al. (2017) borrowed the idea of k-means clustering (Macqueen, 1967) and presented heuristic clustering (HC) to reduce the overlapping spreading influence. Initially, centers of k clusters are randomly chosen. Noncenter nodes are classified into these clusters in accordance with the local path similarity (Zhou et al., 2009) between them and center nodes, which is defined as

$$\boldsymbol{W} = \boldsymbol{A}^2+a\boldsymbol{A}^3, \tag{11}$$

$$B(i) = \sum_{j\in C_t}W_{ij}, \tag{12}$$

where C_t denotes the set of nodes in cluster t, $\boldsymbol{W}$ is the similarity matrix, a is a free parameter, and $B(i)$ is used to update the cluster center, which represents the significant of node i in cluster t. The node classifying step and center updating step are repeated until the steady-state is reached. The center of each cluster is selected as the seed node.

Similarity-based algorithms ensure that those seed nodes will not gather in a local region of networks by setting the number of common neighbors or distance between seed nodes as the constraint. However, the performance of this type of algorithms is sensitive to the selection of the initial seed node. For instance, the LSS algorithm only considers the degree of initial seed node, and structural attributes, such as the local clustering coefficient and the node's position, which are proven to have a significant influence on the spreading influence of nodes, are ignored.

Inspired by the voting process, Zhang et al. (2016) proposed the VoteRank algorithm. In the initial phase, all nodes will be assigned the same voting ability Z_{i0} and voting score S_{i0}. Nodes will then start to vote for their neighbors in an iterative manner. In each voting round, node i's voting score S_i will be updated by using the following equation

$$S_i = \sum_{j \in \Gamma_i} Z_j. \tag{13}$$

The node with the highest voting score of each round will be selected as the seed node, and its voting score will be reset to 0 in the next round. The voting abilities of the node's neighbors will be decreased in the next round to avoid the overlapping influence problem. The VoteRank algorithm initially sets the voting abilities of all nodes to 1, which implicitly assumes that neighbors are the same as the target node, thereby leading to low resolution. To solve this problem, inspired by social conformity theory and community structure of networks, Zhang et al. (2019b) improved VoteRank from the viewpoints of the individual and the group. In common sense, attractiveness between individuals are different, which is quantified by the node's in-degree and out-degree, that is, attractive power (AP), to measure the individual-level voting ability of the node, which is given as

$$AP(i,\ j) = \begin{cases} \dfrac{\left|\Gamma_i^{\text{out}}\right|}{\sum\limits_{v \in \Gamma_j^{\text{out}}} \left|\Gamma_v^{\text{in}}\right|}, & \sum\limits_{v \in \Gamma_j^{\text{out}}} \left|\Gamma_v^{\text{in}}\right| \neq 0 \\ \dfrac{1}{\left|\Gamma_j^{\text{out}}\right|}, & \sum\limits_{v \in \Gamma_j^{\text{out}}} \left|\Gamma_v^{\text{in}}\right| = 0,\ \Gamma_j^{\text{out}} \neq \varnothing \end{cases}, \tag{14}$$

where Γ_i^{in} and Γ_i^{out} denote the set of in- and out-neighbors of node i in the directed network, respectively. The group-level voting ability, that is, initiating power (IP), is measured by the size of the community that the node belongs, which is defined as

$$IP(i,\ j) = \begin{cases} 0, & N_{Com_i} = N_{Com_j} \\ \dfrac{\left|N_{Com_j}\right|}{\max N_{Com_v}}, & v \in V,\ N_{Com_i} \neq N_{Com_j} \end{cases}, \tag{15}$$

where N_{Com_i} represents the size of the community to which node i belongs. The voting score of each node is calculated as follows

$$S_i = \sum_{j \in \Gamma_i} (AP(i,\ j) + IP(i,\ j)). \tag{16}$$

Zhang et al. (2019b) presented node selection strategies from individual and group perspectives to reduce the overlapping spreading influence. From the viewpoint of the individual, the node will be removed from the network once the node is selected as the seed node. From the group viewpoint, the candidate will not be selected when the community to which the candidate node belongs is strongly connected with the communities that the seed nodes belong to. Kumar and Panda (2020) introduced the neighborhood coreness algorithm (NCRank) to enhance the resolution of the VoteRank algorithm. Considering the k-shell values of neighbors, the voting score is given as

$$S_i = \sum_{j \in \Gamma_i} (Z_i \times (1-a) \times C_{\text{nc}}(j)) + Z_i \times a, \tag{17}$$

where $C_{\text{nc}}(j)$ represents the neighborhood coreness of node j, and $a \in [0,\ 1]$ denotes a free parameter. Guo et al. (2020) proposed the EnRenew algorithm by considering the difference between nodes when decreasing their influence on the basis of information entropy, which is defined as

$$entropy(i) = \sum_{j \in \Gamma_i} H_{ij} = -\sum_{j \in \Gamma_i} p_{ij} \log p_{ij}, \tag{18}$$

$$p_{ij} = \frac{k(i)}{\sum\limits_{l \in \Gamma_j} k(l)}. \tag{19}$$

After the node is selected, the voting scores of its l-order neighbors will be decreased by using following equation to avoid the overlapping spreading influence

$$H_{j^{l-1}j^l} = \frac{1}{2^{l-1}} \frac{H_{j^{l-1}j^l}}{entropy(i)_{\langle k \rangle}}, \tag{20}$$

$$entropy(i)_{\langle k \rangle} = \log \frac{1}{\langle k \rangle}, \tag{21}$$

where j^l denotes the l-length reachable nodes of node j, and $\langle k \rangle$ is the average degree of the network. Sun et al. (2019) extended the use of VoteRank in weighted

networks by considering the edge weight and proposed the WvoteRank algorithm. The voting score of the node in weighted networks is defined as

$$S_i = \sqrt{|\Gamma_i| \sum_{j \in \Gamma_i} Z_j w_{ij}}, \tag{22}$$

where w_{ij} is the weight of edge e_{ij}.

The VoteRank-based algorithms are efficient because no distance calculation is included. However, the initial state of each node and the spreading influence decreasing strategy will cause a large influence on the performance of the VoteRank-based algorithms.

The spreading influence of nodes depend on their structural attributes and the spreading mechanism. On the basis of the spreading mechanism of the independent cascading (IC) model, the degree discount (DD) algorithm (Chen et al., 2009) selects a set of seed nodes by discounting a node's degree in accordance with the number of seeds in its neighborhood to alleviate the overlapping spreading influence between seed nodes. Chen et al. (2019) assumed that if the information propagated by a node can easily influence its high-order neighbors, this node will be considered to have more potential to initiate large-scale propagation. Considering the diffusion process and the sum of probabilities that high-order neighbors being influenced by the target node, the spreading influence of nodes is quantified by using the following equation

$$Rank(i) = \sum_{l=1}^{3} \sum_{j \in \Gamma_i^l} score(j, l), \tag{23}$$

where $score(j, l)$ represents the probability of the l-order neighbor j being influenced by node i, which is defined as

$$score(j, l) = 1 - uninF_s(j, l), \tag{24}$$

$$uninF_s(j, l) = \prod_{v \in \Gamma_i^{l-1}} (1 - score(v, l-1) \times \beta), \tag{25}$$

where $uninF_s(j, l)$ represents the probability that node j is not infected by nodes belong to Γ_i^{l-1}, β denotes the infection rate, and $score(v, 0) = 1$.

The diffusion model-based algorithms consider the spreading dynamics when evaluating the spreading influence of nodes, which is more in line with reality. However, the diffusion mechanism varies with the spreading events, thereby restricting the applications of this type of algorithms in different scenarios.

The MSB algorithms have offered solutions for identifying spreading influence nodes in large-scale social networks. Specifically, the MSB algorithms can be further classified into five main method streams based on the idea used to design identification algorithms: The high-order-neighbor-degree-based, the local-clustering-coefficient-based, the similarity-based, the VoteRank-based, and the diffusion-model-based methods. The advantages and disadvantages of representative algorithms and the five main method streams mentioned in this section are listed in Tables 2 and 3. Although the MSB algorithms have achieved promising performance, several challenges still need to be addressed in the future.

For the node ranking problem, the high-order-neighbor-degree-based algorithms are efficient because they mainly focus on the degree values of nodes. However,

Table 2 Advantages and disadvantages of representative MSB algorithms, where r is the total number of iteration rounds, $\langle k \rangle$ is the average degree of nodes, and n and m are total number of nodes and edges of a network, respectively

Methods	Advantages	Disadvantages	Computational complexity
DC (Freeman, 1978)	Low computational complexity; Simple and easy to understand	The structure information of high-order neighbors is ignored	$O(n)$
LC (Chen et al., 2012)	Considers the degree of high-order neighbor nodes	Other structural attributes are disregarded, except the degree of the node	$O(\langle k \rangle n^2)$
LNC (Dai et al., 2019)	Measures the contributions of different neighbors to the spreading influence of the node; Outperforms DC and betweenness while keeping low computational complexity	Unsuitable for the random network	$O(\langle k \rangle n)$
VoteRank (Zhang et al., 2016)	Uses the idea of voting to aggregate the structural information of high-order neighbors; The accuracy is higher than PageRank and LeaderRank; Low computational complexity	The differences in the initial voting abilities of nodes are ignored	$O(n)$
AIRank (Zhang et al., 2019b)	Distinguishes the voting abilities of nodes from the perspective of individuals and groups	The performance will be unstable when the edge density between communities is low	$O(n)$
NCRank (Kumar and Panda, 2020)	Distinguishes the voting abilities of nodes by considering the position of the node	Adjusting free parameters to obtain a stable performance is time consuming	$O(n)$
EnRenew (Guo et al., 2020)	Different initial voting abilities of nodes are distinguished on the basis of the information entropy	The computational complexity is higher than VoteRank	$O\left(m+n+r\log(n)+\frac{m^2}{n^2}\right)$
DynamicRank (Chen et al., 2019)	Uses the probabilities of high-order neighbors being influenced to measure the spreading influence of nodes	Adjusting the free parameter to obtain a stable performance is time consuming	$O(n)$

Table 3 Advantages and disadvantages of five main MSB method streams

Method streams	Related works	Advantages	Disadvantages
High-order-neighbor-degree-based	LC (Chen et al., 2012); NC (Liu et al., 2016b); LNC (Dai et al., 2019); SS (Yu et al., 2019)	Efficient and easy to understand	Topological information, such as interactions between neighbors and the edge density, is ignored
Local-clustering-coefficient-based	CR (Chen et al., 2013); LSC (Gao et al., 2014); Centrality (Berahmand et al., 2018); DCC (Yang et al., 2020)	Higher resolution; Enables the spreading influence of nodes with the same degree value to be further distinguished	The performance will be suppressed in densely connected networks because it only focuses on the edge density in the node's neighborhood
Similarity-based	DegreeDistance (Sheikhahmadi et al., 2015); LSS (Liu et al., 2017a); HC (Bao et al., 2017)	Ensures that seed nodes are distributed in different parts of the network	The selection of initial seed nodes has a large influence on its performance
VoteRank-based	VoteRank (Zhang et al., 2016); AIRank (Zhang et al., 2019b); NCRank (Kumar and Panda, 2020); EnRenew (Guo et al., 2020)	No distance calculation is included	The initial state of each node and the spreading influence decreasing strategy will cause large influence on its performance
Diffusion-model-based	DD (Chen et al., 2009); DynamicRank (Chen et al., 2019)	The topological information and the diffusion mechanism are considered	When the diffusion mechanism changes, its performance will be affected

the topological connections between neighbors did not receive sufficient attention. The local-clustering-coefficient-based algorithm improved this shortcoming by considering the relations of a node's first- and second-order neighbors. The two types of MSB algorithms implicitly assume that spreading influence nodes are nodes located in densely connected regions of networks, which might lead to identification algorithms performing poorly in networks with a high edge density because the community structure and macro-level information are ignored. Therefore, improving the performance of MSB algorithms in densely connected networks while keeping low computational complexity can be a future direction worthy of attention.

For the influence maximization problem, the similarity-based algorithms try to reduce overlapping spreading influence by suppressing the similarity between seed nodes, which depend on the selection of initial seed nodes and similarity measurements. Researchers are encouraged to modify the initial seed node selection strategies and similarity indices to develop more effective algorithms. The VoteRank-based algorithms select seed nodes by considering neighbors' contributions in the spreading influence of nodes in an iterative manner. How to quantify the difference in neighbors' contributions is still worthy of study. Although diffusion-model-based algorithms consider the topological attributes of nodes and spreading dynamics when choosing seed nodes, they depend on a specific diffusion model. Whether this type of algorithms can work well when using different diffusion models needs to be further explored.

4 CSB algorithms

Community structure is a ubiquitous characteristic of social networks (Girvan and Newman, 2002; Yang et al., 2018b; Dong et al., 2021) because individuals tend to organize as groups based on their interests, occupations, social status, and other attributes. Each community can be viewed as a subnetwork in social networks, where nodes in the same subnetwork are densely connected, and edges between subnetworks are relatively sparse. Exploring the community structure can help researchers obtain an in-depth understanding of the network structure and the mechanism of the spreading process. For example, information transmission between communities requires the participation of individuals serving as the bridge. These individuals often have strong control over the information flow in common sense. In terms of this idea, Zhao et al. (2014) combined the centralities with index of the number of communities a node is connected with to identify the spreading influence nodes and discovered that this strategy can help detect spreading influence nodes that single centrality may ignore. However, the community structure of a network may change when applying different community detection algorithms (Palla et al., 2005; Newman, 2006; Pan et al., 2010; Tang et al., 2016), leading to this strategy being unstable. To alleviate this shortcoming, Zhao et al. (2015) considered the size of the community and the distribution of neighbors to reduce the dependence on community detection algorithms (Cantwell and Newman, 2019) and proposed community-based centrality (CbC), which is defined as

$$CbC(i) = \sum_{q=1}^{c} k_{iq} \frac{N_{Com_h}}{n}, \tag{26}$$

where k_{iq} denotes the number of node i's neighbors in community q, c is the total number of communities, and N_{Com_h} is the size of community h. Tulu et al. (2018) introduced the Shannon entropy to measure the spreading influence of nodes and proposed the community-based mediator (CbM), which defines the relations between the target node and nodes in the same community it belongs and nodes in other communities as its spreading influence, which is given as

$$CbM(i) = H(i) \times \frac{k(i)}{\sum_{j=1}^{n} k(j)}, \tag{27}$$

$$H(i)=\left(-\sum p_i^{\text{in}}\log\left(p_i^{\text{in}}\right)\right)+\left(-\sum p_{ih_1}^{\text{ex}}\log\left(p_{ih_1}^{\text{ex}}\right)\right), \quad (28)$$

where $H(i)$ denotes the internal entropy and external edge density of node i, and h_1 is the community. $p_{ih_1}^{\text{ex}}$ and p_i^{in} represent the external and internal edge densities of node i, respectively. Zhao et al. (2020c) improved the accuracy of closeness centrality (CC) (Sabidussi, 1966) by introducing the community structure information of the node and its neighbors. The mathematical formulation of the improved CC (ICC) is given as

$$ICC(i)=CC(i)\frac{N_{Com_i}}{n}+\sum_{w\in W_i}\max\{CC(j)\}\frac{N_{Com_w}}{n},\ (j\in W), \quad (29)$$

where $CC(i)$ denotes the CC of node i, and W_i is the set of communities the node i's neighbors are connected to, except the community containing node i.

From the perspective of network division based on the community structure of networks, Ghalmane et al. (2019b) proposed a CSB spreading influence identification framework called modular centrality (MC). Specifically, the whole process is as follows: 1) Construct the local network and the global network by removing edges between communities and within each community; 2) Calculate the spreading influence of nodes in the local network and the global network by using the selected node spreading influence identification algorithm; and 3) The final spreading influence of a node is the sum of its spreading influence on local and global networks. They found that identifying spreading influence nodes in the local network will be more accurate than in the global network when a clear community structure is found in the original network, and the accuracy will be higher in the global network otherwise. In the real world, one node can belong to multiple communities, indicating that networks may have an overlapping community (OC) structure. In such a case, the local and global networks constructed by the proposed framework will be inappropriate. Therefore, Ghalmane et al. (2019a) modified the network construction rule, which is shown in Fig. 3.

To detect the spreading influence nodes in networks with the OC structure, Wei et al. (2018) used the BigCLAM model (Yang and Leskovec, 2013) to identify the OC structure of the network. The seed nodes are found in accordance with network constraint coefficient and the number of communities the first- and second-order neighbors are connected with. The mathematical formulation of the proposed method is defined as

$$OC(i)=\frac{\sum_{j\in\Gamma_i}\sum_{v\in\Gamma_j}10^{-C_k}\times Com_N(v)}{\max\{OC(j)\mid j\in V\}}, \quad (30)$$

where C_k denotes the network constraint coefficient of node v, and $Com_N(v)$ is the number of communities node v is connected with.

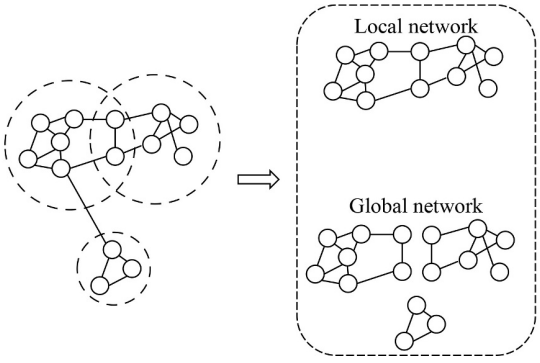

Fig. 3 Diagram of local and global network division of the network with overlapping community structure.

The CSB node ranking algorithms have shown advantages of using the community-level information to identify spreading influence nodes. However, the computational complexity of identification algorithms will increase because more community structural attributes are considered. Therefore, filtering significant attributes is a key step for designing an effective CSB algorithm. The use of the community structure information by splitting networks into subnetworks provides a novel perspective for designing CSB algorithms. However, the community structure of networks depends on community detection algorithms because the community structure of most real-world networks is unknown, which may affect the performance of this type of algorithm.

In addition to helping in analyzing network structure, community structure information can be used to reduce the time complexity when dealing with the influence maximization task. On the basis of submodular property of the influence spread (Galstyan and Cohen, 2007), Halappanavar et al. (2016) found that the overall influence of seed nodes selected from each community independently approximates to the influence of seed nodes identified by using the global structure information. Inspired by the community structure of social networks, Wang et al. (2010) developed a community-based greedy algorithm (CGA) based on the IC model (Kempe et al., 2003), which is more efficient than the MixGreedy algorithm (Kempe et al., 2003). Specifically, the CGA algorithm uses dynamic programming to determine which community can bring the largest importance gain ΔR_i in each step to narrow the searching space from the network level to the community level. The largest importance gain that community i, Com_i, can bring is defined as

$$\Delta R_i=\max\{R_i(I_{k-1}\cup v)-R_i(I_{k-1})\mid v\in Com_i\}, \quad (31)$$

where I_{k-1} denotes the set of nodes with the size $k-1$, and $R_i(I_{k-1})$ represents the overall spreading influence of I_{k-1}. On the basis of Eq. (31), the community selection strategy is defined as

$$R(u,\ k)=\max\{R(u-1,\ k),\ R(u,\ k-1)+\Delta R_c\},\quad (32)$$

$$R(u,\ 0)=0,\ R(0,\ k)=0,\quad (33)$$

where ΔR_c is the largest importance gain that community c, Com_c, can bring, and $R(u-1,\ k)$ denotes the importance gain of selecting the kth seed node from previous $u-1$ communities. As shown in Eq. (32), CGA will attempt to detect the kth spreading influence node in community u if the importance gain of finding the kth seed node in community u is larger than in previous $u-1$ communities. Although this strategy has relatively low complexity, it cannot guarantee high accuracy. On the basis of spreading dynamics, Shang et al. (2017) divided the whole process into two phases: 1) The set of seed nodes S influences their nearest neighbors in $N(S)$; and 2) $N(S)$ influences nonseed nodes in each community. In the first phase, the probability of nodes in $N(S)$ being influenced by seeds is defined as

$$P_i(S)=1-\prod_{j\in\Gamma_i\cap S}(1-p_{ji}),\quad (34)$$

where p_{ji} denotes the probability of node j being influenced by seed node i. In the second phase, the influence of $N(S)$ is calculated as follows

$$f(S)=\sum_{Com_i\in Com} f(S,\ S',\ Com_i),\quad (35)$$

$$f(S,\ S',\ Com_i)=\sum_{v\in Com_i} P_v(S,\ S',\ Com_i),\quad (36)$$

where $P_v(S,\ S',\ Com_i)$ represents the probability of node v in community i being influenced. The spreading influence of the set of seeds S is defined on the basis of the weighted cascading model (Guo et al., 2020)

$$g(S)=|N(S)|+a|NC(S)|,\quad (37)$$

where $NC(S)$ denotes the set of neighbors of S, and a is a free parameter. Determining a reasonable probability of nonseed nodes being influenced by $N(S)$ to ensure a high accuracy will be time consuming.

The CSB algorithms based on the spreading mechanism can identify seed nodes more comprehensively. The main limitation of this type of algorithm is similar to the diffusion model-based algorithms mentioned in the MSB algorithm section. Specifically, the performance of this type of algorithms might be influenced once the diffusion mechanism changes.

Intuitively, communities' contributions to information diffusion are different. For example, compared with an independent small-scale community, a large-scale community with high inner edge density and strong relations with other communities is more likely to spread information on a large scale. On the basis of this intuition, Chen et al. (2014) used the community size as a criterion to first filter a set of significant communities and then identify seed nodes in each significant community by comprehensively considering the degree and the similarity of neighbor nodes, and whether the node is a hub. Although this strategy can achieve high accuracy while keeping low complexity, the information provided by the community size is limited and fails to reflect the relations between nodes in different communities. Wang et al. (2018) suggested that the spreading influence of a community can be measured from two aspects, which are its internal and external spreading influence, and the proposed community-hole index (CHR) considers the community spreading influence and the node's position. Specifically, the spreading influence of community i, that is, C_{r_i}, is defined as

$$C_{r_i}=a\times I_i^{\text{out}}+b\times I_i^{\text{in}},\quad (38)$$

where a and b denote two free parameters. I_i^{in} and I_i^{out} represent the edge density of the community containing i and the influence of the community structure on community importance, respectively. The position of a node is quantified by using the following equation

$$B_{r_i}=\frac{\sum_{j\in\Gamma_i^{\text{out}}} e_{ij}C_{r_j}}{\sum_{t\in\Gamma_i^{\text{in}}} e_{it}},\quad (39)$$

where Γ_i^{in} and Γ_i^{out} denote the set of neighbors that belongs and does not belong to the community containing node i, respectively. The final spreading influence of a node is calculated by combining the community-level spreading influence and its position, which is given as

$$CHR(i)=C_{r_i}\times B_{r_i}.\quad (40)$$

From the viewpoint of the seed node selection process, Qiu et al. (2019) divided the process into three steps: 1) Find candidates on the basis of the number of communities the node is connected to and its degree; 2) Further filter candidates by considering the number and the average size of communities the node is connected to, and the node's degree; and 3) Select seed nodes from candidates based on the greedy algorithm and the IC model. Specifically, in the first phase, the core node set S_{core} and the periphery node set S_{boundary} are identified in each community independently. In the second phase, candidates will be selected in accordance with the following equation

$$CI(i)=\begin{cases} k(i)+N_{Com_i}+\text{avg}N_{Com_i}/3, & i\in S_{\text{boundary}} \\ k(i)+N_{Com_i}/2, & i\in S_{\text{core}} \end{cases},\quad (41)$$

where $CI(i)$ is the influence of community i, and $\text{avg}N_{Com_i}$ denotes the average size of communities node i is connected to. After the searching space is narrowed,

the greedy algorithm will identify seed nodes from the set of candidates.

The CSB algorithms have shown their potential in increasing the accuracy of node ranking and accelerating the speed of seed node selection. The advantages and disadvantages of representative CSB algorithms and the main method streams mentioned in this section are summarized in Tables 4 and 5. However, several challenges still need to be addressed.

For the node ranking problem, the community-structural-attribute-based algorithms directly exploit the community-level information when identifying spreading influence nodes. However, the more consideration of the community-level attributes, the better the performance of the algorithms is not always the case because the efficiency of the identification algorithms is important. Thus, how to balance the accuracy and efficiency of the community-structural based algorithm needs to be further explored.

The CSB node ranking and influence maximization algorithms do rely on community detection algorithms because the community structures of most real-world networks are unknown. The community structure identified by different community detection algorithms may be different even for the same network. Thus, how to reduce the dependence of CSB algorithms on community structure detection algorithms is still a challenge in the future. The community structural attributes will be considered when designing the CSB algorithms. However, few studies focus on the relationship between node spreading influence and community structural attributes. Most real-world networks display an overlapping community structure, but most existing CSB algorithms are developed on the basis of the nonoverlapping community structure. Therefore, another challenge is extending these algorithms' application to networks with the overlapping community structure.

Table 4 Advantages and disadvantages of representative CSB algorithms, where K is the number of seed nodes, M is the total number of communities, T_p is the time to compute the degree of a node in community p, N_{Com_p} is the size of community p, and n' and m' are the number of candidate nodes and edges, respectively

Methods	Advantages	Disadvantages	Computational complexity
Vc (Zhao et al., 2014b)	Identifies spreading influence nodes that cannot be detected by a single centrality	The performance will be unstable when the community structure of the network changes	$O(n)$
CbC (Zhao et al., 2015)	Alleviates the instability of Vc by considering the community size and the distribution of neighbors	Other community structural attributes are not used except the size of the community	$O(n\langle k\rangle)$
CbM (Tulu et al., 2018)	The edge densities within and between communities are considered; CbM outperforms CbC	The computational complexity is higher than CbC	$O(mn\langle k\rangle)$
CGA (Wang et al., 2010)	Reduces the computational complexity of MixGreedy by narrowing the searching space to the community scale	The accuracy is lower than MixGreedy	$O\left(MKT_p + KN_{Com_p}T_p\right)$
Community-based framework for influence maximization (CoFIM) (Shang et al., 2017)	Divides the propagation process into two phases, which is more explainable; The computational complexity is low while keeping high accuracy	Determining an appropriate infection rate and free parameters is time consuming	$O\left(K^2nk_{\max}\right)$
PHG (Qiu et al., 2019)	Reduces the computational complexity by filtering candidates before applying the greedy algorithm	Only considers the degree of nodes when identifying the core node set	$O(n\log n + n + Kn'm')$
ICC (Zhao et al., 2020c)	Improves the performance of CC by adopting community structural attributes	Unsuitable for large-scale networks	–
Community-based influence maximization (CIM) (Chen et al., 2014)	Narrows the seed nodes' searching space by the size of the community	Other community structural attributes are ignored, except the size of the community	–
CHR (Wang et al., 2018)	Considers the influence of community-level importance to node-level importance	Unsuitable for large-scale networks	–

Note: "–" denotes that the time complexity is not provided in the original paper.

Table 5 Advantages and disadvantages of three main CSB method streams

Method streams	Related works	Advantages	Disadvantages
Community-structural-attribute-based	Vc (Zhao et al., 2014b); CbC (Zhao et al., 2015); CbM (Tulu et al., 2018); ICC (Zhao et al., 2020c); OC (Wei et al., 2018); MC (Ghalmane et al., 2019a)	Community-level structural information can help detect spreading influence nodes that centralities may ignore	Its performance relies on the community detection algorithms and the selection of community structural attributes
Diffusion-mechanism-based	CGA (Wang et al., 2010); CoFIM (Shang et al., 2017)	The topological information and diffusion mechanism are considered; Community structure information helps to accelerate the speed of seed node selection	Application scenarios are limited by the diffusion mechanism
Community-importance-based	CIM (Chen et al., 2014); CHR (Wang et al., 2018); PHG (Qiu et al., 2019)	The difference in the importance of communities is considered; Community structure information helps to accelerate the speed of seed node selection	How to quantify the importance of community has not been well studied

5 MASB algorithms

Regarding the high time complexity, the MASB algorithms, identifying the spreading influence nodes based on the network's global structure information, perform well in densely connected networks (Namtirtha et al., 2018). The commonly used MASB centralities include betweenness centrality (BC) (Freeman, 1977), closeness centrality (CC) (Sabidussi, 1966), eigenvector centrality (EC) (Bonacich, 1972), and k-shell decomposition (Kitsak et al., 2010). BC defines nodes serving as the role of the bridge of two disconnected groups as spreading influence nodes, which is given as

$$BC(i) = \sum_{v,u \in V} \frac{\sigma(v, u \mid i)}{\sigma(v, u)}, \tag{42}$$

where $\sigma(v, u)$ denotes the number of shortest paths between node v and node u, and $\sigma(v, u \mid i)$ represents the number of shortest paths between node v and node u that pass node i.

CC defines nodes located in the center of the network as spreading influence nodes. Specifically, the location of a node is quantified by the average distance between it and the rest, which is defined as

$$CC(i) = \frac{n-1}{\sum_{j \neq i} d_{ij}}, \tag{43}$$

where d_{ij} is the length of the shortest path between node i and node j.

EC measures the spreading influence of a node by the spreading influence of its neighbors, which is defined as

$$EC(i) = p \sum_{j=1}^{n} a_{ij} x_j, \tag{44}$$

where $a_{ij} = 1$ means there is an edge between nodes i and j, $a_{ij} = 0$ otherwise. p denotes a proportional parameter, and $\boldsymbol{x} = (x_1, x_2, ..., x_n)$ is an eigenvector where each element corresponds to each node's spreading influence. A convergence state can be achieved by updating $\boldsymbol{x}$ iteratively. PageRank is one of the most well-known algorithms developed on the basis of EC and is designed to rank the searching results returned by the Google search engine (Brin and Page, 1998). The original version of PageRank can only work well in strongly connected networks. A return probability was added to overcome this limitation. However, determining a proper return probability requires many tests, which is less efficient when used in social networks. Lü et al. (2011) proposed a parameter-free version of PageRank called LeaderRank (LR), which can converge faster than PageRank. Specifically, in the initialization phase, LeaderRank will first add a ground node interconnected with all existing nodes to make the network strongly connected. The LR values of all nodes, except the ground node, will be set to 1. After the initialization step, the LR value of each node will be updated iteratively by using the following equation

$$LR_i(t) = \sum_{j=1}^{n+1} a_{ij} \frac{LR_j(t-1)}{k_j^{\text{out}}}. \tag{45}$$

When the steady-state is reached, the LR value of the ground node will be evenly allocated to other nodes, and the final LR value of each node corresponds to its spreading influence. Li et al. (2014) suggested the LR value of the ground node should not be evenly assigned. Nodes with larger in-degree should obtain more LR value from the ground node. For example, in social networks, the larger the in-degree of users is, the more followers they have, which can reflect users' popularity. The improved update rule by considering the in-degree of the node is defined as

$$LR_i(t) = \sum_{j=1}^{n+1} w_{ij} \frac{LR_j(t-1)}{\sum_{l=1}^{n+1} w_{jl}}, \tag{46}$$

where for any node i and the ground node g, $w_{ig} = (k_i^{\text{in}})^a$ and a denotes a free parameter. For any node pair i and j, $w_{ij} = 1$ when $a_{ij} = 1$, $w_{ij} = 0$ otherwise.

The main idea of iteration-based algorithms is to aggregate high-order neighbors' structural information in an iterative manner, which is relatively more effective than algorithms that directly exploit the distance information. However, neighbors' contributions to the spreading influence of the target node are mainly quantified by degree values. Measuring the importance of each neighbor node in a more comprehensive manner might further improve the performance of iterative-based algorithms.

Liu et al. (2016a) presented the dynamic-sensitive (DS) algorithm by simulating the discrete susceptible-infected-recovered (SIR) model to measure the spreading influence of nodes. Specifically, the DS algorithm assumes that the activated node has the probability β to activate inactive nodes, and the activated node has the probability μ to return to the inactive state. The probability that a node will be activated at time t is as follows

$$x(t) - x(t-1) = \beta \boldsymbol{A}(\beta \boldsymbol{A} + (1-\mu)\boldsymbol{I})^{t-1} x(0), \tag{47}$$

where $x(t)$ denotes the cumulative probability of the node being activated from time 1 to time t, and $\boldsymbol{I}$ is the identity matrix. $x(t)$ is further approximated as

$$x(t) = \sum_{r=2}^{t} (x(r) - x(r-1)) + x(1). \tag{48}$$

Let $\boldsymbol{e}_i = (0, ..., 0, 1, 0, ..., 0)^{\mathrm{T}}$ when node i is the initial

activated node, and the spreading power of node i can be calculated by

$$S_i(t)=\sum_{r=0}^{t-1}\beta \boldsymbol{A}\boldsymbol{H}^r\boldsymbol{e}_i, \tag{49}$$

where $\boldsymbol{H}=\beta\boldsymbol{A}+(1-\mu)\boldsymbol{I}$. DS considers the network topology and the spreading dynamics.

As mentioned above, the MSB algorithms give less attention to the macro-level position of the node. However, Kitsak et al. (2010) found that a node's position in the network can reflect its spreading influence more accurately than its degree and proposed k-shell decomposition, which splits nodes into different layers by recursively removing nodes in accordance with their degree. A simple example of finding nodes belonging to shell layer 1 by k-shell decomposition is shown in Fig. 4. Specifically, the whole process is as follows: 1) Removing nodes with degree value 1 in the original network (Fig. 4(a)), and a new network can be obtained (Fig. 4(b)); 2) Continue to remove nodes with degree value 1 (Fig. 4(b)); and 3) Repeat step 2 until all remaining nodes' degree is larger than 1 (Fig. 4(c)).

Although the k-shell decomposition can identify nodes located in the core–shell of networks, the spreading influence of nodes in the same shell cannot be distinguished. In recent years, considerable research efforts have been made to improve this limitation. Specifically, the proposed ideas can be mainly classified into distance-based, degree-based, removing-order-based and edge-diversity-based (Zeng and Zhang, 2013; Ren et al., 2013a; Maji et al., 2020). Liu et al. (2013a) alleviated this issue by considering the length of shortest paths between the target node and core nodes, which is given as

$$\theta(i\,|\,ks)=(ks_{\max}-ks+1)\sum_{j\in J}d_{ij},\ i\in S_{ks}, \tag{50}$$

where J denotes the set of nodes in the core–shell layer and S_{ks} represents the set of nodes in the ksth shell layer. Instead of using the distance, Ma et al. (2014) borrowed the idea of "the rich get richer" and designed an algorithm that combined resource allocation dynamic and k-shell decomposition to measure the spreading influence of nodes. In the initialization phase, every node will be assigned the same resources. Then each node will iteratively allocate its resources to neighbor nodes according to their k-shell values. The number of resources allocated by node j to node i at time $t+1$ is defined as

$$R_{j\to i}(t+1)=\left(\frac{ks(i)}{\sum_{u\in\Gamma_j}ks(u)}a_{ij}\right)I_j(t), \tag{51}$$

where $ks(i)$ denotes the k-shell value of node i and $I_j(t)$ denotes resources that node j has at time t. This iteration process will stop once the resource gain is lower than a given threshold. The following equation calculates the total resource value of node i at time $t+1$

$$I_i(t+1)=\sum_{j\in\Gamma_i}R_{j\to i}(t+1). \tag{52}$$

By modifying the pruning rule of the k-shell decomposition algorithm to be more fine-grained, Liu et al. (2015c) presented an improved k-shell (IKs) algorithm. Specifically, the rule is changed to remove nodes with the smallest degree value in each iteration and assign the IKs value to each node. Based on the IKs algorithm, Zareie et al. (2019) proposed an algorithm that measures the spreading influence of nodes by considering the diversity of its neighbor nodes, and the spreading and the intensity of influence. In order to calculate the diversity of node i's neighbor nodes, the Shannon entropy, $H_1(i)$, is introduced, which is defined as

$$H_1(i)=-\sum_{j\in\Gamma_i}\frac{IKs(j)}{IKs(\Gamma_i)}\times\log\left(\frac{IKs(j)}{IKs(\Gamma_i)}\right), \tag{53}$$

where $IKs(\Gamma_i)$ denotes the sum of IKs values of node i's neighbors. The spreading and intensity of the node are quantified by Jensen-Shannon Divergence (JSD) (Lin, 1991), namely

$$JSD(j\in\Gamma_i)=H\left(\sum_{j\in\Gamma_i}\frac{1}{k(i)}\times X_j\right)-\sum_{j\in\Gamma_i}\frac{1}{k(i)}\times H(X_j), \tag{54}$$

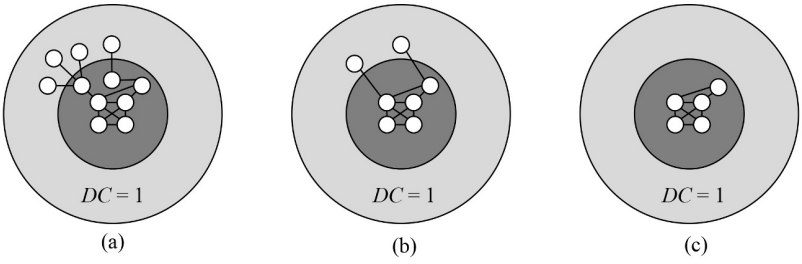

Fig. 4 Diagram of k-shell decomposition: (a) original network; (b) network after the first removal of nodes with degree 1; and (c) network after all nodes with degree 1 have been removed.

where $X_j = (p_{j1}, p_{j2}, ..., p_{jks_{\max}})$ denotes the distribution of IKs values of node j's neighbors. The diversity-strength centrality, $DSC(i)$, of node i is calculated as follows

$$DSC(i) = \overline{IKs(\Gamma_i)} \times H_1(i) \times JSD(j \in \Gamma_i). \quad (55)$$

Although many nodes will be regarded as the same by k-shell decomposition algorithm, these nodes may have different DC values. Bae and Kim (2014) used degree and k-shell values of nodes' neighbors to distinguish differences in the spreading influence of nodes located in the same shell layer and proposed neighborhood coreness (C_{nc}), which is defined as

$$C_{\text{nc}}(i) = \sum_{j \in \Gamma_i} ks(j). \quad (56)$$

Focusing on nodes with the largest k-shell value, Lin et al. (2014) used the sum of neighbors' k-shell value to identify the most influential node of the networks, and proposed improved neighbors' k-core (INK) algorithm, which is given as

$$INK(i) = \sum_{j \in \Gamma_i} ks(j)^a, \quad (57)$$

where a is a tunable parameter.

In addition to the difference in degree value of nodes in the same layer, the order in which nodes located in the same shell are removed is also different. Wang et al. (2016) suggested that iteration factors of the k-shell decomposition can be used to distinguish the spreading influence of nodes in the same shell layer. Compared with these early removed nodes, later removed nodes are closer to the core layer of the network and thus have stronger spreading influence. The mathematical formulation is defined as

$$\delta(i) = ks(i) \times \left(1 + \frac{iter(i)}{iter(\text{total})}\right), \quad (58)$$

where $iter(i)$ denotes the iteration round that node i being removed and $iter(\text{total})$ represents the total number of iteration rounds. Based on this, the influence capability of node i, $InfC(i)$, is defined as

$$InfC(i) = \delta(i)k(i) + \sum_{j \in \Gamma_i} \delta(j)k(j). \quad (59)$$

According to the order in which nodes are removed and nodes' k-shell values, Li et al. (2018) divided neighbors of each node into four categories according to the order in which nodes are removed and nodes' k-shell value: 1) The upper class contains the set of neighbors that have greater k-shell values than the target node; 2) The equal upper class contains neighbors with the same k-shell value of the target node, and the deletion order is the same or later than the target node; 3) The equal lower class represents the set of neighbor nodes that have the same k-shell value as the target node, and the deletion order is the same as the deletion order of the target node or before the target node; and 4) The lower class contains neighbor nodes with smaller k-shell value than the target node. Based on the numbers of different types of neighbors, the spreading capability of node i, $Ks^{\text{CN}}(i)$, is quantified as follows

$$Ks^{\text{CN}}(i) = a_1 \times e^u + a_2 \times e^{eu} + a_3 \times e^{el} + a_4 \times e^l, \quad (60)$$

where a_1, a_2, a_3 and a_4 are free parameters, and e^u, e^{eu}, e^{el}, and e^l denote the number of neighbors belonging to upper, equal upper, equal lower, and lower class, respectively.

Except for the low resolution, the k-shell decomposition algorithm cannot always guarantee a good performance across different types of networks. Liu et al. (2015a) found that nodes located in the core–shell layer can be further divided into true core and core-like groups in accordance with the edge diversity between these core nodes and nodes in other shell layers. Specifically, nodes in the true core group refer to nodes located in the core–shell layer and have a stronger spreading influence than nodes in other shell layers, and the node belongs to a core-like group otherwise. The edge density is used to judge whether core nodes belong to the true core group, which is defined as

$$H_{ks} = -\frac{1}{\ln L} \sum_{ks'=1}^{ks_{\max}} r_{ks,ks'} \ln r_{ks,ks'}, \quad (61)$$

where L denotes the total number of shell layers of the network and $r_{ks,ks'}$ represents the average edge strength of the ksth shell to the ks'th shell. Inspired by this, Liu et al. (2015) presented a strategy to improve the accuracy of k-shell decomposition, that is, removing redundant edges to better identify nodes belonging to the true core group by adding a weight to each edge and setting a threshold value before applying k-shell decomposition. Specifically, the redundant edge refers to the edge that has relatively low spreading influence but may lead to form a core-like group, and the weight of each edge is related to the number of common neighbors between two connected nodes. The fewer the common neighbors are, the larger the weight is. Namtirtha et al. (2021) suggested that MSB and MASB algorithms have their advantages in networks with different connectivity strengths and discovered that k-shell decomposition performs well in networks with strong connectivity and neighbor degree centrality is suitable for sparse networks. On this basis, network global structure-based centrality (NGSC) that combines k-shell decomposition and neighbor degree centrality was proposed, which is defined as

$$NGSC(i) = \sum_{j \in \Gamma_i} (a \times ks(i) + b \times k(j)) + (a \times ks(j) + b \times k(j)), \quad (62)$$

where a and b are two free parameters that can be modified in accordance with the network's connectivity strength to ensure that NGSC can obtain a stable performance in different types of networks. However, tuning parameters is a time-consuming procedure. To solve this issue, Maji (2020) has designed a parameter-free version of NGSC, that is

$$ksd(i)=\sum_{j\in\Gamma_i} ks(i)+ks(j)+\lambda(k(i)+k(j)), \tag{63}$$

where $\lambda=\langle ks\rangle/\langle k\rangle$. Ma et al. (2020) proposed an algorithm that simultaneously measures the spreading influence of the node from the viewpoints of its local and global spreading influence. The entropy of k-shell values is introduced to quantify the global spreading influence of nodes, which is defined as

$$E_i=-\sum_{j=1}^{ks_{\max}} p_i(x_j)\times\log_2 p_i(x_j), \tag{64}$$

$$p_i(x_j)=\frac{|x_j|}{\sum_{j=1}^{ks_{\max}} x_j}, \tag{65}$$

where $x_j=(1, 2, ..., ks_{\max})$ denotes the set of k-shell values of node j's neighbors, and $|x_j|$ represents the number of nodes in the jth shell layer. The local spreading influence of a node is measured on the basis of the assumption that the higher the similarity between the target node and its neighbors is, the higher the belonging of its neighbors to the target node is. The similarity between the target node and its neighbors is defined as

$$B_i=\sum_{j=\Gamma_i} s(i,\ j), \tag{66}$$

$$s(i,\ j)=\frac{2w_{ij}+\sum_{t\in\Gamma_i\cap\Gamma_j} w_{it}w_{jt}}{\sqrt{\left(1+\sum_{t\in\Gamma_i} w_{it}^2\right)\left(1+\sum_{t\in\Gamma_j} w_{jt}^2\right)}}, \tag{67}$$

$$w_{ij}=\frac{|\Gamma_i\cap\Gamma_j|}{|\Gamma_i\cup\Gamma_j|}. \tag{68}$$

On the basis of the global and local importance, the final importance of node i, that is, $Influence(i)$, is calculated as follows

$$Influence(i)=aE_i+bB_i, \tag{69}$$

where $a+b=1$.

Although the combination of micro and macro structural attributes has significantly improved the generalizability of the identification algorithm of spreading influence nodes, the weight assigned to each attribute needs to be defined manually, limiting the introduction of more structural information.

Structural hole theory (Burt et al., 2013) suggested that nodes with strong control over the information flow in the entire network are spreading influence nodes. As shown in Fig. 5, a structural hole node 1 exists in Fig. 5(a) compared with three interconnected nodes in Fig. 5(b). In Fig. 5(a), if node 2 would like to transmit information to node 3 or node 4, then it requires the participation of node 1. If the edge between nodes 2 and 3 is added, the control ability of node 1 will decrease. The network constraint coefficient is used to decide whether the node is a structural hole, which is given as

$$C_i=\sum_{j\in\Gamma_i}\left(p_{ij}+\sum_q p_{iq}p_{qj}\right)^2,\ q\neq i,\ j, \tag{70}$$

$$p_{ij}=\frac{a_{ij}}{\sum_{j\in\Gamma_i} a_{ij}}. \tag{71}$$

However, the network constraint coefficient only considers the relations between the node and its first-order neighbors, which may lead to low resolution. Su and Song (2015) improved the network constraint coefficient by further considering the interactions between second-order neighbors, which is defined as

$$p_{ij}'=\frac{Q(j)}{\sum_{v\in\Gamma_i} Q(v)}, \tag{72}$$

where $Q(j)$ denotes the sum of the degree of j's neighbor nodes. Zhang et al. (2019a) used the improved version of network constraint coefficient (Su and Song, 2015) and network connectivity to measure the local and global importance of the node and proposed CumulativeRank. Specifically, the local spreading influence of node i, denoted by $INCC(i)$, is defined as

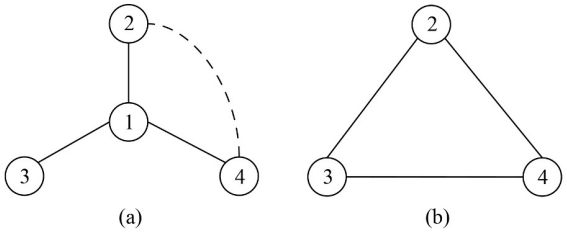

Fig. 5 Diagram of the structural hole (adapted from Su and Song (2015)): (a) network with a structural hole; and (b) network without a structural hole.

$$INCC(i)=\sum_{j\in\Gamma_i}\left(p_{ij}'+\sum_{k=1,k\neq i,j}p_{ik}'p_{kj}'\right). \tag{73}$$

The global importance of CumulativeRank of node i, denoted by R_i, is defined as the influence caused by removing the node on the network structure

$$R_i=\min\left\{\frac{|i|+m(G-i)}{w(G-i)},\ w(G-i)\geqslant 2\right\}, \tag{74}$$

where $|i|$ denotes the cost of removing node i, $m(G-i)$ represents the size of the largest connected component, and $w(G-i)$ is the number of connected components after node i is removed. The final spreading influence of the node i is calculated as follows

$$CumulativeRank(i)=\frac{INCC(i)}{\sqrt{\sum_{j=1}^{n}INCC(j)}}+\frac{TC(i)}{\sqrt{\sum_{j=1}^{n}TC(j)}}, \tag{75}$$

where $TC(i)$ denotes the normalized global importance. Ullah et al. (2021) proposed a global structure model (GSM) by combining the self-importance and global spreading influence of the node, which is defined as

$$GSM(i)=e^{\frac{ks(i)}{n}}\times\sum_{i\neq j}\frac{ks(j)}{d_{ij}}, \tag{76}$$

where the node's self-importance is measured by its k-shell value, and the global importance of the node is quantified by the distances between the node and other nodes in the network and k-shell values. Inspired by the gravity law, Ma et al. (2016) first introduced the gravity formulation to measure the importance of nodes and proposed gravity centrality (GC), which sets the k-shell value as the mass and the length of the shortest path as the distance. The mathematical formulation of GC is defined as

$$GC(i)=\sum_{j\in\Gamma_i}\frac{ks(i)ks(j)}{d_{ij}^2}. \tag{77}$$

Li et al. (2019) thought that nodes with larger degree values are more influential than nodes with smaller degree values. The gravity model (GM) was proposed by considering the distance between a node and all other nodes and replacing the k-shell values as degree values, which is given as

$$GM(i)=\sum_{i\neq j}\frac{k(i)k(j)}{d_{ij}^2}. \tag{78}$$

The GM algorithm considers all the paths between the target node and other nodes, thereby resulting in high computational complexity. To this end, Li et al. (2019) designed a local version of GM algorithm (LGM) by only considering the interactions between nodes within a local area, which is defined as

$$LGM(i)=\sum_{d_{ij}\leqslant R,j\neq i}\frac{k(i)k(j)}{d_{ij}^2}, \tag{79}$$

where R is the truncation radius. Yan et al. (2020) considered multilevel structural attributes when evaluating the spreading influence of nodes. On the basis of GC, the mass is defined as the weighted sum of different centralities, where weights are calculated by entropy technology. In the real-world diffusion process, the distance from node i to node j is not necessarily the same as the distance from node j to node i. However, the Euclidean distance used in the GC assumes that these distances are equal, thereby leading to the dynamic information being ignored. Shang et al. (2021) introduced the effective distance (Brockmann and Helbing, 2013) to capture the hidden dynamic information rather than modifying the mass part of the GC, which is given as

$$C_{\mathrm{EffG}}(i)=\sum_{j=1,j\neq i}^{n}\frac{k(i)\times k(j)}{D_{j|i}^2}, \tag{80}$$

where $D_{j|i}$ denotes the effective distance from node i to node j, which is defined as

$$D_{j|i}=1-\log_2 P_{j|i}, \tag{81}$$

$$P_{j|i}=\frac{a_{ij}}{k(i)},\ i\neq j. \tag{82}$$

If multiple paths are found between nodes i and j, the shortest path will be used.

The MASB algorithms can accurately identify spreading influence nodes in densely connected networks because they focus on the macro-level structural attributes of networks, such as the position and the distance. However, in the era of the explosive growth of information and data, researchers are often faced with large-scale networks, making the use of MASB algorithms to be more limited. Specifically, the challenges of the MASB algorithms are as follows.

For the node ranking problem, the k-shell-based algorithms pay more attention to the difference in the spreading influence of nodes located in the same shell. Studies have shown that a group of core-like nodes exists, which have a high coreness, but they are not the spreading influence nodes. Identifying these core-like nodes can help filter the spreading influence nodes more accurately. The GM has received increasing attention from researchers in the identification of spreading influence nodes due to its great potential in measuring the spreading influence of nodes. However, the distance calculation is time consuming. Thus, improving the effectiveness of gravity-model-based algorithms is still a challenge.

Researchers have developed spreading influence identification algorithms with high generalizability by considering the micro and macro structural attributes simultaneously. However, the weights assigned to each attribute need to be predefined when using this type of algorithms, which is time consuming. How to solve this limitation is also important in the future.

The advantages and disadvantages of representative MASB algorithms and the main method streams mentioned in this section are presented in Tables 6 and 7.

6 MLB algorithms

Driven by growing demands of using graph form datasets to solve real-world problems (Peng et al., 2019) and the increasing trend of interdisciplinary cooperation, the integration of machine learning and network science has received increasing attention from researchers in the two fields (Belkin and Niyogi, 2003). Studies of network science have built a solid foundation for researchers to better use graph form datasets (Peng et al., 2019). Machine learning models can dig out more network topological information, thereby providing strong support for studies of network science (Silva and Zhao, 2012). In recent years, the MLB algorithms have begun to appear in many branches of network science, including node classification (Hall et al., 2009), link prediction (Zhang and Chen, 2018; Chen et al., 2021a), and network statistical feature extraction (Sacchet et al., 2014). As one of the core issues in the study of social networks, machine learning models have been introduced in the research on the identification of spreading influence nodes. A number of the MLB algorithms that can achieve promising performance have been developed in recent years. The MLB algorithms can be mainly divided into two categories: The statistical machine learning-based (SMLB) algorithms and deep learning-based (DLB) algorithms.

Table 6 Advantages and disadvantages of representative MASB algorithms

Methods	Advantages	Disadvantages	Computational complexity
BC (Freeman, 1977)	Identifies nodes that have strong control over the spreading process	High computational complexity; Unsuitable for large-scale networks	$O(n^3)$
CC (Sabidussi, 1966)	Uses the distances between nodes to measure the spreading of nodes	High computational complexity; Unsuitable for large-scale networks	$O(n^3)$
PageRank (Brin and Page, 1998)	Aggregates the structure information of neighbor nodes iteratively; Low computational complexity	Difficult to converge when there are nodes with an out-degree 0	$O(m)$
k-shell (Kitsak et al., 2010)	Considers the position of the node on the basis of the node's degree	The spreading influence of nodes in the same shell layer cannot be distinguished	$O(n)$
LeaderRank (Lü et al., 2011)	Ensures a faster convergence speed than PageRank by adding a ground node	Only suitable for directed networks	$O(m)$
Local and global node influence (LGI) (Ma et al., 2020)	Introduces the entropy to distinguish differences in the spreading influence of nodes in the same shell layer	Tuning free parameters is time consuming	$O(n^2+m)$
CumulativeRank (Zhang et al., 2019a)	The local and global spreading influence of nodes is considered simultaneously	Unsuitable for random networks	$O(n^2+n\langle k\rangle^2)$
GSM (Ullah et al., 2021)	Considers the self-importance of the node and the relationship of the node with other nodes simultaneously	High computational complexity; Unsuitable for large-scale networks	$O(n^2)$
IKs (Liu et al., 2015b)	The nodes are divided into different shell layers in a more granular manner	Unsuitable for random networks	$O(n)$

Table 7 Advantages and disadvantages of four main MASB method streams

Method streams	Related works	Advantages	Disadvantages
Iteration-based	EC (Bonacich, 1972); PageRank (Brin and Page, 1998); LeaderRank (Lü et al., 2011); WleaderRank (Li et al., 2014)	Aggregates the structural information of neighbor nodes in an iterative manner, which is more efficient than directly exploiting the structural information of the entire network	The contribution of neighbor nodes to the spreading influence of the target node is mainly measured by the degree information
k-shell-based	θ (Liu et al., 2013a); Resource Allocation Dynamics (Ma et al., 2014); IKs (Liu et al., 2015b); DSC (Zareie et al., 2019); C_{nc} (Bae and Kim, 2014); Link entropy (Liu et al., 2015a); Influence capacity (Wang et al., 2016); Classified neighbors (CN) (Li et al., 2018)	The position information of nodes in the network is considered, which can avoid to identify nodes located in peripheral regions of the network as spreading influence nodes	The nodes located in the core–shell layer are not always the spreading influence nodes
Micro-macro-based	NGSC (Namtirtha et al., 2021); *ksd* (Maji, 2020); Influence (Ma et al., 2020)	Guarantees stable performance in both sparsely and densely connected networks	Weights assigned to each attribute need to be predefined, which is time consuming
Gravity-model-based	GC (Ma et al., 2016); GM (Li et al., 2019); Yan et al. (2020); C_{EffG} (Shang et al., 2021); Effective distance gravity model (EDGM) (Chen et al., 2021b)	Considers the position of nodes and the effect of the distance between nodes on their interaction	The calculation of distance results in high computational complexity

Unlike the three different types of algorithms mentioned above, the MLB algorithm aims to train a model by a given dataset so that it can be used to predict the spreading influence of nodes or to judge whether a node is important in unseen networks. The statistical machine learning models, such as the support vector machine (SVM), decision tree, linear regression, and logistic regression, can consider multiple structural attributes at once when identifying spreading influence nodes. However, the models' accuracy will not be necessarily improved as more attributes are used. Therefore, feature selection is a key procedure to ensure stable performance of the SMLB algorithms. Bucur (2020) found that combining two complementary centralities to identify spreading influence nodes can achieve higher accuracy than using a single centrality. Hu et al. (2019) used principal component analysis (Moore, 1981) to test different centralities' contributions, including DC, BC, CC, clustering coefficient, HITS value, Laplacian centrality (Qi et al., 2013), and network constraint coefficient, to the identification accuracy of spreading influence nodes and discovered that the Laplacian centrality and network constraint coefficient are important in all seven different types of networks. Han et al. (2015) set the network constraint coefficient, BC, hierarchy, efficient, the network size, PageRank, and clustering coefficient as the input of the ListNet algorithm (Cao et al., 2007) to evaluate the spreading influence of nodes. Zhao et al. (2020a) transformed the identification task of spreading influence nodes into a classification problem rather than viewing it as a regression problem and used nine centralities and the infection rate β of the SIR model to train classification models, including random forest, SVM, and Naive Bayes. The simulation result of the SIR model is numerical, which is unsuitable to be directly used as classification labels. Therefore, labels were obtained by using the following equation

$$label_i = \frac{Scale_i - \min Scale}{range} + 1, \tag{83}$$

where $Scale_i$ denotes the infected scale of node i acquired by simulating the SIR model, $\min Scale$ represents the minimum spreading scale among all nodes, and $range = (\max Scale - \min Scale)/N$, where N is the number of labels. Ivanov et al. (2018) directly labeled the nodes as the seed node and the nonseed node, and proposed a framework that can find other seed nodes based on given seed nodes. The framework contains the following steps: 1) Map nodes into low-dimensional vectors on the basis of network embedding algorithms; 2) Train the classification model on the basis of positive and negative samples; and 3) Use the trained classification model to find l nodes with the highest probability of being the positive sample to complete the seed set. Specifically, positive samples refer to given seed nodes, and negative samples are selected from nonseed nodes in accordance with their degree. The smaller the degree of the node is, the higher the probability that the node is a negative sample.

Intuitively, information can be spread more easily and efficiently between densely connected nodes (Freeman et al., 1991). To measure the connectivity of the local area that a node is located rather than using the length of the shortest path between nodes, Yang and Xiong (2021) used Euclidean distance between nodes by introducing DeepWalk, a network embedding algorithm, to map nodes into low-dimensional vectors. The spreading influence of node i, $NCL(i)$, is defined as

$$NCL(i) = \sum_{j \in \Gamma_i} ks(i) \times e^{-|x_i - x_j|^2}, \tag{84}$$

where x_i denotes the low-dimensional representation of node i.

To enhance the robustness of interference-sensitive algorithms, such as k-shell decomposition, Tixier et al. (2019) was inspired by the idea of ensemble learning and presented a strategy that will first generate multiple perturbed networks on the basis of the original network, then calculate the spreading influence of nodes in all perturbed networks and the original network independently, and rank nodes by the average spreading influence of nodes in all networks. This strategy has effectively increased the robustness of methods such as PageRank and k-shell, while keeping low computational complexity.

In addition to network topology attributes, the spreading influence of users in online social platforms can be described by nontopological features, such as their occupation, age, and content of their posts. Nargundkar and Rao (2016) measured the spreading influence of Twitter users by feeding the linear regression model with nontopological features, such as the number of posts and reposts of Twitter users.

As mentioned above, feature engineering is required before training SMLB algorithms, which is extremely time consuming. With the rapid development of deep learning, the end-to-end deep learning model has become a preferred tool when dealing with the identification task of spreading influence node because DLB algorithms can finish feature selection automatically. The graph form data are not Euclidean data such as images and audios, to which deep learning models, such as the convolutional neural network (CNN) and recurrent neural network (RNN), can be directly applied. The graph neural networks (GNNs), which attempt to extend the use of CNN and RNN on graph form data, were proposed to address this challenge. Specifically, the GNNs can be further classified as the recurrent GNNs, the convolutional GNNs (ConvGNNs), and the spatial–temporal GNNs (Wu et al., 2021). In the identification of spreading influence nodes, the ConvGNNs are widely used owing to their promising performance in processing graph form data. From the perspective of node embedding,

ConvGNNs can be divided into transductive learning-based and inductive learning-based. The transductive learning-based model learns node embeddings for a given network, which must be retrained once the network structure changes. The inductive learning-based model can learn embeddings of unseen nodes after the training step. Niepert et al. (2016) proposed PATCHY-SAN, a transductive learning-based algorithm that extends the use of the CNN to complex networks. The graph convolutional networking (GCN) (Kipf and Welling, 2016) uses the normalized Laplacian matrix of the network as the parameter to aggregate the information of neighbors to learn the low-dimensional representation of the node. Different from the abovementioned transductive learning-based methods, GraphSAGE (Hamilton et al., 2017) is an inductive learning-based method that aims to learn an aggregator that can aggregate the structure information of neighbor nodes so that it can generate node embeddings for unseen networks. Among the ConvGNNs, the GCN algorithm has been used by many GNN-based identification algorithms owing to its simplicity and effectiveness. Wang et al. (2019) designed a DLB algorithm called influence deep learning (IDL) for social networks, which considers the topological and the action logs of the social network users when evaluating their spreading influence. Specifically, IDL samples a fixed-size subnetwork for each node based on the action logs of users by using the random walk method. The pretrained network embedding method is then used to obtain the low-dimensional representation of each node and feeds these vectors into GCN to generate the trained embeddings for each node for predicting the spreading influence of nodes. The instance normalization technique is adopted to make the algorithm focus on the relative position of the node rather than the absolute position. The normalized representation of node i, y_i, is defined as

$$y_i = \frac{x_i - \mu}{\sqrt{\sigma^2 - \varepsilon}}, \tag{85}$$

where $x_i \in R^d$ denotes the low-dimensional vector of user i, μ and σ represent the mean and variance of user representation vectors, respectively, and ε is a small number for numerical stability. The framework of IDL is shown in Fig. 6.

Zhao et al. (2020b) generated the neighborhood network for each node by using the breadth-first-search (BFS) algorithm rather than using random walk. The normalized Laplacian matrix of the neighborhood network, DC, BC, CC, and clustering coefficient are the input of GCN for learning the embeddings of nodes. The output of GCN is used as the input of a fully connected neural network to predict the spreading influence of nodes. The discrimination capabilities of labels under different infection rates β are tested by using Eq. (86) to generate high-quality labels based on the SIR model, because the chosen β has a significant influence on evaluating the spreading influence of nodes.

$$D = \frac{XH - XL}{p(H - L)}, \tag{86}$$

where XH and XL represent the influence of the high influence and low influence groups, respectively, H and L denote the largest and smallest influence, respectively, and p is the percentage of the high influence groups. Yu et al. (2020) used the BFS algorithm to extract the fixed-size neighborhood network for each node, re-encoded the nodes of the neighborhood network in the order in which they are selected, and transformed the adjacency matrix of the neighborhood network in accordance with Eq. (87) to generate the input of each node.

$$B_u = \begin{cases} a_{0j}k(j), & i = 0,\ j = 1,\ 2,\ ...,\ L-1 \\ a_{i0}k(i), & i = 1,\ 2,\ ...,\ L-1,\ j = 0 \\ k(i), & i = j = 0,\ 1,\ 2,\ ...,\ L-1 \\ a_{ij}, & \text{other case} \end{cases}, \tag{87}$$

where L denotes number of nodes in each neighborhood network. When the input matrix of each node is obtained, these transformed matrices will be fed into a CNN model (RCNN) to train a spreading influence prediction model. The framework of RCNN is shown in Fig. 7. Although RCNN algorithm is applicable to large-scale networks,

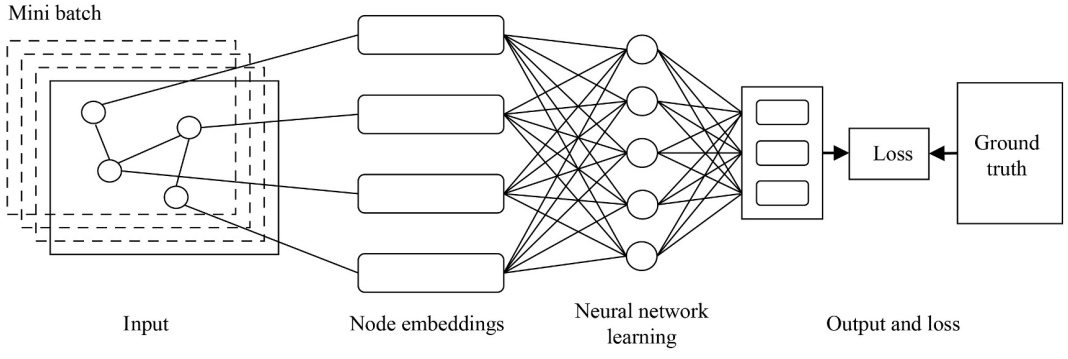

Fig. 6 Framework of IDL (adapted from Wang et al. (2019)).

it uses only the degree information. Ou et al. (2022) improved the performance of RCNN by constructing a three-channel input for each node that preserves the micro, community and macro attributes.

The architecture of the ConvGNN-based algorithms can be simplified as Fig. 8. Specifically, the details of the architecture are as follows: 1) Extract the neighborhood network for each node from the training network by using strategies, such as the random walk or BFS; 2) Construct the input vector for each node based on its topological attributes or nontopological attributes; 3) Obtain the label of each node (if the ground truth labels are available, this step is not required; otherwise, the labels can be obtained by using diffusion models); 4) Generate node embeddings via GCN layers; 5) Predict the spreading influence of each node by using fully-connected networks; 6) Calculate the loss by comparing the predictions with the labels; 7) Optimize the parameters of the models based on the loss of the model; and 8) Test the performance of the trained model on other networks. The ConvGNN-based algorithms have shown their great potential for measuring the spreading influence of nodes. However, existing algorithms give less attention in balancing their efficiency and accuracy. The performance of the ConvGNN-based algorithms, such as RCNN, will be influenced by the structural difference in training and test networks.

Reinforcement learning (RL) has attracted increasing attention. Fan et al. (2020) adopted the RL technique to evaluate spreading influence. The objective function considering the network connectivity is defined as

$$Rc(v_1, v_2, ..., v_n) = \frac{1}{n}\sum_{k=1}^{n}\frac{\sigma(G/\{v_1, v_2, ..., v_k\})}{\sigma(G)}, \quad (88)$$

where $\sigma(G)$ denotes the connectivity of the network G. Equation (88) measures the change in network connectivity after removing selected nodes. The smaller the Rc value is, the more important the selected node is. The basic idea of this algorithm is to transform the node importance identification into a Markov decision process, which lets the agent interact with the environment based on a series of states and actions for maximizing rewards. In the identification task of spreading influence nodes, the environment is the target network, the state is the residual network, the actions are activating or removing the selected node, and the reward is the decrease in Rc. The

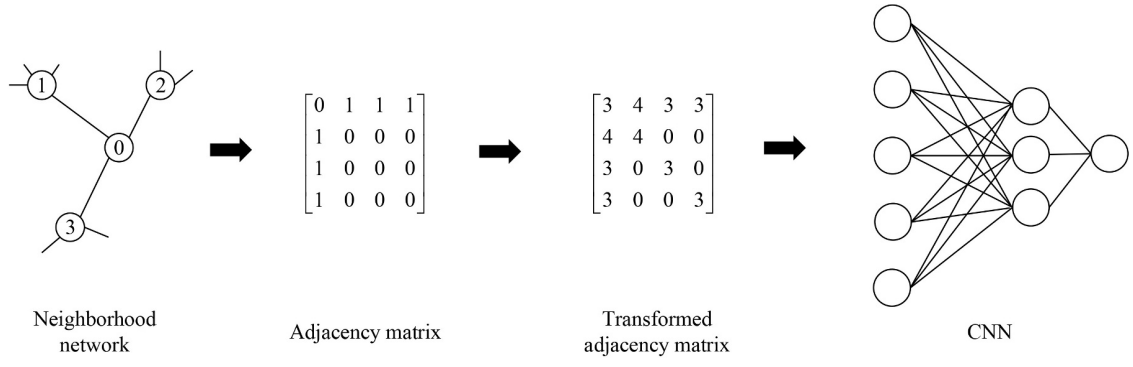

Fig. 7 Framework of RCNN.

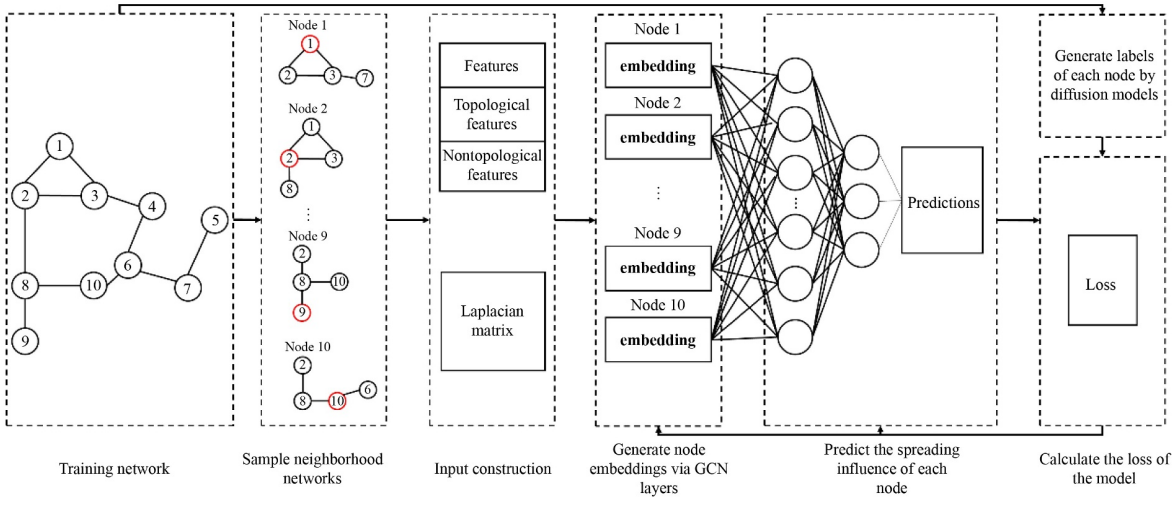

Fig. 8 Simplified architecture of the ConvGNN-based algorithms.

network embedding algorithm was introduced to solve state and action representation problems. The experimental result shows that this algorithm can achieve high accuracy in real-world networks after training on 200000 small-scale artificial random networks.

The advantage of machine learning models in processing multiple features enables the MLB algorithms to achieve a high identification accuracy. However, the following challenges still need to be addressed in the future.

First, the spreading influence of nodes in social networks depends on the topological and nontopological attributes of nodes. However, most existing MLB algorithms only consider the topological attributes of nodes. Therefore, introducing the nontopological information when identifying the spreading influence of nodes via MLB algorithms can be a future development direction of the MLB algorithms.

Second, most MLB algorithms are supervise learning-based, which requires labels of nodes. Diffusion models, such as SIR (Hethcote, 2000), IC (Kempe et al., 2003), and linear threshold (LT) (Kempe et al., 2003), are commonly used to generate the label. However, this strategy will cost considerable time and computational resources when dealing with large-scale networks.

Third, the performance of the MLB algorithms will be unstable when a significant difference is found between the structure of the training network and the test network. How to use small-scale networks to train such models that can achieve strong generalization capabilities is a key issue that needs to be solved in the future.

Finally, the SMLB and DLB algorithms attempt to enhance the identification accuracy of spreading influence nodes by using a set of structural attributes, such as DC, BC, and CC. However, the computational complexity of calculating multiple structural attributes is high, limiting the use of the SMLB and DLB algorithms in large-scale networks. Thus, more attention should be paid to balancing the accuracy and efficiency when designing MLB algorithms. The advantages and disadvantages of representative MLB algorithms and the two main method streams of MLB algorithms introduced in this section are listed in Tables 8 and 9.

7 Diffusion models

We do not know which nodes are important in social networks, otherwise, we do not need to identify the spreading influence of nodes. Therefore, diffusion models are used to generate approximations of nodes' real spreading influence and to test the performance of algorithms. Given that diffusion models are designed on the basis of different spreading mechanisms (Cohen, 1992; Fu et al., 2020), choosing an appropriate diffusion

Table 8 Advantages and disadvantages of representative MLB algorithms

Methods	Advantages	Disadvantages	Machine learning models
InfEmb (Ivanov et al., 2018)	Determines other seed nodes based on the structure information of the given seed nodes	Unable to identify the remaining seed nodes with few positive samples accurately; Only the degree of the node is considered when selecting negative samples	Linear regression
InfluenceRank (Nargundkar and Rao, 2016)	Considers the topological and nontopological features simultaneously	Cannot ensure that the selected features accurately reflect the node's spreading influence	DeepWalk; SVM
P&C (Tixier et al., 2019)	Enhances the robustness of algorithms with weak anti-interference ability	Relies on the network perturbation method	Bagging
IDL (Wang et al., 2019)	Considers the action logs of online users and network topology	Only suitable for social networks	GCN
InfGCN (Zhao et al., 2020b)	Uses the low-dimensional vector representation and centralities of nodes as the input to predict the spreading influence of nodes	High computational complexity	GCN
RCNN (Yu et al., 2020)	Only uses the degree of the node to transform the matrix representation of the node, which is more efficient than InfGCN and IDL	The performance is unstable when the structure of the training network and the test network are different	CNN
FINDER (Fan et al., 2020)	Applies the reinforcement learning technique to evaluate the spreading influence of nodes	High computational complexity	RL; DeepWalk

Table 9 Advantages and disadvantages of two main MLB method streams

Method streams	Related works	Advantages	Disadvantages
SMLB	Multicentrality predictors (Bucur, 2020); Hu et al. (2019); Han et al. (2015); Zhao et al. (2020a); InfEmb (Ivanov et al., 2018); NCL (Yang and Xiong, 2021); InfluenceRank (Nargundkar and Rao, 2016); P&C (Tixier et al., 2019)	Weights assigned to each structural attribute can be obtained automatically by training the model; Highly interpretable	Requires the time-consuming feature engineering process; The accuracy of the model is unstable when the network structure of the training set and the test set are different
DLB	IDL (Wang et al., 2019); InfGCN (Zhao et al., 2020b); RCNN (Yu et al., 2020); FINDER (Fan et al., 2020)	Feature engineering is not required; Learning from streaming data	Easy to overfit in the training process; The training process is a black box; The accuracy of the model is unstable when the network structure of the training set and the test set are different

model that can best describe the spreading process of the target task is important to ensure that test results are credible. In this section, we introduce diffusion models that are widely used in studies of identification of spreading influence nodes, such as the susceptible-infected (SI) model (Barabási and Albert, 1999), the susceptible-infected-susceptible (SIS) model (Cohen, 1992), SIR model (Hethcote, 2000), LT model (Kempe et al., 2003), and IC model (Kempe et al., 2003).

7.1 SI model

As one of the classic epidemic models, the SI model sets each node in one of the two states: The susceptible state S and the infected state I. In the initialization phase, seed nodes will be set as infected. When the diffusion starts, the susceptible nodes will be infected by their infected neighbors with an infection rate β. The propagation process stops until no newly infected node is found, and the number of total infected nodes is the overall influence of seed nodes.

7.2 SIS model

The SIS model considers the possibility of infected nodes becoming susceptible nodes again based on the SI model. The SIS model sets each node to be in one of the two states: The susceptible state S and the infected state I. The only difference between the SIS and SI models is that the infected nodes may become susceptible nodes again.

7.3 SIR model

On the basis of the SI model, the SIR model adds the recovered state. Specifically, the SIR model sets each node to be in one of three states, which are the susceptible state S, the infected state I, and the recovered state R. After the diffusion process starts, susceptible nodes will be infected by their infected neighbor nodes with the infection rate β, and infected nodes will recover with probability r.

7.4 LT model

The LT model was designed to simulate the diffusion process in directed networks. Specifically, each edge of the directed network will be assigned a weight. For example, $w(u, v) = 1/k(v)$ is the weight of the edge pointing from node u to node v, which represents the spreading influence of node u to node v among all node v's in-neighbors. In the initialization phase, seed nodes will be set to either activated nodes or inactivated nodes. When the diffusion process starts, the inactivated node will be activated if the sum of weights of edges between the node and all its activated in-neighbors is greater than a given threshold γ. The diffusion process stops until no newly activated node is found.

7.5 IC model

The IC model was also designed for directed networks. In the initialization phase, seed nodes will be set to either activated or inactivated nodes. After the propagation starts, inactivated nodes will be activated by their activated in-neighbors with probability β. If an inactive node has multiple activated in-neighbor nodes, these in-neighbor nodes will independently try to activate the node in random order. The propagation process stops when no new activated node is found.

7.6 Weighted IC model

The weighted IC model (Palla et al., 2005) is a weighted version of the IC model. During the diffusion process, assuming the inactive node v is the out-neighbor of the activated node u at time t, node v will be activated by node u with the probability of $1/k(v)$ at time $t+1$. If the inactive node v has n active in-neighbors at time t, the node v will be activated at time $t+1$ with the probability of $1-(1-1/k(v))^t$.

7.7 Conformity-aware IC model

Conformity awareness plays a vital role in spreading information, opinion, and beliefs in the real world. For example, online users are more likely to repost the information that most users believe in the social platform. On this basis, Li et al. (2013) proposed the conformity-aware IC model. Specifically, assuming the inactive node v is the out-neighbor of the active node u at time t, node v will be activated by node u with the probability

$$\Pr(v \mid u) = 1 - \prod_{u \in \Gamma_v} (1 - \Phi(u)\Omega(v)), \tag{89}$$

where $\Phi(u)$ denotes the influence of node u, and $\Omega(v)$ represents the conformity of node v.

8 Performance evaluation metrics

Evaluation metrics are needed to compare the ranking results obtained by diffusion models and identification algorithms. This process is required to test the accuracy of the identification algorithms of spreading influence nodes. This section introduces eight widely used evaluation metrics in this research field.

8.1 Average spreading influence

Comparing the average influence of top $p \times n$ ($p \in [0, 1]$) most influential nodes identified by different identification

algorithms (Chen et al., 2013; Berahmand et al., 2018) can measure algorithms' performance to some extent. The average spreading influence of nodes identified by a specific algorithm, $AvgSI$, is defined as

$$AvgSI = \frac{\sum_{v \in S} \sigma(v)}{p \times n}, \tag{90}$$

where S denotes the set of seed nodes identified by using a specific algorithm, n is the total number of nodes, and $\sigma(v)$ represents the spreading influence of node v.

8.2 Influence scale

The influence scale $F(t)$ can reflect the change in influence of nodes identified by a specific algorithm over time, which is defined as

$$F(t) = \frac{N_{I(t)} + N_{R(t)}}{n}, \tag{91}$$

where $N_{I(t)}$ and $N_{R(t)}$ denote the number of infected and recovered nodes at time t, respectively.

8.3 Imprecision function

The imprecision function $\varepsilon(p)$ (Kitsak et al., 2010) is introduced to quantify the difference in the average spreading scales between top $p \times n$ nodes identified by the identification algorithm and the $p \times n$ most efficient spreaders identified by diffusion models ($p \in [0, 1]$). The spreading efficiency M_i is defined as the number of nodes infected by node i, $\delta_{\text{eff}}(p)$ denotes the set of top $p \times n$ nodes selected in accordance with the spreading efficiency, and $\delta_x(p)$ represents the set of top $p \times n$ nodes identified by the identification algorithm x of spreading influence nodes. The imprecision value of x is defined as

$$\varepsilon_x(p) = 1 - \frac{M_x(p)}{M_{\text{eff}}(p)}, \tag{92}$$

where $M_x(p)$ and $M_{\text{eff}}(p)$ denote the average influence of $\delta_x(p)$ and $\delta_{\text{eff}}(p)$, respectively. The closer the imprecision value $\varepsilon_x(p)$ is to 0, the closer the average influence of the node set identified by x is to the average influence of the most influential node set identified by diffusion models.

8.4 Relative difference of spreading scales

The relative difference of spreading scales (Knight, 1966; Zhao et al., 2014a) between two sets of top $p \times n$ most important nodes identified by two different node importance identification algorithms is defined as

$$\Delta_y(p) = \frac{S_y - S_x}{S_x}, \tag{93}$$

where S_y denotes the total influence of the set of seed nodes identified by algorithm y. The total influence of seed nodes identified by algorithm y is greater than that by algorithm x when $\Delta_y(p) > 0$.

8.5 Kendalls' τ correlation coefficient

The Kendalls' τ correlation coefficient (Wang et al., 2016) is used to measure the similarity of two ordered lists and is widely utilized when testing the performance of identification algorithms. Assuming two ordered lists A and B, each of which contains n elements. (A_i, B_i) denotes the ith element pair of A and B. When any two element pairs of A and B have the same ranking, such as $A_i > A_j$ and $B_i > B_j$ or $A_i < A_j$ and $B_i < B_j$, the two element pairs are a concordant pair, otherwise, they are a discordant pair. The Kendalls' τ correlation coefficient is calculated in accordance with the number of concordant pairs and discordant pairs of two ordered lists, which is defined as

$$\tau = \frac{2(C - D)}{k(k-1)}, \tag{94}$$

where C and D denote the number of concordant pairs and discordant pairs, respectively, and k is the total number of elements in each order list. The closer the Kendalls' τ coefficient is to 1, the more similar the two ordered lists are. Another evaluation metric that has a similar function as the Kendalls' τ coefficient is the Jaccard correlation coefficient (Wang et al., 2018), which is given as

$$J_c = \frac{|X(c) \cap Y(c)|}{|X(c) \cup Y(c)|}, \tag{95}$$

where $X(c)$ and $Y(c)$ represent the seed nodes selected by the identification algorithm and the most influential nodes acquired by stimulating the diffusion model, respectively.

8.6 Monotonicity

The monotonicity (Wang et al., 2016) is used to measure the uniqueness of nodes' rank obtained by the identification algorithms of spreading influence nodes, which is given as

$$M(X) = \left(1 - \frac{\sum_{i \in I} n_i(n_i - 1)}{n(n-1)}\right)^2, \tag{96}$$

where n_i denotes the number of nodes assigned to rank i, X represents a spreading influence nodes identification algorithm, I contains all unique values obtained by applying X, and n is the total number of nodes. The value of $M(X)$ is between 0–1. The closer the value is to 1, the fewer nodes are assigned to the same level.

The complementary cumulative distribution function (CCDF) (Li et al., 2018; Zareie et al., 2019) has a similar function as monotonicity, which describes the distribution of nodes in different rankings. The mathematical formulation is defined as

$$CCDF(Z) = \Pr(Z > z) = 1 - CDF(z), \quad (97)$$

where the cumulative distribution function $CDF(z)$ denotes the probability that the node's rank is less than or equal to z.

9 Discussion and conclusions

In this review, we briefly summarized the recent progress of studies on the identification of spreading influence nodes, emphasizing the applications of the identification algorithms of spreading influence nodes in social networks. An increasing number of novel algorithms based on new techniques resulting from the studies of other research fields, especially ideas and tools from community detection and machine learning, have emerged in recent years due to the confluence of improved computational capabilities, the explosive growth of new datasets, increasing trend of interdisciplinary development, and fast-changing demands.

No single algorithm can achieve stable performance in all types of networks (Lü et al., 2016; Bucur, 2020; Namtirtha et al., 2021). An in-depth understanding of social network structural attributes that affect algorithmic performance can be considered the guide for the choice of algorithms when the accuracy and complexity have to be considered. Specifically, the MSB algorithms may be a good choice if the social network is large and sparse because they can achieve relatively high accuracy with extremely low computational complexity. However, if nodes are densely connected and the size of the social network is small, then the MSB algorithms may perform not as well as expected because they mainly focus on the edge density within a local area of nodes. In such cases, we can try to use CSB algorithms or MASB algorithms to utilize community-level and macro-level information. For social networks with a clear modular structure, the CSB algorithms may provide highly accurate identification results and an in-depth understanding of how information is actually spread between nodes belonging to different communities. The spreading influence of nodes in social networks can be determined by topological and nontopological attributes, such as interaction frequency and the number of reposts and comments, making the study of the identification of spreading influence nodes a natural place to apply machine learning models. Compared with the micro-macro-based MASB algorithms, which require weights assigned to each attribute to be predefined, the MLB algorithms are more suitable for identifying spreading influence nodes based on multiple attributes. To sum up, the advantages and disadvantages of MSB, CSB, MASB, and MLB algorithms are summarized in Table 10. Although great advancement has been made in recent years, a number of unsolved problems that can affect the future development on the identification of spreading influence nodes still exist.

First, as discussed previously, the performance of the identification algorithms of spreading influence nodes is highly correlated with the structural attributes of social networks, making the benchmark datasets for comparing the accuracy of different types of algorithms a prerequisite to ensure the credibility of testing results. However, the performance of most existing algorithms is tested on different networks due to the lack of unifying datasets, which may lead to an unwanted outcome that only the good aspects of the algorithm are reported, thereby hindering their application.

Second, another huge challenge is the identification of spreading influence nodes in temporal networks, where connections between nodes will change over time. Although algorithms designed for static networks can achieve relatively high accuracy, the temporal networks are more in line with real-world situations. Therefore, further improving these algorithms so that they can be applied in temporal networks will be worth paying attention to.

Third, the algorithms' performance can be significantly enhanced by considering the structural attributes and the nonstructural attributes of nodes. In social networks, the spreading influence of individuals can be influenced by their occupations, social status, and online reputations.

Table 10 Advantages and disadvantages of MSB, CSB, MASB, and MLB algorithms

Algorithms	Advantages	Disadvantages
MSB	Estimates the spreading influence of nodes via micro-level structural information, thereby enabling the identification of spreading influence nodes in large-scale networks to be feasible	Focuses on the edge density within a local area of the target node, leading to nodes located in peripheral regions of networks to be misclassified as spreading influence nodes
CSB	Accelerates the speed of seed node selection for the influence maximization task; Community structural attributes can help to improve the accuracy of centralities	The performance of CSB algorithms depends on the community detection algorithms
MASB	Helps to rectify the identification results of MSB algorithms; Performs well in densely connected networks	High computational complexity hinders the use of MASB algorithms in large-scale networks; Weights of MASB algorithms that consider multiple structural attributes need to be predefined
MLB	Automatically calculates the weights assigned to different attributes of nodes; Higher accuracy of identification than traditional algorithms	The computational complexity of MLB algorithms did not receive sufficient attention; The performance of MLB algorithms is influenced by the structural difference in the training network and the test network

However, most of the existing algorithms mainly focused on the network's topological attributes and did not use nontopological information. The machine learning model might be an appropriate choice to consider multiple features at once while identifying important social network nodes.

Finally, although MLB algorithms can predict the node's spreading influence by simultaneously considering different features of the node, the learning strategy of these algorithms is supervised learning, which requires labels to be generated by information diffusion models or disease diffusion models. Training these models on large-scale networks with more than millions of nodes is costly due to limited time and resources. Therefore, unsupervised learning-based algorithms may become new hot spot in the near future. The distribution between the training data and test data will affect the performance of machine learning models. However, the influence of the structural differences in training networks and test networks on the performance of MLB algorithms has not been well studied.

References

Albert R, Jeong H, Barabási A L (1999). Diameter of the World-Wide Web. Nature, 401(6749): 130–131

Bae J, Kim S (2014). Identifying and ranking influential spreaders in complex networks by neighborhood coreness. Physica A: Statistical Mechanics and Its Applications, 395: 549–559

Bao Z K, Liu J G, Zhang H F (2017). Identifying multiple influential spreaders by a heuristic clustering algorithm. Physics Letters A, 381(11): 976–983

Barabási A L, Albert R (1999). Emergence of scaling in random networks. Science, 286(5439): 509–512

Barabási A L, Bonabeau E (2003). Scale-free networks. Scientific American, 288(5): 60–69

Belkin M, Niyogi P (2003). Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation, 15(6): 1373–1396

Berahmand K, Bouyer A, Samadi N (2018). A new centrality measure based on the negative and positive effects of clustering coefficient for identifying influential spreaders in complex networks. Chaos, Solitons, and Fractals, 110: 41–54

Bertozzi A L, Franco E, Mohler G, Short M B, Sledge D (2020). The challenges of modeling and forecasting the spread of COVID-19. Proceedings of the National Academy of Sciences of the United States of America, 117(29): 16732–16738

Boguñá M, Pastor-Satorras R, Díaz-Guilera A, Arenas A (2004). Models of social networks based on social distance attachment. Physical Review E: Statistical, Nonlinear, and Soft Matter Physics, 70(5): 056122

Bonacich P (1972). Factoring and weighting approaches to status scores and clique identification. Journal of Mathematical Sociology, 2(1): 113–120

Borge-Holthoefer J, Moreno Y (2012). Absence of influential spreaders in rumor dynamics. Physical Review E: Statistical, Nonlinear, and Soft Matter Physics, 85(2): 026116

Brin S, Page L (1998). The anatomy of a large-scale hypertextual web search engine. In: Proceedings of the 7th International Conference on World Wide Web. Brisbane: Association for Computing Machinery, 107–117

Brockmann D, Helbing D (2013). The hidden geometry of complex, network-driven contagion phenomena. Science, 342(6164): 1337–1342

Bucur D (2020). Top influencers can be identified universally by combining classical centralities. Scientific Reports, 10(1): 20550

Burt R S, Kilduff M, Tasselli S (2013). Social network analysis: Foundations and frontiers on advantage. Annual Review of Psychology, 64(1): 527–547

Buyalskaya A, Gallo M, Camerer C F (2021). The golden age of social science. Proceedings of the National Academy of Sciences of the United States of America, 118(5): e2002923118

Campan A, Cuzzocrea A, Truta T M (2017). Fighting fake news spread in online social networks: Actual trends and future research directions. In: IEEE International Conference on Big Data. Boston, MA, 4453–4457

Cantwell G T, Newman M E J (2019). Mixing patterns and individual differences in networks. Physical Review E: Statistical, Nonlinear, and Soft Matter Physics, 99(4): 042306

Cao Z, Qin T, Liu T Y, Tsai M F, Li H (2007). Learning to rank: From pairwise approach to listwise approach. In: Proceedings of the 24th International Conference on Machine Learning. Corvallis, OR: Association for Computing Machinery, 129–136

Chen D B, Gao H, Lü L, Zhou T (2013). Identifying influential nodes in large-scale directed networks: The role of clustering. PLoS One, 8(10): e77455

Chen D B, Lü L Y, Shang M S, Zhang Y C, Zhou T (2012). Identifying influential nodes in complex networks. Physica A: Statistical Mechanics and Its Applications, 391(4): 1777–1787

Chen D B, Sun H L, Tang Q, Tian S Z, Xie M (2019). Identifying influential spreaders in complex networks by propagation probability dynamics. Chaos, 29(3): 033120

Chen J Y, Zhang J, Xu X H, Fu C B, Zhang D, Zhang Q P, Xuan Q (2021a). E-LSTM-D: A deep learning framework for dynamic network link prediction. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 51(6): 3699–3712

Chen S, Ren Z M, Liu C, Zhang Z K (2020). Identification methods of vital nodes on temporal network. Journal of University of Electronic Science and Technology of China, 49(2): 291–314 (in Chinese)

Chen W, Wang Y J, Yang S Y (2009). Efficient influence maximization in social networks. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Paris, 199–208

Chen Y, Guo Q, Liu M, Liu J G (2021b). Improved gravity model for identifying the influential nodes. Europhysics Letters, 136(6): 68004

Chen Y C, Zhu W Y, Peng W C, Lee W C, Lee S Y (2014). CIM: Community-based influence maximization in social networks. ACM Transactions on Intelligent Systems and Technology, 5(2): 1–31

Cohen J E (1992). Infectious diseases of humans: Dynamics and control. Journal of the American Medical Association, 268(23): 3381

Dai J Y, Wang B, Sheng J F, Sun Z J, Khawaja F R, Ullah A, Dejene D A, Duan G H (2019). Identifying influential nodes in complex networks based on local neighbor contribution. IEEE Access, 7: 131719–131731

Dai L, Guo Q, Liu X L, Liu J G, Zhang Y C (2018). Identifying online user reputation in terms of user preference. Physica A: Statistical Mechanics and Its Applications, 494: 403–409

Dong G, Wang F, Shekhtman L M, Danziger M M, Fan J, Du R, Liu J, Tian L, Stanley H E, Havlin S (2021). Optimal resilience of modular interacting networks. Proceedings of the National Academy of Sciences of the United States of America, 118(22): e1922831118

Dorogovtsev S N, Goltsev A V, Mendes J F F (2008). Critical phenomena in complex networks. Reviews of Modern Physics, 80(4): 1275–1335

Fan C, Zeng L, Sun Y, Liu Y Y (2020). Finding key players in complex networks through deep reinforcement learning. Nature Machine Intelligence, 2(6): 317–324

Freeman L C (1977). A set of measures of centrality based on betweenness. Sociometry, 40(1): 35–41

Freeman L C (1978). Centrality in social networks conceptual clarification. Social Networks, 1(3): 215–239

Freeman L C, Borgatti S P, White D R (1991). Centrality in valued graphs: A measure of betweenness based on network flow. Social Networks, 13(2): 141–154

Fu J Q, Liu M, Deng C Y, Huang J, Jiang M Z, Guo Q, Liu J G (2020). Spreading model of the COVID-19 based on the complex human mobility. Journal of University of Electronic Science and Technology of China, 49(3): 383–391 (in Chinese)

Galstyan A, Cohen P (2007). Cascading dynamics in modular networks. Physical Review E: Statistical, Nonlinear, and Soft Matter Physics, 75(3): 036109

Galvão V, Miranda J G, Andrade R F, Andrade Jr J S, Gallos L K, Makse H A (2010). Modularity map of the network of human cell differentiation. Proceedings of the National Academy of Sciences of the United States of America, 107(13): 5750–5755

Gao S, Ma J, Chen Z M, Wang G H, Xing C M (2014). Ranking the spreading ability of nodes in complex networks based on local structure. Physica A: Statistical Mechanics and Its Applications, 403: 130–147

Ghalmane Z, Cherifi C, Cherifi H, Hassouni M E (2019a). Centrality in complex networks with overlapping community structure. Scientific Reports, 9(1): 10133

Ghalmane Z, El Hassouni M, Cherifi C, Cherifi H (2019b). Centrality in modular networks. EPJ Data Science, 8(1): 15

Girvan M, Newman M E J (2002). Community structure in social and biological networks. Proceedings of the National Academy of Sciences of the United States of America, 99(12): 7821–7826

Guimerà R, Danon L, Díaz-Guilera A, Giralt F, Arenas A (2003). ELF-similar community structure in a network of human interactions. Physical Review E: Statistical, Nonlinear, and Soft Matter Physics, 68(6): 065103

Guo C, Yang L, Chen X, Chen D, Gao H, Ma J (2020). Influential nodes identification in complex networks via information entropy. Entropy, 22(2): 242–260

Guo Q, Yin R R, Liu J G (2019). Node importance identification for temporal networks via the TOPSIS method. Journal of University of Electronic Science and Technology of China, 48(2): 296–300 (in Chinese)

Halappanavar M, Sathanur A V, Nandi A K (2016). Accelerating the mining of influential nodes in complex networks through community detection. In: Proceedings of the ACM International Conference on Computing Frontiers. Como, 64–71

Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten L H (2009). The WEKA data mining software: An update. SIGKDD Explorations, 11(1): 10–18

Hamilton W L, Ying R, Leskovec J (2017). Inductive representation learning on large graphs. In: Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach, CA: Curran Associates Inc., 1025–1035

Han Z M, Wu Y, Tan X S, Duan D G, Yang W J (2015). Ranking key nodes in complex networks by considering structural holes. Acta Physica Sinica, 64(5): 058902

Hethcote H W (2000). The mathematics of infectious diseases. SIAM Review, 42(4): 599–653

Hou L, Liu J G, Pan X, Wang B H (2014). A social force evacuation model with the leadership effect. Physica A: Statistical Mechanics and Its Applications, 400: 93–99

Hu G, Xu X, Zhang W M, Zhou Y (2019). Contribution analysis for assessing node importance indices with principal component analysis. Acta Electronica Sinica, 47(2): 358–365 (in Chinese)

Hu Y, Ji S, Jin Y, Feng L, Stanley H E, Havlin S (2018). Local structure can identify and quantify influential global spreaders in large scale social networks. Proceedings of the National Academy of Sciences of the United States of America, 115(29): 7468–7472

Huang H, Shen H, Meng Z, Chang H, He H (2019). Community-based influence maximization for viral marketing. Applied Intelligence, 49(6): 2137–2150

Ivanov S, Durasov N, Burnaev E (2018). Learning node embeddings for influence set completion. In: IEEE International Conference on Data Mining Workshops. Singapore, 1034–1037

Jeong H, Mason S P, Barabási A L, Oltvai Z N (2001). Lethality and centrality in protein networks. Nature, 411(6833): 41–42

Jia J S, Lu X, Yuan Y, Xu G, Jia J, Christakis N A (2020). Population flow drives spatio–temporal distribution of COVID-19 in China. Nature, 582(7812): 389–394

Kempe D, Kleinberg J, Tardos E (2003). Maximizing the spread of influence through a social network. In: Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. Washington, D.C., 137–146

Kipf T N, Welling M (2016). Semi-supervised classification with graph convolutional networks. arXiv preprints, arXiv:1609.02907

Kitsak M, Gallos L K, Havlin S, Liljeros F, Muchnik L, Stanley H E, Makse H A (2010). Identification of influential spreaders in complex network. Nature Physics, 6(11): 888–893

Kleinberg J M (1999). Authoritative sources in a hyperlinked environment. Journal of the ACM, 46(5): 604–632

Klimt B, Yang Y (2004). The Enron Corpus: A new dataset for email classification research. In: Proceedings of the 15th European Conference on Machine Learning. Berlin: Springer, 217–226

Knight W R (1966). A computer method for calculating Kendall's τ with un-grouped data. Journal of the American Statistical Association, 61(314): 436–439

Kumar A, Snyder M (2002). Protein complexes take the bait. Nature, 415(6868): 123–124

Kumar S, Panda B S (2020). Identifying influential nodes in social networks: Neighborhood coreness based voting approach. Physica A: Statistical Mechanics and Its Applications, 553: 124215

Kunegis J (2016). KONECT: The Koblenz network collection. In: Proceedings of the 22nd International Conference on World Wide Web. Rio de Janeiro: Association for Computing Machinery, 1343–1350

Leskovec J, Kleinberg J, Faloutsos C (2007). Graph evolution: Densification and shrinking diameters. ACM Transactions on Knowledge Discovery from Data, 1(1): 2

Leskovec J, Lang K J, Dasgupta A, Mahoney M W (2009). Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. Internet Mathematics, 6(1): 29–123

Liben-Nowell D L, Kleinberg J (2007). The link-prediction problem for social networks. Journal of the American Society for Information Science and Technology, 58(7): 1019–1031

Li C, Wang L, Sun S W, Xia C Y (2018). Identification of influential spreaders based on classified neighbors in real-world complex networks. Applied Mathematics and Computation, 320: 512–523

Li H, Bhowmick S S, Sun A X (2013). CINEMA: Conformity-aware greedy algorithm for influence maximization in online social networks. In: Proceedings of the 16th International Conference on Extending Database Technology. Genoa: Association for Computing Machinery, 323–334

Li Q, Zhou T, Lü L Y, Chen D B (2014). Identifying influential spreaders by weighted LeaderRank. Physica A: Statistical Mechanics and Its Applications, 404: 47–55

Li Z, Ren T, Ma X, Liu S, Zhang Y, Zhou T (2019). Identifying influential spreaders by gravity model. Scientific Reports, 9(1): 8387

Lin J (1991). Divergence measures based on the Shannon entropy. IEEE Transactions on Information Theory, 37(1): 145–151

Lin J H, Guo Q, Dong W Z, Tang L Y, Liu J G (2014). Identifying node spreading influence with largest *k*-core values. Physics Letters A, 378(45): 3279–3284

Liu J G, Lin J H, Guo Q, Zhou T (2016a). Locating influential nodes via dynamics-sensitive centrality. Scientific Reports, 6(1): 21380

Liu J G, Ren Z M, Guo Q (2013a). Ranking the spreading influence in complex networks. Physica A: Statistical Mechanics and Its Applications, 392(18): 4154–4159

Liu J G, Ren Z M, Guo Q, Wang B H (2013b). Node importance ranking of complex networks. Acta Physica Sinica, 62(17): 178901

Liu J G, Wang Z Y, Guo Q, Guo L, Chen Q, Ni Y Z (2017a). Identifying multiple influential spreaders via local structural similarity. Europhysics Letters, 119(1): 18001

Liu J Q, Li X R, Dong J C (2021). A survey on network node ranking algorithms: Representative methods, extensions, and applications. Science China Technological Sciences, 64(3): 451–461

Liu X L, Liu J G, Yang K, Guo Q, Han J T (2017b). Identifying online user reputation of user-object bipartite networks. Physica A: Statistical Mechanics and Its Applications, 467: 508–516

Liu Y, Tang M, Zhou T, Do Y (2015a). Core-like groups result in invalidation of identifying super-spreader by *k*-shell decomposition. Scientific Reports, 5(1): 9602

Liu Y, Tang M, Zhou T, Do Y (2015b). Improving the accuracy of the *k*-shell method by removing redundant links: From a perspective of spreading dynamics. Scientific Reports, 5(1): 13172

Liu Y, Tang M, Zhou T, Do Y (2016b). Identify influential spreaders in complex networks, the role of neighborhood. Physica A: Statistical Mechanics and Its Applications, 452: 289–298

Liu Z H, Jiang C, Wang J Y, Yu H (2015c). The node importance in actual complex networks based on a multi-attribute ranking method. Knowledge-Based Systems, 84: 56–66

Lou T C, Tang J (2013). Mining structural hole spanners through information diffusion in social networks. In: Proceedings of the 22nd International Conference on World Wide Web. Rio de Janeiro: Association for Computing Machinery, 825–836

Lü L, Zhang Y C, Yeung C H, Zhou T (2011). Leaders in social networks, the delicious case. PLoS One, 6(6): e21202

Lü L Y, Chen D B, Ren X L, Zhang Q M, Zhang Y C, Zhou T (2016). Vital nodes identification in complex networks. Physics Reports, 650: 1–63

Lusseau D, Schneider K, Boisseau O J, Haase P, Slooten E, Dawson S M (2003). The bottlenose Dolphin community of doubtful sound features a large proportion of long-lasting associations. Behavioral Ecology and Sociobiology, 54(4): 396–405

Ma L L, Ma C, Zhang H F, Wang B H (2016). Identifying influential spreaders in complex networks based on gravity formula. Physica A: Statistical Mechanics and Its Applications, 451: 205–212

Ma S J, Ren Z M, Ye C M, Guo Q, Liu J G (2014). Node influence identification via resource allocation dynamics. International Journal of Modern Physics C, 25(11): 1450065

Ma T H, Liu Q, Cao J, Tian Y, Al-Dhelaan A, Al-Rodhaan M (2020). LGIEM: Global and local node influence based community detection. Future Generation Computer Systems, 105: 533–546

Macqueen J (1967). Some methods for classification and analysis of multivariate observations. In: Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability. Berkeley, CA: University of California Press, 281–297

Maji G (2020). Influential spreaders identification in complex networks with potential edge weight based *k*-shell degree neighborhood method. Journal of Computational Science, 39: 101055

Maji G, Mandal S, Sen S (2020). A systematic survey on influential spreaders identification in complex networks with a focus on *k*-shell based techniques. Expert Systems with Applications, 161: 113681

Massa P, Salvetti M, Tomasoni D (2009). Bowling alone and trust decline in social network sites. In: Proceedings of 8th IEEE International Conference on Dependable, Autonomic and Secure Computing. Chengdu, 658–663

McAuley J, Leskovec J (2012). Learning to discover social circles in ego networks. In: Proceedings of the 25th International Conference on Neural Information Processing Systems. Lake Tahoe, NV: Curran Accociates, 539–547

Moore B (1981). Principal component analysis in linear systems: Controllability, observability, and model reduction. IEEE Transactions on Automatic Control, 26(1): 17–32

Muthukrishna M, Schaller M (2020). Are collectivistic cultures more

prone to rapid transformation? Computational models of cross-cultural differences, social network structure, dynamic social influence, and cultural change. Personality and Social Psychology Review, 24(2): 103–120

Namtirtha A, Dutta A, Dutta B (2018). Weighted *k*-shell degree neighborhood method: An approach independent of completeness of global network structure for identifying the influential spreaders. In: 10th International Conference on Communication Systems & Networks. Bengaluru: IEEE, 81–88

Namtirtha A, Dutta A, Dutta B, Sundararajan A, Simmhan Y (2021). Best influential spreaders identification using network global structural properties. Scientific Reports, 11(1): 2254

Nargundkar A, Rao Y S (2016). InfluenceRank: A machine learning approach to measure influence of Twitter users. In: International Conference on Recent Trends in Information Technology. Chennai: IEEE, 1–6

Newman M E J (2001). The structure of scientific collaboration networks. Proceedings of the National Academy of Sciences of the United States of America, 98(2): 404–409

Newman M E J (2006). Finding community structure in networks using the eigenvectors of matrices. Physical Review E: Statistical, Nonlinear, and Soft Matter Physics, 74(3): 036104

Niepert M, Ahmed M, Kutzkov K (2016). Learning convolutional neural networks for graphs. In: Proceedings of the 33rd International Conference on Machine Learning. New York, NY: JMLR.org, 2014–2023

Ou Y, Guo Q, Xing J L, Liu J G (2022). Identification of spreading influence nodes via multi-level structural attributes based on the graph convolutional network. Expert Systems with Applications, 203: 117515

Pal S K, Kundu S, Murthy C A (2014). Centrality measures, upper bound, and influence maximization in large scale directed social networks. Fundamenta Informaticae, 130(3): 317–342

Palla G, Derényi I, Farkas I, Vicsek T (2005). Uncovering the overlapping community structure of complex networks in nature and society. Nature, 435(7043): 814–818

Pan R K, Saramäki J (2012). The strength of strong ties in scientific collaboration networks. Europhysics Letters, 97(1): 18007

Pan Y, Li D H, Liu J G, Liang J Z (2010). Detecting community structure in complex networks via node similarity. Physica A: Statistical Mechanics and Its Applications, 389(14): 2849–2857

Peng C, Wang X, Pei J, Zhu W (2019). A survey on network embedding. IEEE Transactions on Knowledge and Data Engineering, 31(5): 833–852

Qi X, Duval R D, Christensen K, Fuller E, Spahiu A, Wu Q, Wu Y, Tang W, Zhang C (2013). Terrorist networks, network energy and node removal: A new measure of centrality based on Laplacian energy. Social Networking, 2(1): 19–31

Qiu L Q, Jia W, Yu J F, Fan X, Gao W W (2019). PHG: A three-phase algorithm for influence maximization based on community structure. IEEE Access, 7: 62511–62522

Ren X, Zhu Y, Wang S, Liao H, Han X, Lü L (2015). Online social network analysis and the relation with regional economic development. Journal of University of Electronic Science and Technology of China, 44(5): 643–651 (in Chinese)

Ren X L, Lü L Y (2013). Review of ranking nodes in complex networks. Chinese Science Bulletin, 59(13): 1175–1197

Ren Z M (2020). Node influence of the dynamic networks. Acta Physica Sinica, 69(4): 24–32 (in Chinese)

Ren Z M, Liu J G, Shao F, Hu Z L, Guo Q (2013a). Analysis of the spreading influence of the nodes with minimum *k*-shell value in complex networks. Acta Physica Sinica, 62(10): 108902

Ren Z M, Shao F, Liu J G, Guo Q, Wang B H (2013b). Node importance measurement based on the degree and clustering coefficient information. Acta Physica Sinica, 62(12): 128901

Sabidussi G (1966). The centrality index of a graph. Psychometrika, 31(4): 581–603

Sacchet M D, Prasad G, Foland-Ross L C, Thompson P M, Gotilb I H (2014). Elucidating brain connectivity networks in major depressive disorder using classification-based scoring. In: 11th International Symposium on Biomedical Imaging. Beijing: IEEE, 246–249

Shang J X, Zhou S B, Li X, Liu L C, Wu H C (2017). CoFIM: A community-based framework for influence maximization on large-scale networks. Knowledge-Based Systems, 117: 88–100

Shang Q, Deng Y, Cheong K H (2021). Identifying influential nodes in complex networks: Effective distance gravity model. Information Sciences, 577: 162–179

Sheikhahmadi A, Nematbakhsh M A, Shokrollahi A (2015). Improving detection of influential nodes in complex networks. Physica A: Statistical Mechanics and Its Applications, 436: 833–845

Silva T C, Zhao L (2012). Network-based high level data classification. IEEE Transactions on Neural Networks and Learning Systems, 23(6): 954–970

Soffer S N, Vázquez A (2005). Network clustering coefficient without degree-correlation biases. Physical Review E: Statistical, Nonlinear, and Soft Matter Physics, 71(5): 057101

Spring N, Mahajan R, Wetherall D (2002). Measuring ISP topologies with rocketfuel. ACM SIGCOMM Computer Communication Review, 32(4): 133–145

Su X P, Song Y R (2015). Leveraging neighborhood "structural holes" to identifying key spreaders in social networks. Acta Physica Sinica, 64(2): 020101

Sun H L, Chen D B, He J L, Chng E (2019). A voting approach to uncover multiple influential spreaders on weighted networks. Physica A: Statistical Mechanics and Its Applications, 519: 303–312

Tang L, Liu H (2009). Relational learning via latent social dimensions. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Paris, 817–826

Tang L Y, Li S N, Lin J H, Guo Q, Liu J G (2016). Community structure detection based on the neighbor node degree information. International Journal of Modern Physics C, 27(4): 1650046

Tixier A J P, Rossi M E G, Malliaros F D, Read J, Vazirgiannis M (2019). Perturb and combine to identify influential spreaders in real-world networks. In: Proceedings of the IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. Vancouver, 73–80

Tulu M M, Hou R, Younas T (2018). Identifying influential nodes based on community structure to speed up the dissemination of information in complex network. IEEE Access, 6: 7390–7401

Ullah A, Wang B, Sheng J, Long J, Khan N, Sun Z (2021). Identification of nodes influence based on global structure model in complex networks. Scientific Reports, 11(1): 6173

Wang F, She J, Ohyama Y, Wu M (2019). Deep-learning-based identification of influential spreaders in online social networks. In: IECON 45th Annual Conference of the IEEE Industrial Electronics Society. Lisbon, 6854–6858

Wang Y, Cong G, Song G J, Xie K Q (2010). 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. Washington, D.C., 1039–1048

Wang Y F, Yan G H, Ma Q Q, Wu Y, Zhang M (2018). Identifying influential nodes based on vital communities. In: 16th International Conference on Dependable, Autonomic and Secure Computing, 16th International Conference on Pervasive Intelligence and Computing, 4th International Conference on Big Data Intelligence and Computing and Cyber Science and Technology Congress. Athens: IEEE, 314–317

Wang Z X, Zhao Y, Xi J K, Du C J (2016). Fast ranking influential nodes in complex networks using a *k*-shell iteration factor. Physica A: Statistical Mechanics and Its Applications, 461: 171–181

Watts D J, Dodds P S (2007). Influential, networks, and public opinion formation. Journal of Consumer Research, 34(4): 441–458

Watts D J, Strogatz S H (1998). Collective dynamics of "small-world" networks. Nature, 393(6684): 440–442

Wei H, Pan Z, Hu G, Zhang L, Yang H, Li X, Zhou X (2018). Identifying influential nodes based on network representation learning in complex networks. PLoS One, 13(7): e0200091

Wu Z, Pan S, Chen F, Long G, Zhang C, Yu P S (2021). A comprehensive survey on graph neural networks. IEEE Transactions on Neural Networks and Learning Systems, 32(1): 4–24

Xie N (2006). Social Network Analysis of Blogs. Dissertation for the Master's Degree. Bristol: University of Bristol

Yan S, Tang S T, Pei S S, Jiang J, Zhang X, Ding W R, Zheng M Z (2013). The spreading of opposite opinions on online social networks with authoritative nodes. Physica A: Statistical Mechanics and Its Applications, 392(17): 3846–3855

Yan X L, Cui Y P, Ni S J (2020). Identifying influential spreaders in complex networks based on entropy weight method and gravity law. Chinese Physics B, 29(4): 048902

Yang J, Leskovec J (2012). Defining and evaluating network communities based on ground-truth. In: Proceedings of the ACM SIGKDD Workshop on Mining Data Semantics. Beijing, 1–8

Yang J, Leskovec J (2013). Overlapping community detection at scale: A nonnegative matrix factorization approach. In: Proceedings of the 6th ACM International Conference on Web Search and Data Mining. Rome, 587–596

Yang J N, Liu J G, Guo Q (2018a). Node importance identification for temporal network based on inter-layer similarity. Acta Physica Sinica, 67(4): 279–286 (in Chinese)

Yang K, Guo Q, Liu J G (2018b). Community detection via measuring the strength between nodes for dynamics networks. Physica A: Statistical Mechanics and Its Applications, 509: 256–264

Yang X H, Xiong S (2021). Identification of node influence using network representation learning in complex network. Journal of Chinese Computer Systems, 42(2): 418–423 (in Chinese)

Yang Y Z, Wang X, Chen Y, Hu M, Ruan C W (2020). A novel centrality of influential nodes identification in complex networks. IEEE Access, 8: 58742–58751

Yin R R, Guo Q, Yang J N, Liu J G (2018). Inter-layer similarity-based eigenvector centrality measures for temporal networks. Physica A: Statistical Mechanics and Its Applications, 512: 165–173

Yu E Y, Wang Y P, Fu Y, Chen D B, Xie M (2020). Identifying critical nodes in complex networks via graph convolutional networks. Knowledge-Based Systems, 198: 105893

Yu S B, Gao L, Xu L D, Gao Z Y (2019). Identifying influential spreaders based on indirect spreading in neighborhood. Physica A: Statistical Mechanics and Its Applications, 523: 418–425

Zachary W W (1977). An information flow model for conflict and fission in small groups. Journal of Anthropological Research, 33(4): 452–473

Zareie A, Sheikhahmadi A, Jalili M (2019). Influential node ranking in social networks based on neighborhood diversity. Future Generation Computer Systems, 94: 120–129

Zeng A C, Zhang J (2013). Ranking spreaders by decomposing complex networks. Physics Letters A, 377(14): 1031–1035

Zhang D, Wang Y, Zhang Z (2019a). Identifying and quantifying potential super-spreaders in social networks. Scientific Reports, 9(1): 14811

Zhang J X, Chen D B, Dong Q, Zhao Z D (2016). Identifying a set of influential spreaders in complex networks. Scientific Reports, 6(1): 27823

Zhang M H, Chen Y X (2018). Link prediction based on graph neural networks. In: Proceedings of the 32nd International Conference on Neural Information Processing Systems. Montreal: Curran Associates Inc., 5171–5181

Zhang W, Yang J, Ding X Y, Zou X M, Han H Y, Zhao Q C (2019b). Groups make nodes powerful: Identifying influential nodes in social networks based on social conformity theory and community features. Expert Systems with Applications, 125: 249–258

Zhao G H, Jia P, Huang C, Zhou A, Fang Y (2020a). A machine learning based framework for identifying influential nodes in complex networks. IEEE Access, 8: 65462–65471

Zhao G H, Jia P, Zhou A, Zhang B (2020b). InfGCN: Identifying influential nodes in complex networks with graph convolutional networks. Neurocomputing, 414: 18–26

Zhao X Y, Huang B, Tang M, Zhang H F, Chen D B (2014a). Identifying effective multiple spreaders by coloring complex networks. Europhysics Letters, 108(6): 68005

Zhao Z J, Guo Q, Yu K, Liu J G (2020c). Identifying influential nodes for the networks with community structure. Physica A: Statistical Mechanics and Its Applications, 551: 123893

Zhao Z Y, Yu H, Zhu Z L, Wang X F (2014b). Identifying influential spreaders based on network community structure. Chinese Journal of Computers, 37(4): 753–766 (in Chinese)

Zhao Z Y, Wang X F, Zhang W, Zhu Z L (2015). A community-based approach to identifying influential spreaders. Entropy, 17(4): 2228–2252

Zhou M Y, Xiong W M, Wu X Y, Zhang Y X, Liao H (2018). Overlapping influence inspires the selection of multiple spreaders in complex networks. Physica A: Statistical Mechanics and Its Applications, 508: 76–83

Zhou T, Lü L Y, Zhang Y C (2009). Predicting missing links via local information. European Physical Journal B, 71(4): 623–630