

SLGC: Identifying influential nodes in complex networks from the perspectives of self-centrality, local centrality, and global centrality

Da Ai(艾达), Xin-Long Liu(刘鑫龙), Wen-Zhe Kang(康文哲), Lin-Na Li(李琳娜),Shao-Qing Lu(吕少卿), and Ying Liu(刘颖)

Citation: Chin. Phys. B, 2023, 32 (**11**): 118902. DOI: 10.1088/1674-1056/aceee8
Journal homepage: http://cpb.iphy.ac.cn; http://iopscience.iop.org/cpb

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

Robustness of community networks against cascading failures with heterogeneous redistribution strategies

Bo Song(宋波), Hui-Ming Wu(吴惠明), Yu-Rong Song(宋玉蓉), Guo-Ping Jiang(蒋国平),Ling-Ling Xia(夏玲玲), and Xu Wang(王旭)
Chin. Phys. B, 2023, 32 (**9**): 098905. DOI: 10.1088/1674-1056/acd9c3

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

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

An extended improved global structure model for influential node identification in complex networks

Jing-Cheng Zhu(朱敬成) and Lun-Wen Wang(王伦文)
Chin. Phys. B, 2022, 31 (**6**): 068904. DOI: 10.1088/1674-1056/ac380d

A novel method for identifying influential nodes in complex networks based on gravity model

Yuan Jiang(蒋沅), Song-Qing Yang(杨松青), Yu-Wei Yan(严玉为),Tian-Chi Tong(童天驰), and Ji-Yang Dai(代冀阳)
Chin. Phys. B, 2022, 31 (**5**): 058903. DOI: 10.1088/1674-1056/ac4226

Detection of influential nodes with multi-scale information

Jing-En Wang(王静恩), San-Yang Liu(刘三阳), Ahmed Aljmiai, and Yi-Guang Bai(白艺光)
Chin. Phys. B, 2021, 30 (**8**): 088902. DOI: 10.1088/1674-1056/abff2d

---

SLGC: Identifying influential nodes in complex networks from the perspectives of self-centrality, local centrality, and global centrality

Da Ai(艾达)[1,†], Xin-Long Liu(刘鑫龙)[1], Wen-Zhe Kang(康文哲)[1], Lin-Na Li(李琳娜)[1,2], Shao-Qing Lü(吕少卿)[1], and Ying Liu(刘颖)[1]

[1] *School of Communication and Information Engineering, Xi'an University of Posts and Telecommunications, Xi'an* 710121, *China*
[2] *Network Ecological and Environmental Governance Research Center, Xi'an University of Posts and Telecommunications, Xi'an* 710121, *China*

(Received 19 June 2023; revised manuscript received 7 August 2023; accepted manuscript online 10 August 2023)

Identifying influential nodes in complex networks and ranking their importance plays an important role in many fields such as public opinion analysis, marketing, epidemic prevention and control. To solve the issue of the existing node centrality measure only considering the specific statistical feature of a single dimension, a SLGC model is proposed that combines a node's self-influence, its local neighborhood influence, and global influence to identify influential nodes in the network. The exponential function of e is introduced to measure the node's self-influence; in the local neighborhood, the node's one-hop neighboring nodes and two-hop neighboring nodes are considered, while the information entropy is introduced to measure the node's local influence; the topological position of the node in the network and the shortest path between nodes are considered to measure the node's global influence. To demonstrate the effectiveness of the proposed model, extensive comparison experiments are conducted with eight existing node centrality measures on six real network data sets using node differentiation ability experiments, susceptible–infected–recovered (SIR) model and network efficiency as evaluation criteria. The experimental results show that the method can identify influential nodes in complex networks more accurately.

Keywords: influential nodes, self-influence, local and global influence, complex networks

PACS: 89.75.Fb **DOI:** 10.1088/1674-1056/aceee8

1. Introduction

Social structures, road systems, social networks, etc. can be abstracted as complex networks consisting of nodes and connected edges.[1] The opening of the metaverse era, where the node data in complex networks may be virtual and digital, and the connections between nodes may also be virtual, digital, or connected to the real world, has increased the heterogeneity and uncertainty of complex networks while increasing the difficulty of mining important nodes in complex networks.[2] The critical node detection problem (CNDP) is typical in complex network analysis, where its enhancement or failure will significantly improve or degrade the functionality of the target network and can better understand or control the behavior of nodes in complex networks.[3] Identifying a set of influential and important nodes in complex networks plays a crucial role in advertising placement,[4] rumor control,[5] epidemic prevention and control,[6] etc. With the complexity of human society and the increasing demand for large-scale, complex systems, the study of the importance of complex network nodes has received increasing attention. The study of mining important nodes in complex networks has important research significance and practical value for network data science and mining network functions.

In complex networks, different types of network nodes have specific properties, and some special nodes that can manipulate the entire network data transmission and information dissemination are called important nodes,[7,8] for example, predictive dissemination of financial risk research institutions in the field of finance,[9] online social networking platforms with a strong ability to disseminate online bloggers,[10] and important transportation hubs that undertake transportation in road traffic networks.[11] Thus, several ranking algorithms have been proposed to evaluate the importance of nodes based on network features or node attributes, which can be broadly classified into categories such as node neighborhood-based, node path-based, node feature vector-based, node location-based, and node multi-attribute decision-based methods.[12]

Measuring the importance of a node based on its local neighborhood primarily involves evaluating its significance by considering its own information within the local neighborhood structure of the network and the importance of neighboring nodes within that neighborhood. Degree centrality (DC)[13]and H-index[14] are classical algorithms based on the node neighborhood, which mainly consider the number or degree of one-hop neighboring nodes to measure the importance of nodes and are an intuitive and simple node ranking algorithm, but because only one-hop neighboring nodes are considered, the influence of more distant nodes and network topology information is ignored, resulting in the accuracy rate of identifying important nodes not high.[15]

Although algorithms based on local neighborhood node importance sorting can quickly and intuitively reflect the im-

[†]Corresponding author. E-mail: aida@xupt.edu.cn

© 2023 Chinese Physical Society and IOP Publishing Ltd

portance of nodes, their low accuracy leads to ranking results that are not suitable for practical applications. Subsequently, researchers proposed a class of path-based sorting algorithms that measure the importance of nodes in the global network based on the number of shortest paths between nodes.[16] The representative node-path-based algorithms are betweenness centrality (BC)[17] and closeness centrality (CC),[18] BC by calculating the number of shortest paths between nodes in the network and CC by calculating the inverse of the average distance from nodes to other nodes in the network, this type of node-centricity algorithm focuses on the global structure of the network and only considers the shortest path information between nodes without considering the path, which will not be able to describe the importance of the nodes for unconnected networks and lead to the high computational complexity of the model, which has some limitations for large-scale networks.[19]

The importance of a node depends not only on the number of its neighboring nodes but also on the importance of those neighbors. The idea behind feature vector-based sorting algorithms is to calculate the weight value of a node based on the eigenvector of the network adjacency matrix. It emphasizes both the quantity and quality of neighboring nodes in the network where the node resides. The importance of a node is evaluated by considering the relationship between its own weight and the weights of its neighboring nodes. Eigenvector centrality (EC)[20] and PageRank (PR)[21] algorithms are representative algorithms based on the node eigenvector centrality. However, due to the involvement of matrix operations, this type of algorithm has high computation complexity and slow convergence speed, making it unsuitable for large-scale networks and prone to being trapped in loops.[22]

In addition, by decomposing the complex network structure, the importance of a node is determined based on its position in the network. The classical algorithms based on node location are K-shell decomposition, which divides the nodes in the network into different levels according to K-shell values, and the closer a node is to the center of the network, the more influential that node is, but this method cannot distinguish the importance of nodes in the same level.[23]

With the increase of uncertainty in complex networks, researchers have proposed a node centrality measure based on the multiple attribute decision-making methods (MADMs).[24,25] The idea of this algorithm is to define node centrality as a combined performance under multiple attribute perspectives, or to use the technique for order of preference by similarity to ideal solution (TOPSIS) to perform weighted calculations based on different attribute weights, and ultimately obtain a comprehensive ranking result for the nodes.[26] Wang *et al.*[27] proposed a centroidal of centripetalism to evaluate the importance of nodes more comprehensively from their global, local, and semi-local topological information. Qiu *et al.*[28] improved the traditional degree centrality by introducing the concept of local clustering coefficients to measure the local influence of nodes, improving the classical K-shell algorithm to measure the global influence of nodes, and proposing a local and global influence (LGI) algorithm for identification and ranking of influential nodes in complex networks.

In the aforementioned method based on the multi-attribute decision-making for nodes, the comprehensive consideration of local and global features of nodes overlooks the self-influence of nodes. In this paper, we propose a new node importance ranking algorithm called SLGC algorithm from the perspective of multi-attribute decision-making for nodes, considering the self-influence of nodes. Compared to existing methods based on the multi-attribute decision-making for nodes, the proposed SLGC algorithm takes into account the self-influence of nodes, defines a new metric for local influence by considering the quantity and quality of the neighboring nodes within the node's local neighborhood, and introduces the concept of Shannon entropy. Additionally, it defines a new metric for global influence based on the node's position in the overall network and the topology structure between nodes. By integrating these three dimensions of metrics — node's self-influence, local-influence, and global-influence — the SLGC algorithm comprehensively determines the node's importance in a complex network and identifies influential nodes. The main innovation of our research in the SLGC algorithm lies in:

(i) From the perspective of the self-influence of nodes. Starting from the node's own attributes, we introduce an e-exponential function to limit the overestimation of the node's own importance due to large differences in node degrees. Based on the degree of the node and the maximum degree of the network, we define a new metric for self-influence of nodes, constraining it between (1, e).

(ii) From the perspective of the local influence of nodes. Within the node's local neighborhood, we consider the quantity and quality of the node's first-order and second-order neighboring nodes to define a new metric called node's local average degree. Then, we calculate the node's local decision probability and introduce Shannon entropy to construct the node's entropy-weighted local degree centrality as an evaluation of the node's local influence.

(iii) From the perspective of the global influence of nodes. In the global topological structure of the network, we assign each node with a corresponding K-shell value based on its position in the network. Simultaneously, we calculate the shortest path between any two nodes in the network to evaluate the node's global influence.

The rest of this article is composed as follows. Section 2 briefly introduces the basic concepts of complex net-

work graph data and node centrality ranking algorithm. Section 3 outlines the SLGC algorithm proposed in this paper. Section 4 analyzes the experimental results, including the data set, node distinguishability experiments, node propagation ability experiments, and network model efficiency experiments. And Section 5 summarizes our research results and algorithm highlights.

2. Preliminaries

Mathematically, any complex network can be represented as a graph $G=(V,E)$, where the set of nodes is denoted as $V=\{v_1,v_2,v_3,\ldots,v_n\}$ and the set of connected edge relations between nodes is denoted as $E=\{e_1,e_2,e_3,\ldots,e_m\}$.

Without loss of generality, we assume that the graph $G=(V,E)$ is an undirected and unweighted complex graph network data, and the information of nodes and edges in the complex network is described by the adjacency matrix $A=\{a_{ij}\}$. The adjacency matrix describes the connection relationship between each node, and for the undirected and unweighted complex network, the adjacency matrix of the nodes is a symmetric matrix, and both a_{ij} and a_{ji} in the adjacency matrix are 1 if there is a connection between node v_i and node v_j, otherwise they are 0.

2.1. Degree centrality

The degree of a node is defined as the number of neighboring nodes that are directly connected to it. According to the degree of a node, the more neighboring nodes that are directly connected to it, the greater its influence in the network[13]

$$k_i=\sum_{j\in\phi(i)}a_{ij}, \tag{1}$$

where a_{ij} denotes the existence of a connection between node v_i and node v_j in the network adjacency matrix.

Normalized degree centrality

$$\mathrm{DC}(v_i)=\frac{k_i}{N-1}, \tag{2}$$

where k_i denotes the degree of node v_i and N is the total number of nodes in the network.

2.2. Betweenness centrality

Betweenness centrality is a node centrality measure based on the global topology of the network, and the importance of a node is measured by the shortest path of the node, according to which the more the shortest path, the more important the node is[17]

$$\mathrm{BC}(v_i)=\sum_{j,k\neq i}\frac{g_{jk}(v_i)}{g_{jk}}, \tag{3}$$

where g_{jk} is the number of the shortest paths between node v_j and v_k. $g_{jk}(v_i)$ is the number of shortest paths between node v_j and node v_k through node v_i.

2.3. Closeness centrality

Closeness centrality is a centrality measure based on the global topology of the network, where the closeness centrality of a node is calculated as the reciprocal of the sum of the shortest paths between the node and other nodes, and the closeness centrality of a node is defined based on the reciprocal of the average shortest path of the node[18]

$$\mathrm{CC}(v_i)=\frac{N-1}{\sum\limits_{j\neq1,j\neq i}d_{ij}}, \tag{4}$$

where N denotes the number of nodes in the network and d_{ij} is the shortest path between node v_i and node v_j.

2.4. Eigenvector centrality

The importance of a node takes into account not only the information of the node itself but also the information of neighboring nodes. Feature vector centrality defines node importance based on the centrality of neighboring nodes. Based on the adjacency matrix A of a given network, the feature vector centrality is defined as[20]

$$Ax=\lambda x,\ \ \mathrm{EC}(v_i)=x_i=\frac{1}{\lambda}\sum_{j=1}^{n}a_{ij}x_j,\ \ i=1,2,3,\ldots,n, \tag{5}$$

where λ is the maximum eigenvalue of the adjacency matrix of the network and $1/\lambda$ is used as a scaling constant, only the eigenvector corresponding to the maximum eigenvalue is required for the centrality measure, and $x^{\mathrm{T}}=\{x_1,x_2,\ldots,x_n\}$ is the corresponding eigenvector.

2.5. PageRank centrality

The PageRank algorithm is a variant of eigenvector centrality, which was initially used by Google to rank web pages. The algorithm is calculated based on the structure and number of connections of nodes in the network, and the PR of a node is not simply calculated with the number of its neighboring nodes, but the sum of the weights of the related nodes, which are derived based on the iterative formula of the PageRank, which is calculated as follows:[21]

$$\mathrm{PRC}(v_i)=\frac{1-d}{N}+d\sum_{v_j\in\Gamma(v_i)}\frac{\mathrm{PR}(v_j)}{\mathrm{Out}(v_j)}, \tag{6}$$

where $\mathrm{PRC}(v_i)$ denotes the PageRank value of node v_i, d is the damping factor, N is the total number of the network nodes, $\mathrm{PR}(v_j)$ denotes the PageRank value of node v_j, $\mathrm{Out}(v_j)$ denotes the number of connections from node v_i to other nodes.

2.6. Clustered local-degree centrality

The algorithm calculates the sum of the degrees of the nearest neighbors of a given node and introduces the clustering coefficients of the nodes to propose a new centrality measure called clustering local degree centrality for identifying important nodes in the network, defined as follows:[29]

$$\mathrm{CLD}(v_i) = [1 + C(v_i)] \sum_{v_j \in \Gamma(v_i)} k(v_j), \tag{7}$$

$$C(v_i) = \frac{2L_i}{(k_i(k_i - 1))}, \tag{8}$$

where k_i denotes the degree of the node, $C(v_i)$ denotes the clustering coefficient of the node, and L_i denotes the number of contiguous edges existing between the neighboring nodes of node v_i.

2.7. Global structure model

The algorithm proposes a new node centrality metric for identifying influential nodes based on the node global structure model, which considers the node's self influence while emphasizing the node's global influence in the network topology, defined as follows:[30]

$$\mathrm{GSM}(v_i) = \mathrm{e}^{\frac{Ks(v_i)}{N}} \times \sum_{i \neq j} \frac{Ks(v_j)}{d_{ij}}, \tag{9}$$

where $Ks(v_i)$ denotes the K-shell value of the node and d_{ij} denotes the shortest distance between nodes v_i and v_j.

2.8. Local-and-global centrality

The algorithm considers the degree of nodes and the shortest distance between nodes and proposes a new algorithm based on local and global centrality measures of nodes to identify key nodes by processing the local and global topology of nodes in the network, defined as follows:[31]

$$\mathrm{LGC}(v_i) = \frac{d(v_i)}{n} \times \sum_{i \neq j} \frac{\sqrt{d(v_j) + \alpha}}{d_{ij}}, \tag{10}$$

where $d(v_i)$ denotes the degree of the node, d_{ij} denotes the shortest distance between nodes v_i and v_j, and α, as the degree of influence of the adjustable parameter control network, lies in the range between 0 and 1.

3. The proposed method

The importance of nodes in a complex network is determined by three key factors: the influence of a node, the local influence of a node, and the global influence of a node. The influence of a node is mainly provided by the degree of the node, while the influence of a node is measured by introducing the exponential function of e. Local influence is determined based on the node's one-hop neighboring nodes and two-hop neighboring nodes jointly, and the concept of information entropy is introduced to reflect the information sharing ability between the node and its one- and two-hop neighboring nodes. The global influence considers the information dissemination ability of the node in terms of the global structure of the whole network, i.e., the location and shortest path of the node in the network.

3.1. Definitions related to the SLGC algorithm

To describe the node importance evaluation index in complex networks, the following definitions of node's self-influence, local influence, and global influence are given in this paper.

Definition 1. Self-influence of node (SIN) The degree of a node can directly reflect its information dissemination capacity. Nodes with higher degrees tend to be more important. However, this may lead to the neglect of nodes with lower degrees. To avoid this situation, we introduce the e-exponential function to limit the node's self-influence within the range of (1, e), thereby preventing an excessive evaluation of the node's own importance, defining SIN, the node's self-influence is calculated as

$$\mathrm{SIN}(v_i) = \mathrm{e}^{\frac{d(v_i)}{1 + k_{\max}}}, \tag{11}$$

where $d(v_i)$ is the degree of the node and $k_{\max}$ is the maximum degree of the node in the network.

Definition 2. Local-influence of nodes (LIN) The importance of a node depends not only on the number of nodes it directly connects to but also on the quality of its immediate neighbors and second-order neighbors. The larger the degree of a node's immediate neighbors and second-order neighbors, the more significant the node is considered to be. Therefore, a new local degree centrality method is proposed as follows by introducing the Shannon entropy[32] in the local information dimension of the node.

(i) Local average degree The local average degree of a node reflects the average level of connectivity within its local neighborhood, revealing the node's capacity for information sharing and propagation with its neighboring nodes. The local average degree is calculated as the ratio of the sum of the degrees of a node's first-order and second-order neighboring nodes to the degree of the node itself, denoted by k_{nn} and defined as follows:

$$k_{\mathrm{nn}}(v_i) = \frac{\sum\limits_{v_j \in \Gamma(v_i)} d(v_j) + \sum\limits_{v_k \in \Gamma(v_j)} d(v_k)}{d(v_i)}, \tag{12}$$

where $d(v_i)$ denotes the degree of the node v_i, $d(v_j)$ and $d(v_k)$ denote the degree of the node's one-hop neighboring nodes v_j and two-hop neighboring nodes v_k, and $\Gamma(v_i)$ and $\Gamma(v_j)$ denote the set of nearest neighbor nodes and two-hop neighboring nodes of the node v_i.

(ii) Local decision probability Based on the node's own attributes and the network structure between its first-order and

second-order neighboring nodes, the local decision probability of a node is defined to evaluate its level of connectivity within the local neighborhood network, thereby revealing its importance in information dissemination or control. The local decision probability of a node is defined as the reciprocal of the local average degree and is expressed as follows:

$$p(v_i) = \frac{1}{k_{\mathrm{nn}}(v_i)}. \tag{13}$$

(iii) Node local information entropy Shannon entropy is a concept in information theory used to measure the uncertainty of a random variable. In the context of ranking the importance of nodes in complex networks, nodes can be seen as information sources, and their neighboring nodes can be seen as information related to that source. By introducing the definition of the Shannon entropy, we can evaluate the information transmission capacity of a node within its local neighborhood. Based on the local decision probability of each node involved in information propagation, the definition of node's local information entropy is as follows:

$$E(v_i) = -\sum_{i=0}^{v_j \in \Gamma(v_i)} p(v_i) \log p(v_i), \tag{14}$$

where the probability of the local information entropy of a node is represented by the local decision probability of the node, and $\Gamma(v_i)$ denotes the set of one-hop neighboring nodes of node v_i.

(iv) Node local influence On the basis of the node degree centrality measurement method, the node information entropy is introduced to construct the entropy-weighted local degree centrality as an indicator to measure the local influence of nodes, which is defined as follows:

$$\mathrm{LIN}(v_i) = [1 + E(v_i)] \frac{d(v_i)}{N}, \tag{15}$$

where $E(v_i)$ is the information entropy of node v_i, $d(v_i)$ is the degree of node v_i, and N is the total number of nodes in the network.

Definition 3. Global-influence of node (GIN) When considering only the local influence of a node often loses information about the topology of the node network, the time and ability of the node to spread information with other nodes as one of the indicators to measure the global influence of the node, the K-shell value of the node's direct neighbors also indirectly increases the influence of the node, the K-shell value measures the nucleus of each node in the network from the global dimension of the network, the shortest path between any two nodes in the network directly affects the ability of the node to share information with other nodes, the shortest distance between two nodes is inversely proportional to the node, so the global dimension influence is defined as follows:

$$\mathrm{GIN}(v_i) = \sum_{i \neq j} \frac{\sqrt{Ks(v_j) \times d(v_j)}}{d_{ij}^2}, \tag{16}$$

where $Ks(v_j)$ denotes the value of K-shell of nodes and d_{ij} denotes the shortest path between node v_i and node v_j.

Definition 4. Total influence of node According to the above-proposed node's self-influence, local influence, and global influence, the product of the three is used as the basis for assessing the importance of the node by the SLGC method proposed in this paper, therefore, the SLGC value of the node is defined as

$$\begin{aligned} \mathrm{SLGC}(v_i) &= \mathrm{SIN}(v_i) \times \mathrm{LIN}(v_i) \times \mathrm{GIN}(v_i) \\ &= \mathrm{e}^{\frac{d(v_i)}{1+k_{\max}}} \times [1 + E(v_i)] \frac{d(v_i)}{N} \\ &\quad \times \sum_{i \neq j} \frac{\sqrt{Ks(v_j) \times d(v_j)}}{d_{ij}^2}. \end{aligned} \tag{17}$$

Algorithm 1 provides the algorithmic description of SLGC.

Algorithm 1 SLGC

Input: A network $G = (V, E)$ // G is an undirected and unweighted graph with V as the set of nodes and E as the set of edges.
Output: A ranking list of node's influence.
Begin
Initialize the information
For each node v in V do
  For each node u in V do
  Calculate the degree $\deg(v)$ and the max degree $k_{\max}$
  Calculate the $Ks(v)$ using K-shell decomposition
  Using Dijkstra calculate the shortest distance d_{uv}
  End for
End for
$\mathrm{SIN} = \emptyset$ // self influence
$\mathrm{LIN} = \emptyset$ // local influence
$\mathrm{GIN} = \emptyset$ // global influence
For each node in V do
  Calculate self influence $\mathrm{SIN}(v)$ using Eq. (11).
End for
For each node v in V do
  Calculate the k_{nn} using Eq. (12).
For each $(u, v) \in E$ do
  $p_{\mathrm{uv}} = 1/k_{\mathrm{nn}}$
  $H_{\mathrm{uv}} = -p_{\mathrm{uv}} \log p_{\mathrm{uv}}$
  For each node v in V do
  $E_v = \sum_{u \in \Gamma_v} H_{\mathrm{uv}}$
  Calculate local influence $\mathrm{LIN}(v)$ using Eq. (15).
  End for
End for
For each node v in V do
  Calculate global influence $\mathrm{GIN}(v)$ using Eq. (16).
  $\forall$ indices $i \neq j$ for Eqs. (16) and (17).
End for
For each node v in V do
  Calculate $\mathrm{SLGC}(v)$ using Eq. (17).
End for
Sort the nodes in descending order based on SLGC scores to obtain the ranking list.
End

3.2. Example of SLGC algorithm

The procedure of the proposed method is illustrated below using a simple random network. The random network is shown in Fig. 1, which contains 16 nodes and 21 edges. By using the SLGC algorithm for this random network, the important nodes in the network are identified and the nodes are ranked.

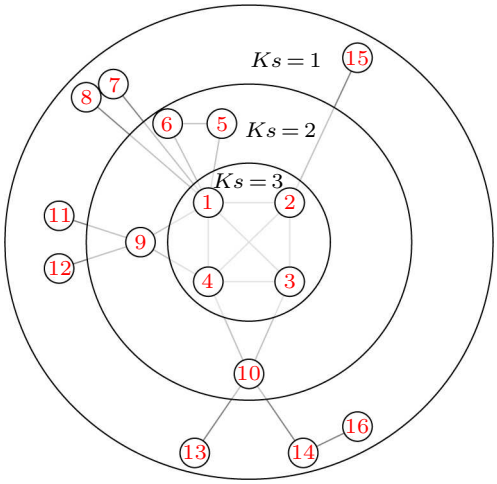


Fig. 1. The random network for illustration.

Calculate step-by-step according to the SLGC algorithm flow.

Step 1: in a given network, the first step is to perform K-shell decomposition by partitioning the network based on the position of nodes within the network, resulting in each node being assigned a K-shell value. Simultaneously, the degree value of each node is calculated. The values are shown in the following table.

Table 1. Degree and Ks values of nodes.

Node	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}	v_{11}	v_{12}	v_{13}	v_{14}	v_{15}	v_{16}
$d(v_i)$	8	4	4	5	2	2	1	1	4	4	1	1	1	2	1	1
$Ks(v_i)$	3	3	3	3	2	2	1	1	2	2	1	1	1	1	1	1

Step 2: by calculating the degree values of nodes in the given random network, we obtain the maximum degree value $k_{\max}$ as 8. Based on the degree values of each node obtained in the first step and the maximum degree value in the network, we can calculate the SIN value of nodes in the random network using the e-exponential function.

Table 2. SIN value of the node.

Node	SIN(v_i)	Node	SIN(v_i)
v_1	2.4324	v_9	1.5596
v_2	1.5596	v_{10}	1.5596
v_3	1.5596	v_{11}	1.1175
v_4	1.7429	v_{12}	1.1175
v_5	1.2488	v_{13}	1.1175
v_6	1.2488	v_{14}	1.2488
v_7	1.1175	v_{15}	1.1175
v_8	1.1175	v_{16}	1.1175

Step 3: to calculate the local average degree and local clustering probability of each node in a random network based on the node's degree within its local neighborhood and the network structure, we introduce the concept of Shannon entropy to compute the local information entropy of each node. Subsequently, we can calculate the LIN value of nodes in the random network.

Table 3. LIN value of the node.

Node	$k_{\text{nn}}(v_i)$	$p(v_i)$	$E(v_i)$	LIN(v_i)
v_1	5.75	0.1739	1.7794	1.3897
v_2	9.0	0.1111	1.0237	0.5059
v_3	8.0	0.1250	0.2441	0.5206
v_4	10.5	0.0952	1.0826	0.7311
v_5	9.6	0.1042	1.3396	0.1918
v_6	10.0	0.1000	0.5345	0.1918
v_7	10.0	0.1000	0.5345	0.0815
v_8	16.0	0.0625	0.3042	0.0815
v_9	16.0	0.0625	0.3042	0.0815
v_{10}	7.5	0.1333	1.0597	0.5103
v_{11}	6.0	0.1667	1.0414	0.0793
v_{12}	8.0	0.1250	0.2687	0.0793
v_{13}	8.0	0.1250	0.2687	0.0812
v_{14}	8.0	0.1250	0.2986	0.2057
v_{15}	5.0	0.2000	0.6452	0.0778
v_{16}	4.0	0.2500	0.3219	0.0826

Step 4: based on the degree and K-shell values of nodes in the random network obtained in step one, we can use Dijkstra's algorithm to calculate the shortest paths between any two nodes in the network, and then calculate the GIN value for each node in the random network.

Table 4. GIN value of the node.

Node	GIN(v_i)	Node	GIN(v_i)
v_1	21.3550	v_9	15.1530
v_2	16.6407	v_{10}	13.7992
v_3	18.4585	v_{11}	7.2755
v_4	20.4487	v_{12}	7.2755
v_5	11.6449	v_{13}	6.9334
v_6	11.6449	v_{14}	7.7187
v_7	10.3949	v_{15}	8.0942
v_8	10.3949	v_{16}	4.1071

Step 5: based on the above-obtained node SIN value, LIN value, and GIN value, calculate the SLGC value of the nodes and sort the nodes.

Table 5. SLGC values of nodes and sorting results.

Node	SLGC(v_i)	Rank	Node	SLGC(v_i)	Rank
v_1	72.18644	1	v_5, v_6	2.789185	7
v_4	26.05643	2	v_{14}	1.982765	8
v_3	14.98697	3	v_7, v_8	0.946729	9
v_2	13.12954	4	v_{15}	0.703722	1
v_9	12.16844	5	v_{11}, v_{12}	0.644738	11
v_{10}	10.98228	6	v_{16}	0.379108	12

From Table 5, it can be seen that node v_1 has the largest SLGC value, indicating that it has the strongest importance; nodes v_5 and v_6, nodes v_7 and v_8, and nodes v_{11} and v_{12} have the same SLGC value, indicating that they have the same importance; in Table 1, node v_9 and node v_{10} have the same degree value and K-shell value, therefore, the traditional method cannot distinguish the importance of the two nodes; whereas, from Table 3, it can be seen that the local influence of node v_9 is slightly higher than that of node v_{10}, and Table 4 shows that the global influence of node v_9 is higher than that of node v_{10}, which leads to a more obvious differentiation between the influence of node v_9 and node v_{10} in Table 5, indicating that the accuracy of the proposed method for node identification is better than that of the existing methods. The detailed experimental analysis will be given in Section 4.

3.3. Computational complexity analysis of SLGC

Time complexity is an important indicator to measure the quality of an algorithm. The proposed SLGC algorithm consists of three main components: computing the self-influence, local influence, and global influence of nodes. The time complexity of the SLGC algorithm is given below.

(i) For an undirected graph, when computing the self-influence of nodes, it is necessary to traverse all nodes n, calculate the degree value of each node and perform exponential function computation once, with a time complexity of $O(n)$.

(ii) When calculating the sum of degrees of one- and two-hop neighboring nodes for each node, it is necessary to traverse the first- and second-order neighboring nodes of each node once, resulting in a time complexity of $O(n^2)$.

(iii) When calculating the information content of a node, it is necessary to perform a local average degree calculation on the neighboring nodes of each node, with a time complexity of $O(n^2)$.

(iv) When computing the information entropy of nodes, it is necessary to traverse each node's neighbor nodes and accumulate information value during the traversal process, with a time complexity of $O(n^2)$.

(v) Using Dijkstra's algorithm to calculate the shortest path between two nodes or calculate the K-shell value of a node has a time complexity of $O(n^2)$.

(vi) When computing the self-influence, local influence, and global influence of nodes, it is necessary to perform one computation for each node, with a time complexity of $O(n)$.

Therefore, the time complexity of the SLGC algorithm is $O(n^2)$.

4. Experiment and analysis

In this section, we use six real-world complex networks with different properties as test datasets to verify the effectiveness of our proposed method. We compare our method with a total of eight most commonly used and latest node centrality measures, including degree centrality, betweenness centrality, closeness centrality, eigenvector centrality, PageRank, cluster local degree centrality (CLD), global structure model (GSM), and local-and-global centrality (LGC).

The following three metrics are used to evaluate the proposed SLGC node centrality metric.

Differentiation This reflects the ability of the method to distinguish the node influence ranking, mainly including the monotonicity function, the distinct metric based evaluation criterion, and the complementary cumulative distribution function.

Accuracy This reflects that the node ranking obtained by the method can be more conducive to the dissemination of information and mining the nodes with strong dissemination ability in the network.

Efficiency This reflects the network running time of the algorithm in identifying important nodes.

4.1. Datasets

In this experiment, six real-world networks with different properties are selected, including Jazz,[33] EEC,[34] Email,[35] Hamster,[36] Facebook,[37] and PGP.[38] As Table 6 shows the relevant statistical characteristics of the six real networks, where $|N|$ and $|M|$ are the number of edges and nodes, $\langle K\rangle$ is the average degree. The last two columns indicate the maximum degree $K_{\max}$ and average cluster coefficients $\langle C\rangle$ of the networks.

Table 6. The statistical features of six real-world networks.

Networks	$\|N\|$	$\|M\|$	$\langle K\rangle$	$K_{\max}$	$\langle C\rangle$
Jazz	198	2742	27.68	100	0.520
EEC	986	16064	32.584	273	0.4505
Email	1133	5451	9.622	71	0.2202
Hamster	2426	16631	13.711	273	0.5376
Facebook	4039	88234	43.691	1045	0.6638
PGP	10680	24316	25.316	205	0.2659

4.2. Differentiation

4.2.1. Monotonicity based evaluation criterion

In this experiment, the discriminative power of the proposed method versus other methods on the generated ranking results is studied by using the monotonicity function[39]

$$M(R)=\left[1-\frac{\sum\limits_{r\in R}|V|_r\times(|V|_r-1)}{|V|\times(|V|-1)}\right]^2, \tag{18}$$

where R is the ranked list of node importance generated by different methods, $|V|_r$ denotes the number of nodes assigned to the ranked list r, and $|V|$ denotes the total number of nodes in the network. If the ranking list R has better discriminative power, then a higher value indicates that the method has better node ranking performance.

Table 7. The M values of different methods on six real-world networks.

Network	DC	BC	CC	EC	PR	CLD	GSM	LGC	SLGC
Jazz	0.9659	0.9885	0.9878	0.9992	0.9993	0.9992	0.9992	0.9992	1
EEC	0.9571	0.9551	0.9982	0.9998	0.9998	0.9997	0.9998	0.9998	0.9999
Email	0.8873	0.9400	0.9988	0.9998	0.9998	0.9974	0.9998	0.9998	0.9999
Hamster	0.8979	0.7123	0.9850	0.9856	0.9622	0.9836	0.9857	0.9856	0.9999
Facebook	0.9739	0.9855	0.9966	0.9998	0.9998	0.9998	0.9998	0.9998	0.9999
PGP	0.6193	0.5118	0.9996	0.9997	0.9997	0.9641	0.9997	0.9997	0.9998

Table 8. DM values of different methods on six real-world networks.

Network	DC	BC	CC	EC	PR	CLD	GSM	LGC	SLGC
Jazz	0.3131	0.8939	0.6414	0.9646	0.9696	0.9646	0.9646	0.9646	1
EEC	0.1419	0.8488	0.6855	0.9756	0.9756	0.9513	0.9756	0.9766	0.9797
Email	0.0423	0.8181	0.7405	0.9761	0.9761	0.7466	0.9744	0.9761	0.9779
Hamster	0.0457	0.5193	0.5539	0.6838	0.6838	0.5746	0.6846	0.6859	0.9901
Facebook	0.0562	0.8695	0.3005	0.9571	0.9581	0.9408	0.9160	0.9546	0.9801
PGP	0.0077	0.3038	0.6617	0.7486	0.7470	0.2144	0.7706	0.7890	0.8415

Table 7 shows the monotonicity values of eight different node centrality measures for six real complex networks. The monotonicity value of SLGC tends to be 1 on each network, which is significantly higher than other methods, verifying the goodness and monotonicity of the proposed method for node ranking performance.

4.2.2. The distinct metric based evaluation criterion

The distinct metric (DM) of nodes is another metric used to test the ability to distinguish nodes differentiation by performing statistics on the ranking results of nodes, defined as follows:[40]

$$\mathrm{DM}(R) = \frac{\text{number of nodes with distinct ranks}}{|V|}, \tag{19}$$

where R denotes the ranking list generated by different methods, $|V|$ denotes the total number of nodes in the network, and DM takes values in the range of $[0,1]$, the higher the value of DM indicates that the method has a better ability to differentiate the nodes in the network.

According to the display in Table 8, it can be concluded that the DM values of SLGC are higher than other methods on all six complex networks, which indicates that the nodes in the network are resolved to be assigned to different ranks and have high differentiation ability for nodes.

4.2.3. Complementary cumulative distribution function based evaluation criterion

The complementary cumulative distribution function (CCDF) further standardizes the node ranking distribution of the ranking methods by generating node rankings for different node centrality metrics to better assess the differences in node ranking lists between methods and the node importance capabilities. The mathematical definition of CCDF is as follows:[41]

$$\mathrm{CCDF}(R) = \frac{n - \sum_{i=1}^{R} n_i}{n}, \tag{20}$$

where R denotes the node ranking list and n denotes the number of network nodes. According to this formula, from the perspective of distinguishing nodes' differentiation ability, the more nodes are assigned different ranks for ranking the better the method can show and the stronger the method's ability to distinguish nodes, which helps to select the more effective top N nodes from the network node ranking for disseminating information; if each node is assigned a different rank, the CCDF curve result will gradually converge to zero with a smooth slope, and if all nodes are assigned the same rank, the CCDF curve result will rapidly converge to zero.

As Fig. 2 shows the CCDF curve results on six real network datasets, it can be concluded that due to the degree centrality assigning node ranking ranks according to node degrees, which leads to most nodes in the same rank, the corresponding CCDF curve results will also converge to zero rapidly; the CCDF curve graph of the SLGC algorithm proposed in this paper converges to zero gently, reflecting that the algorithm has the optimal node. The CCDF curve of the proposed SLGC algorithm tends to zero gently, which reflects that the algorithm has optimal node distribution and monotonicity performance, and better distinguishes the differences between node rankings.

4.3. Accuracy

The SIR model is a common mathematical model used to describe the spread and control of infectious diseases, which is applied to describe the spreading process of a disease in a closed population. In this experiment, the SIR epidemic transmission model is used to model the transmission of infectious

diseases for ranked lists generated by different methods, in order to identify the transmission capacity and efficiency of nodes in complex networks. In the SIR model, nodes are divided into three states: susceptible, infected, and recovered. A susceptible person can be infected by contacting with an infected person with a certain probability. The infected node remains infected for a certain period and then enters the recovered state with a certain recovery probability and no longer infects other nodes. The SIR model can be described by a set of differential equations, where S, I, and R denote the number of susceptible, infected, and recovered individuals, respectively. The differential equations of SIR are[42]

$$\begin{cases} \dfrac{\mathrm{d}s(t)}{\mathrm{d}(t)} = -\beta s(t) i(t), \\ \dfrac{\mathrm{d}i(t)}{\mathrm{d}(t)} = \beta s(t) i(t) - \mu i(t), \\ \dfrac{\mathrm{d}r(t)}{\mathrm{d}(t)} = \mu i(t), \end{cases} \tag{21}$$

where $s(t)$, $i(t)$, and $r(t)$ denote the number of nodes in susceptible, infected, and recovered states, respectively; β denotes the infection probability and μ denotes the recovery probability.

The performance of the proposed algorithm can be measured by the spreading capability of the selected initial nodes. Based on the SIR model, the network infection quantity $F(t)$ at time t is defined as the ratio of the sum of infected nodes and recovered nodes within a unit of time to the total number of nodes in the network. It is defined as follows:

$$F(t) = \frac{n_{\mathrm{I}}(t) + n_{\mathrm{R}}(t)}{N}, \tag{22}$$

where $F(t)$ represents the network infection level at time t, $n_{\mathrm{I}}(t)$ represents the number of infected nodes at time t, $n_{\mathrm{R}}(t)$ represents the number of recovered nodes at time t, and N represents the total number of nodes in the network.

In this experiment, the first ten nodes of each method are selected as the initially infected nodes, and the remaining nodes are in the susceptible state. During the simulation experiment, the node overview in the network is the sum of the number of infected nodes and the number of recovered nodes, and the infected nodes infect the surrounding nodes in the susceptible state with the infection probability β. The infection probability β is set to 0.1, and the recovery probability is set to μ as 1. The propagation time is denoted by t, and each experiment is run 100 times independently. As shown in Fig. 3, the number of infected nodes $F(t)$ is used as the size of the infection ability of the selected initial node in a specific period, and the larger $F(t)$ is in the same propagation time, the larger the size of the node's impact under the method, which further indicates the stronger own influence of the node's initial test infected node selected according to the method.

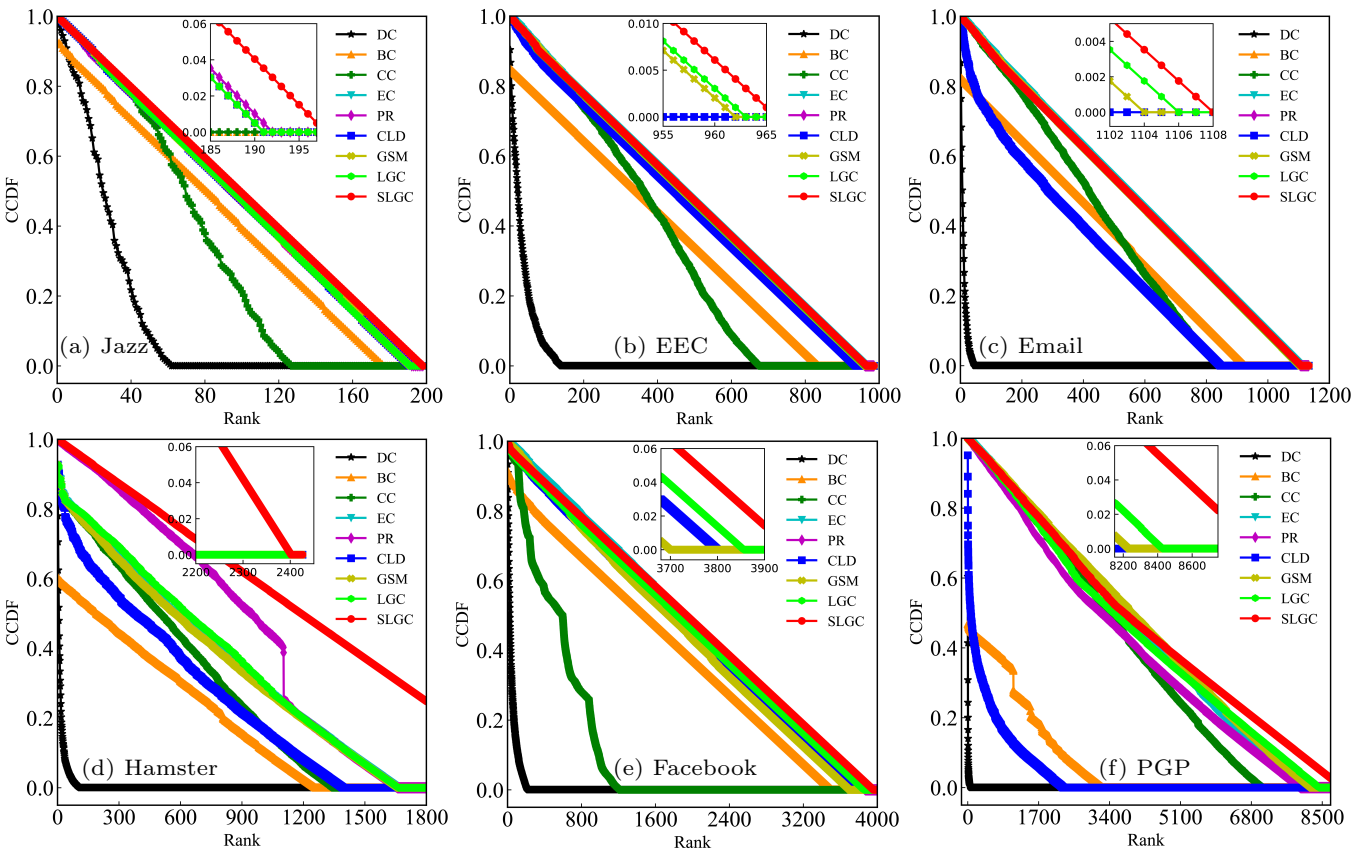


Fig. 2. The CCDF plots for ranking lists were obtained by different methods on six real-world networks.

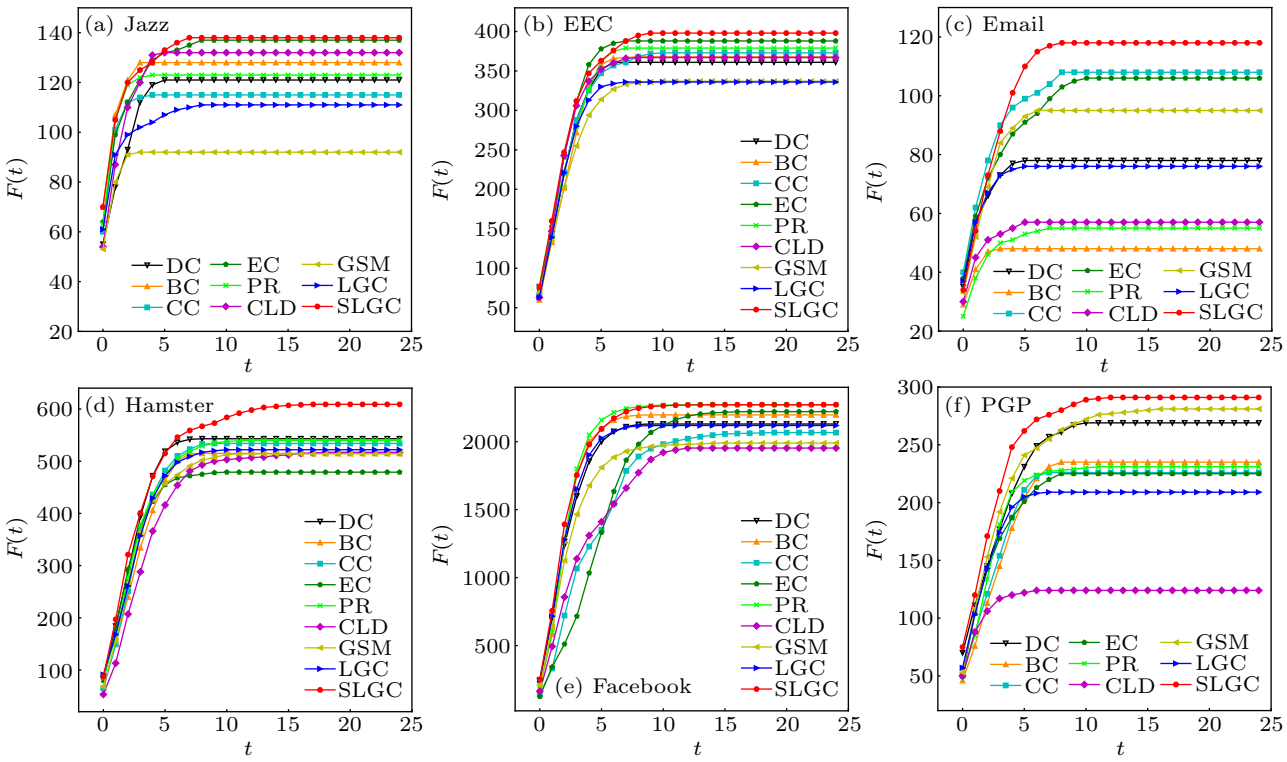


Fig. 3. The influence of propagation for the top-10 raking effect using SLGC and other benchmarks, where $F(t)$ represents the number of infected and recovered nodes at the time t in the range between 0 and 20.

4.4. Efficiency

In this experiment, the running time of each method on six real network datasets is investigated, and a shorter running time for a certain method indicates better efficiency. The efficiency of each method in identifying and ranking important nodes is reflected in terms of the running time. As shown in the figure, each node centrality measure is independently executed 100 times on each network, and the average run time is taken to quantify the efficiency of each method for node identification and ranking.

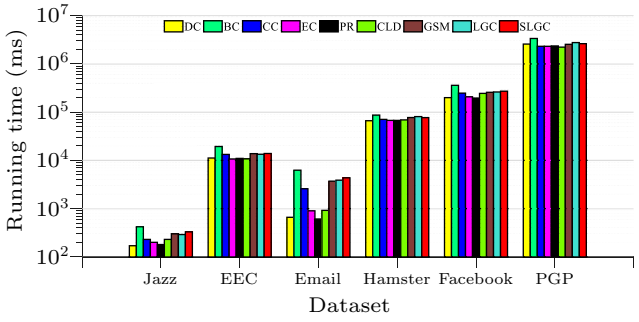


Fig. 4. Running time of different methods on six datasets.

As shown in Fig. 4, the SLGC algorithm takes less time to run than BC on all six datasets. On Jazz, EEC, and Email datasets, the SLGC algorithm takes the same amount of time as GSM and LGC, and on Facebook and PGP large-scale datasets, the SLGC algorithm not only takes less time than BC, but also takes less time than LGC algorithm. It is obvious that although DC takes little time to calculate node centrality, DC only considers the number of connected edges of nodes as an evaluation index and has a lower monotonicity value, while BC algorithm is a slower algorithm on most data and data pairs because it considers the shortest path and number of nodes in the data. As a whole, the SLGC algorithm increases the computing time relatively by a small amount on the basis of improving the accuracy of the node centrality metric; therefore, in terms of network running time, the SLGC algorithm is in a relatively intermediate position among all methods.

5. Conclusion

Mining and ranking of important nodes of complex networks has become one of the research hotspots of complexity theory. In this paper, we propose a new node centrality measurement method, the SLGC algorithm. In response to the limitations of traditional classical algorithms that only consider single-dimensional node features and the inaccurate results of existing node centrality measurement algorithms, we adopt the idea of multi attribute decision making for node importance ranking. We explore influential nodes in complex networks by considering the node's own attributes, the quantity and quality of neighboring nodes within its local neighborhood, and introducing information entropy to study the node's local influence. Additionally, we calculate the global influence through K-shell decomposition and the shortest paths between nodes.

By applying the SLGC algorithm to six different types and scales of real-world complex networks, we fully demonstrate its effectiveness in partitioning nodes into different ranks

and better distinguish the influence of nodes from three aspects: monotonicity value, ranking value, and complementary cumulative distribution function. Furthermore, we validate the effectiveness of the proposed algorithm by simulating the spread of nodes in practical scenarios using the SIR model. The proposed method measures the importance of nodes from multiple dimensions and statistical features, and its time complexity is intermediate among node centrality measurement methods. This method improves the ranking of node importance in complex networks while ensuring network operational efficiency, and it can be applied to complex networks of various sizes and scales.

Experimental results show that the proposed method effectively differentiates and propagates nodes, outperforming existing benchmark methods. However, the current SLGC algorithm is limited to unweighted networks. Therefore, in future research, we will incorporate the study of weighted networks into the proposed method, which could present one of the challenges for developing a more accurate and widely applicable algorithm for ranking the importance of nodes in complex networks.

Acknowledgments

Project supported by the Natural Science Basic Research Program of Shaanxi Province of China (Grant No. 2022JQ-675) and the Youth Innovation Team of Shaanxi Universities.

References

[1] Boccaletti S, Latora V, Moreno Y, Chavez M and Hwang D U 2006 *Phys. Rep.* **424** 175
[2] Roberto M V 2022 *J. Metaverse* **2** 16
[3] Bai Y G, Liu S Y, Li Q and Yuan J 2021 *IEEE Trans. Syst. Man Cyber. Syst.* **52** 3378
[4] Huang H M, Shen H, Meng Z Q, Chang H J and He H W 2019 *Appl. Intell.* **49** 2137
[5] Ai S, Hong S, Zheng X Y, Wang Y and Liu X Z 2021 *Int. J. Intell. Syst.* **36** 1903
[6] Yao S Y, Fan N and Hu J 2022 *Optim. Lett.* **16** 1563
[7] Gupta M and Kumar P 2020 *Eur. J. Oper. Res.* **284** 660
[8] Zhao J, Wang Y C and Deng Y 2020 *Chaos Soliton Fractals* **133** 109637
[9] Bardoscia M, Barucca P, Battiston S, Caccioli F, Garlaschelli D, Saracco F, Squartini T and Caldarelli G 2021 *Nat. Rev. Phys.* **3** 490
[10] Alp Z Z and Öğüdücü Ş G 2019 *Knowl-Based Syst.* **163** 944
[11] Ding R, Ujang N, Hamid H B, Manan M S A, Li R, Albadareen S S M, Nochian A and Wu J J 2019 *Netw. Spat. Econ.* **19** 1281
[12] Ait Rai K, Machkour M and Antari J 2023 *Beni-Suef University J. Basic Appl. Sci.* **12** 18
[13] Freeman L C 1978 *Soc. Netw.* **1** 215
[14] Lü L, Zhou T, Zhang Q M and Stanley H E 2016 *Nat. Commun.* **7** 10168
[15] Wen T, Pelusi D and Deng Y 2020 *Knowl-Based Syst.* **195** 105717
[16] Ding L H, Hong L, Sun B and Shi P 2019 *Acta Phys. Sin.* **68** 128902 (in Chinese)
[17] Opsahl T, Agneessens F and Skvoretz J 2010 *Soc. Netw.* **32** 245
[18] Sabidussi G 1966 *Psychometrika* **31** 581
[19] Luan Y Y, Bao Z K and Zhang H F 2021 *J. Syst. Sci. Complex.* **34** 2168
[20] Bonacich P and Lloyd P 2001 *Soc. Netw.* **23** 191
[21] Brin S and Page L 1998 *Comput. Netw. ISDN Syst.* **30** 107
[22] Curado M, Tortosa L and Vicent J F 2023 *Inf. Sci.* **628** 177
[23] Kitsak M, Gallos L K, Havlin S, Liljeros F, Muchnik L, Stanley H E and Makse H A 2010 *Nat. Phys.* **6** 888
[24] Maji G, Dutta A, Malta M C and Sen S 2021 *Expert Syst. Appl.* **179** 115061
[25] Yan X L, Cui Y P and Ni S J 2020 *Chin. Phys. B* **29** 048902
[26] Du Y X, Gao C, Hu Y, Mahadevan S and Deng Y 2014 *Physica A* **395** 549
[27] Wang Y, Li H Z, Zhang L, Zhao L L and Li W L 2022 *Chaos Soliton Fractals* **162** 112513
[28] Qiu L Q, Zhang J Y and Tian X B 2021 *Appl. Intell.* **51** 4394
[29] Li M T, Zhang R S, Hu R J, Yang F, Yao Y B and Yuan Y N 2018 *Int. J. Mod. Phys. B* **32** 1850118
[30] Ullah A, Wang B, Sheng J F, Long J and Sun Z J 2021 *Sci. Rep.* **11** 6173
[31] Ullah A, Wang B, Sheng J F, Long J, Khan N and Sun Z J 2021 *Expert Syst. Appl.* **186** 115778
[32] Shannon C E 1948 *Bell Syst. Tech. J.* **27** 379
[33] http://konect.cc/networks/arenas-jazz/
[34] https://networkrepository.com/email-EU.php
[35] http://konect.cc/networks/arenas-email/
[36] http://konect.cc/networks/petster-hamster/
[37] https://snap.stanford.edu/data/ego-Facebook.html
[38] http://konect.cc/networks/arenas-pgp/
[39] Bae J and Kim S 2014 *Physica A* **395** 549
[40] Gao S, Ma J, Chen Z M, Wang G H and Xing C M 2014 *Physica A* **403** 130
[41] Pianosi F and Wagener T 2015 *Environ. Model. Softw.* **67** 1
[42] Pastor-Satorras R and Vespignani A 2001 *Phys. Rev. E* **63** 066117