

Article

The Self-Information Weighting-Based Node Importance Ranking Method for Graph Data

Shihu Liu  and Haiyan Gao *

School of Mathematics and Computer Sciences, Yunnan Minzu University, Kunming 650504, China

* Correspondence: 21213037550007@ymu.edu.cn

Abstract: Due to their wide application in many disciplines, how to make an efficient ranking for nodes, especially for nodes in graph data, has aroused lots of attention. To overcome the shortcoming that most traditional ranking methods only consider the mutual influence between nodes but ignore the influence of edges, this paper proposes a self-information weighting-based method to rank all nodes in graph data. In the first place, the graph data are weighted by regarding the self-information of edges in terms of node degree. On this base, the information entropy of nodes is constructed to measure the importance of each node and in which case all nodes can be ranked. To verify the effectiveness of this proposed ranking method, we compare it with six existing methods on nine real-world datasets. The experimental results show that our method performs well on all of these nine datasets, especially for datasets with more nodes.

Keywords: graph data; information entropy; node importance ranking; self-information weighting



Citation: Liu, S.; Gao, H. The Self-Information Weighting-Based Node Importance Ranking Method for Graph Data. *Entropy* **2022**, *24*, 1471. <https://doi.org/10.3390/e24101471>

Academic Editor: Luis Javier Garcia Villalba

Received: 3 September 2022

Accepted: 11 October 2022

Published: 15 October 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Node importance ranking [1] aims to construct a suitable score function for all nodes, in which case all nodes can be ranked with the help of this score function. Especially in recent years, with the popularity of graph data, the problem of node importance ranking for graph data has been widely studied and applied in many fields, such as blocking rumors [2,3], disease detection [4,5], information transmission [6,7], and so on.

To date, for the problem of node importance ranking, the methods of constructing score function can be roughly classified into three categories, which are local-information-based score functions [8,9], global-information-based score functions [10,11] and node-position-based score functions [12,13]. The local-information-based score functions mainly consider the local topology information of node itself and neighbors. Although they have low complexity, the accuracy of the rank result is also low. The global-information-based score functions usually need to traverse the entire graph data, so they might have the expensive time costs and cannot be directly applied to large-scale graph data. The node-position-based score functions are relatively rare, because these methods usually assign the same score to a large number of nodes and cannot accurately identify their importance.

Bearing what was discussed above in mind, plenty of methods have been proposed and investigated carefully. For instance, the degree centrality [14] method constructed the simplest local-information-based score function. It defined the importance of nodes as the number of neighbors, which reflected the direct influence of a node on others. Zhang et al. [15] analogized the problem of node importance ranking to the voting process based on the degree of neighbors. The eigenvector centrality [16] method determines the importance of nodes by taking the eigenvalues and eigenvectors of adjacency matrix into consideration, which constructs a global-information-based score function. Fu et al. [17] constructed the two-step framework that combines the global information and local topology features to identify influential nodes. The closeness centrality [18] method quantifies the importance of nodes by calculating the average distance from one node to all other

nodes. The betweenness centrality [19] method characterizes node importance as the number of shortest paths through the node. The more times a node acts as the bridge, the more important it is. The K-shell decomposition centrality [20] method recursively deletes nodes in the outer layer of the graph data. It considers that the nodes at the core of graph data have strong influence. The PageRank method [21] was applied in the Google search engine, and considers each web page as a node and hyperlinks between pages as edges. The importance of node in PageRank method depends on the importance of other nodes pointed to this node. Wang et al. [22] proposed a label propagation algorithm based on the similarity to identify the influential node. The problem of node importance ranking is regarded as a multi-attribute decision making problem in reference [23], which can take many factors that affect the importance of nodes into account.

Besides the above mentioned, the theory of entropy has been used by many researchers to deal with the problem of node importance ranking [24–26]. For example, Guo et al. [27] proposed the VoteRank algorithm, which introduced information entropy as the influence of node on its neighbors. Zareie et al. [28] used information entropy while considering the degree distribution of first-order neighbors and second-order neighbors of nodes. Based on the hypothesis that the removal of a more important node is likely to cause more structural variation, entropy variation [29] is proposed to study the problem of node importance ranking. The local structure entropy approach [30], proposed by Lei et al., comprehensively considers the relationship between a node's Tsallis entropy and its neighbors. Fei et al. [31] proposed a novel method to identify influential nodes using relative entropy and TOPSIS method, which combines the advantages of existing centrality measures.

Although the accuracy of rank results can be improved with the help of entropy, most of these methods only consider the mutual influence between nodes and ignore the influence of edges that directly connected to the node itself [32–34]. Certainly, as the important component of graph data, the information contained in the edges can make a huge influence for the final ranking [35]. Therefore, how to measure the amount of information contained in the edges and make full use of them is vital.

Inspired by the studies mentioned above, in this paper, we will still study the problem of node importance ranking for graph data. However, here we pay attention to the edge and propose a self-information weighting-based node importance ranking method. In summary, this paper makes the following contributions:

- The graph data are weighted by regarding the self-information of edges in terms of the node degree.
- The information entropy of nodes is constructed to measure the importance of each node. What is more, the rank result can be obtained according to the value of the information entropy.
- Nine real-world datasets are used to show the validity of the self-information weighting-based node importance ranking method for graph data. The experimental results manifest that our method has great advantage in terms of monotonicity, node distribution and accuracy.

The remainder of this paper is organized as follows. Section 2 makes a brief review of some basic knowledge. Section 3 introduces the proposed node importance ranking method, i.e., the self-information weighting-based node importance ranking method. Section 4 is composed of three parts, which are experimental platform, datasets description and evaluation criteria. Section 5 shows the detailed comparison between the proposed node importance ranking method and some existing ranking methods on nine real-world datasets. Section 6 concludes this paper and also makes a possible direction for future research.

2. Preliminaries

In this section, we propose some basic concepts that are closely related to the work of this article, such as graph data and the benchmark methods of how to rank the nodes. For more detailed description, one can refer to the Refs. [36–39].

2.1. Graph Data

Mathematically, the so-called graph data can be expressed as a tuple $G = (V, E)$, where

- $V = \{v_1, v_2, \dots, v_n\}$ is the collection of nodes and n represents the number of nodes.
- $E = \{(v_i, v_j) | v_i, v_j \in V\}$ is the collection of edges, in which case $(v_i, v_j) \in E$ means that there is an edge between nodes v_i and v_j . As that of V , we apply m , i.e., $|E| = m$, to denote the number of edges.

Without loss of generality, in this paper we adhere to the hypothesis that the graph data $G = (V, E)$ is an undirected and unweighted graph data. In other words, $(v_i, v_j) = (v_j, v_i)$ for any $v_i, v_j \in V$. In addition, adjacency matrix of graph data $G = (V, E)$ can be expressed as a matrix

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix},$$

where a_{ij} represents the connectivity between nodes $v_i, v_j \in V$, for $i, j = 1, 2, \dots, n$. Obviously, $a_{ij} = 1$ if and only if $(v_i, v_j) \in E$, otherwise $a_{ij} = 0$.

2.2. Benchmark Methods for Node Importance Ranking

The key step of node importance ranking is to construct a suitable score function for all nodes, in which case all nodes can be ranked with the help of this proposed score function. At present, the existing methods of constructing score function can be divided into three categories: the local-information-based score function, the global-information-based score function and the node-position-based score function.

2.2.1. The Local-Information-Based Score Function

The degree centrality method, abbreviated to DC for convenience, takes the number of neighbor nodes into account to quantify the importance of the node, and the mathematical expression of it can be expressed as

$$DC(v_i) = \sum_{j=1}^n a_{ij}. \quad (1)$$

The mutual information method, abbreviated to MI for convenience, defines the mutual information between any two connected nodes as

$$I(v_i, v_j) = \begin{cases} \ln\left(\frac{d_i}{d_j}\right) & a_{ij} = 1 \\ 0 & a_{ij} = 0 \end{cases}, \quad (2)$$

where $d_i = \sum_{j=1}^n a_{ij}$ is the degree of node v_i . On this base, the amount of information of v_i can be defined as

$$MI(v_i) = \sum_{v_j \in N(v_i)} I(v_i, v_j), \quad (3)$$

where $N(v_i) = \{v_j | (v_i, v_j) \in E\}$ is the set of neighbors of node v_i .

2.2.2. The Global Information Based Score Function

The closeness centrality method, abbreviated to CC for convenience, defines the importance of nodes as the reciprocal of the average length from one node to all other nodes in graph datum $G = (V, E)$. The corresponding computing formula is

$$CC(v_i) = \frac{n-1}{\sum_{i \neq j} d(i, j)}, \quad (4)$$

where $d(i, j)$ shows the length of the shortest path from node v_i to v_j . If there is no path from v_i to v_j , then $d(i, j) = 0$.

The eigenvector centrality method, abbreviated to EC for convenience, determines the importance of nodes by taking the eigenvalues and eigenvectors of A , the adjacency matrix of $G = (V, E)$, into consideration. The calculation formula is defined as

$$EC(v_i) = \lambda^{-1} \sum_{j=1}^n a_{ij} e_j, \quad (5)$$

where λ is the largest eigenvalue, and e_j for $j = 1, 2, \dots, n$ is the j th eigenvector of A .

2.2.3. The Node Position Based Score Function

The K-shell decomposition method, abbreviated to KS for convenience, evaluates the importance of nodes by sequentially removing nodes in the outer layer of the graph data. The main principle of it is to sign the node with original degree at first, nodes with degree 1 are removed, and this process continues until there are no nodes with degree 1 in the graph data. The importance of all these removed nodes is labeled as 1. Next, nodes with degree 2 are removed. The process continues until there are no nodes with degree less than or equal to 2 in the graph data. Similarly, the importance of these removed nodes is labeled as 2. In addition, nodes with degree 3, 4, \dots , are removed and labeled until all nodes are completed.

The improved K-shell decomposition method, abbreviated to IKS for convenience, only removes nodes with the lowest degree in the graph data each time, which is the biggest difference between the IKS method and KS method. That is to say, the selection of nodes to remove each time is not necessarily in an increasing sequence of degrees, i.e., 1, 2, \dots . For example, if all nodes with degree 2 in the graph data are removed in the last iteration and their importance is labeled as 2, but nodes with degree 1 appear in the rest of graph data, these nodes will be removed in the next iteration, and their importance will be labeled as 3. This process continues until all nodes are completed.

3. Proposed Method

As can be seen from foregoing discussion, most traditional node importance ranking methods only consider the mutual influence between nodes, while ignoring the influence of edges that directly connected to the node itself. For example, the DC method simply regards the number of neighbors as the importance of nodes. In fact, since each neighbor node has different local topology information, their contributions are not equal. Certainly, the DC method does not distinguish between the contributions of different neighbor nodes, which will lead to unsatisfactory rank results. Bearing this in mind, in what follows, we construct a new method to rank the nodes in graph data $G = (V, E)$. Distinguished from the existing ranking methods, the proposed node ranking method will start from the perspective of edges. Herein, the self-information is regarded as the weight of edges, and it turns an unweighted graph datum into a weighted graph datum. The contribution of neighbor nodes can be distinguished by using different weight values of edges. In this case, the score function used to measure the importance of each node is determined by considering the information entropy of nodes.

3.1. Edge Weight Construction in Terms of Self-Information

The self-information proposed by Shannon [40] is usually used to measure the amount of information of a event. Given that $X = \{x_1, x_2, \dots, x_n\}$ is a discrete random variable and its probability distribution is expressed as $P = \{p_1, p_2, \dots, p_n\}$, then the self-information of each event $x_i \in X$ can be expressed as

$$W(x_i) = -\log_2 p_i. \tag{6}$$

The self-information indicates that the amount of information contained in a basic event is inversely proportional with its probability of occurrence. In other words, frequent events usually contain less information. Conversely, events that occur less often contain huge amounts of information. Taking node v_i for example, the nodes that have edges connected to v_i are much smaller than those without edges connected to v_i in the whole network. According to the definition of self-information, these edges contain more valuable information. Therefore, we construct the weight of these edges with the help of the self-information.

Certainly, the degree of any two nodes, taking $v_i, v_j \in V$ for example, can be applied to depict the information of corresponding edge $(v_i, v_j) \in E$ to some extent. Even more, the amount of information obtained from this can be used to describe the weight of corresponding edge. Bearing what was discussed above in mind, we have that the probability corresponding to any $(v_i, v_j) \in E$ can be defined as

$$P(v_i, v_j) = \frac{1}{d_i d_j}, \tag{7}$$

where d_i is the degree of v_i , the same as that of d_j .

To this, the self-information of edge (v_i, v_j) is equivalent to its weight, in which case it can be calculated by the following equation

$$\begin{aligned} W(v_i, v_j) &= -\log_2 P(v_i, v_j) \\ &= \log_2(d_i d_j). \end{aligned} \tag{8}$$

Obviously, we can find that for the graph data $G = (V, E)$, it is easy to obtain the equation

$$W(v_i, v_j) = W(v_j, v_i). \tag{9}$$

The reason of it is that on one hand, the Equation (9) can be obtained from Equation (8). On the other hand, as $G = (V, E)$ is an undirected graph data, $(v_i, v_j) \in E$ if and only if $(v_j, v_i) \in E$, and both of edges (v_i, v_j) and (v_j, v_i) should have the same self-information.

3.2. Node Importance Induced by Information Entropy

Given that X is a random variable and its corresponding probability distribution is $\vec{P} = (p_1, p_2, \dots, p_n)$, if we let $W(X) = (W(x_1), W(x_2), \dots, W(x_n))^T$, then we have the following equation based on Equation (6).

$$\begin{aligned} \vec{P} \cdot W(X) &= (p_1, p_2, \dots, p_n) \begin{pmatrix} W(x_1) \\ W(x_2) \\ \vdots \\ W(x_n) \end{pmatrix} \\ &= -\sum_{i=1}^n p_i \log_2 p_i. \end{aligned} \tag{10}$$

Up to now, $\vec{P} \cdot W(X)$, abbreviated to $E(X)$ for convenience, can be regarded as the expected value of self-information. According to Equation (6), the negative log of probability represents the amount of information contained in a basic event, i.e., the self-

information. The expected value of the amount of information contained in all basic events is called information entropy. In other words, it can be applied to quantify the amount of information contained in the random variable X . Herein, we use information entropy to quantify the importance of nodes mainly because of the special properties of information entropy. Following the ideology of the above Equation (10), the properties of information entropy are listed as follows.

Property 1. *Given that X is a random variable and its corresponding probability distribution is \vec{P} , then we have that $E(X)$ reaches the maximum when \vec{P} is an uniform distribution.*

Proof. Obviously, for all $p_i \in \vec{P}$, one has that the following constraint

$$\sum_{i=1}^n p_i = 1 \tag{11}$$

is correct. With Equations (10) and (11), we construct the Lagrange function as

$$\begin{aligned} L(p_1, p_2, \dots, p_n, \lambda) &= E(X) + \lambda \left(\sum_{i=1}^n p_i - 1 \right) \\ &= - \sum_{i=1}^n p_i \log_2 p_i + \lambda \left(\sum_{i=1}^n p_i - 1 \right). \end{aligned} \tag{12}$$

By considering the partial derivative of each variable p_i , then let all of them be equal to zero. With this operation, one can have that

$$p_i = e^{\lambda-1} \quad (i = 1, 2, \dots, n). \tag{13}$$

With the help of Equations (11) and (13), the following result

$$p_i = \frac{1}{n} \quad (i = 1, 2, \dots, n) \tag{14}$$

comes naturally. Once n is fixed, the probability distribution \vec{P} will be an uniform distribution, in which case $\vec{P} \cdot W(X)$ reaches the maximum.

This completes the proof. \square

Property 2. *$E(X)$ is an increasing function with respect to the independent variable n which represents the number of basic events.*

Proof. With Equation (10), for any positive integer k , we have

$$\begin{aligned} E(X_{k+1}) - E(X_k) &= - \sum_{i=1}^{k+1} p_i \log_2 p_i + \sum_{i=1}^k p_i \log_2 p_i \\ &= - p_{k+1} \log_2 p_{k+1}. \end{aligned} \tag{15}$$

As $0 \leq p_i \leq 1$, for $i = 1, 2, \dots, k + 1$, then $\log_2 p_{k+1} \leq 0$, which will lead to the fact that

$$- p_{k+1} \log_2 p_{k+1} \geq 0. \tag{16}$$

That is $E(X_{k+1}) \geq E(X_k)$.

This completes the proof. \square

It can be found easily that the above properties are also true for a given graph data $G = (V, E)$. Because in the aspect of a node's degree, once d_i , take $v_i \in V$, for example, is

greater than that of any $v_j \in V$ for $i \neq j$, the importance of node v_i is greater than any other node. Furthermore, a node will have greater importance if its neighbors have uniform degree distribution [28]. On these bases, in what follows, we try to use information entropy, i.e., Equation (10), to determine the node importance in a whole new perspective.

Before giving the score function to measure the node importance, at first we propose two notations, $W(v_i)$ and $W^+(v_i)$. Take v_i for example:

- $W(v_i)$ represents the sum of self-information of edges with v_i as one of its endpoint. In mathematical form, it takes the calculation form

$$W(v_i) = \sum_{v_j \in N(v_i)} W(v_i, v_j). \tag{17}$$

- $W^+(v_i)$ represents the sum of self-information of edges that v_i and its neighbors are one endpoint of these edges, and it has the following calculation formula:

$$W^+(v_i) = \sum_{v_j \in \Gamma(v_i)} W(v_j), \tag{18}$$

where $\Gamma(v_i) = N(v_i) \cup \{v_i\}$.

Obviously, $W(v_i)$ reflects the influence of edges directly connected to v_i , while $W^+(v_i)$ takes edges related to neighbors into account. Based on these discussions, the probability corresponding to any $v_j \in \Gamma(v_i)$ can be defined as

$$P(v_j) = \frac{W(v_j)}{W^+(v_i)}. \tag{19}$$

One can find that this definition satisfies the condition that the sum of probabilities is equal to 1, that is

$$\sum_{v_j \in \Gamma(v_i)} \frac{W(v_j)}{W^+(v_i)} = 1. \tag{20}$$

To this, the information entropy of node v_i , for $i = 1, 2, \dots, n$, can be determined by the following equation:

$$E(v_i) = - \sum_{v_j \in \Gamma(v_i)} P(v_j) \log_2 P(v_j). \tag{21}$$

On one hand, the information entropy can be used to quantify the amount of information contained in a random variable. On the other hand, the amount of information contained in nodes is inseparable from edges in graph data. Therefore, we can use $E(v_i)$ which combines information entropy and edge weights as a suitable score function for each node.

Example 1. To make it easy to understand how to calculate the information entropy of each node, in what follows, we apply a simple graph data shown in Figure 1 to describe the whole process in detail.

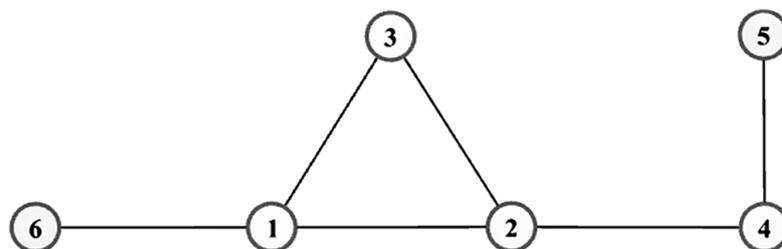


Figure 1. A simple graph data with $n = 6$ and $m = 6$.

With Equation (8), the weight of each existing edge can be determined. The results are listed in Table 1.

Table 1. The weight of each existing edge.

Edge	Weight
(v_1, v_2)	3.1699
(v_1, v_3)	2.5850
(v_1, v_6)	1.5850
(v_2, v_3)	2.5850
(v_2, v_4)	2.5850
(v_4, v_5)	1.0000

Taking node v_1 for example, the value of $W(v_1)$, $W^+(v_1)$ and $E(v_1)$ could be obtained by Equations (17), (18) and (21), which are

$$\begin{aligned}
 W(v_1) &= W(v_1, v_2) + W(v_1, v_3) + W(v_1, v_6) \\
 &= 7.3399,
 \end{aligned}
 \tag{22}$$

$$\begin{aligned}
 W^+(v_1) &= W(v_1) + W(v_2) + W(v_3) + W(v_6) \\
 &= 22.4348
 \end{aligned}
 \tag{23}$$

and

$$\begin{aligned}
 E(v_1) &= - \sum_{v_j \in \Gamma(v_1)} \frac{W(v_j)}{W^+(v_1)} \log_2 \frac{W(v_j)}{W^+(v_1)} \\
 &= 1.8161,
 \end{aligned}
 \tag{24}$$

the same as that of v_2, v_3, v_4, v_5 and v_6 . All in all, the information entropy for all nodes can be calculated and here we list it in Table 2.

For nodes $v_i, v_j \in V$, if $E(v_i) \geq E(v_j)$, then the importance rank result can be expressed as $v_i \succcurlyeq v_j$, and otherwise, it can be expressed as $v_i \prec v_j$ or $v_j \succ v_i$. As can be seen from Table 2, one has that $E(v_2) > E(v_1) > E(v_3) > E(v_4) > E(v_5) > E(v_6)$, then these six nodes can be ranked as $v_2 \succ v_1 \succ v_3 \succ v_4 \succ v_5 \succ v_6$.

Table 2. The information entropy of each node.

Node	$W(v_i)$	$W^+(v_i)$	$E(v_i)$
v_1	7.3399	22.4348	1.8161
v_2	8.3399	24.4348	1.9309
v_3	5.1700	20.8498	1.5579
v_4	3.5850	12.9249	1.2067
v_5	1.0000	4.5850	0.7567
v_6	1.5850	8.9249	0.6748

3.3. Summary of Algorithm

In this part, we give the detailed process of our proposed node importance ranking method. For convenience, in what follows, we apply SIWR to represent the proposed method. The input of the algorithm is a graph data $G = (V, E)$ with n nodes and m edges, and its output is the possible rank result, such as $v_{i1} \succcurlyeq v_{i2} \succcurlyeq \dots \succcurlyeq v_{in}$.

The construction of Algorithm 1 is operated in the following three phases: weight computation (lines 2–4), information entropy computation (lines 5–9) and nodes ranking (lines 10–16).

Algorithm 1: The construction procedure of SIWR algorithm

```

input : Graph data  $G = (V, E)$ .
output: Possible rank result  $v_{i1} \succ v_{i2} \succ \dots \succ v_{in}$ .

1 begin
2   for  $(v_i, v_j) \in E$  do
3     | Compute the edge's weight  $W(v_i, v_j)$  by Equation (8);
4   end
5   for  $v_i \in V$  do
6     | Compute  $W(v_i)$  by Equation (17);
7     | Compute  $W^+(v_i)$  by Equation (18);
8     | Compute the information entropy  $E(v_i)$  by Equation (21);
9   end
10  for  $v_i, v_j \in V$  do
11    | if  $E(v_i) \geq E(v_j)$ 
12    |    $v_i \succ v_j$ ;
13    | else
14    |    $v_i \prec v_j$ ;
15    | end
16  end
17  return  $v_{i1} \succ v_{i2} \succ \dots \succ v_{in}$ .           /*  $v_{ij} \in V$ , for  $j = 1, 2, \dots, n$ 
18 end
  */

```

4. Experimental Construction

In this section, we prepare the experimental environment, such as the experimental platform, experimental datasets and evaluation criteria.

4.1. Experimental Platform

The algorithm development platform is the following: MATLAB 2018a. The computer configuration is the following: Intel(R)Core(TM)i5-8250U CPU and the 64-bit Windows 10 operation system. For ease of reading, the detailed information is listed in Table 3.

Table 3. The experimental platform.

Parameter	Parameter Value
RAM	8 GB
Speed	1.8 GHz
Operation system	Windows 10
Operation programing	MATLAB R2018a
CPU	Intel(R)Core(TM)i5-8250U

4.2. Datasets Description

In this article, we perform our experiment with the following nine real-world datasets that can be downloaded from the corresponding academic website <http://konect.cc/networks/> (accessed on 1 September 2022). The detailed information of related datasets is given below.

- **Karate:** The social network of friendships between 34 members of a karate club at a US university in the 1970s.
- **Dolphins:** The social network of frequent associations between 62 dolphins living in New Zealand.
- **Polbooks:** The network is made up of books about US politics published in 2004.

- **Adjnoun:** The network of co-words between adjectives and nouns commonly used in the novel "David Copperfield".
- **Football:** The network of US football games between division IA colleges.
- **Jazz:** The collaborative network between jazz musicians.
- **Netscience:** The collaborative network of scientists who have published papers in the field of network science.
- **Email:** The interactive network of emails among members of the University of Rovira.
- **Friendships:** The network contains friendships between users of the website.

The topological statistical characteristics of the above datasets are listed in Table 4. Therein, each row from left to right is the name of datasets, number of nodes n , number of edges m , average degree $\langle d \rangle$, maximum degree d_{max} and clustering coefficient cc .

Table 4. Topological statistical characteristics of the eight real-world graph data.

Dataset	n	m	$\langle d \rangle$	d_{max}	cc
Karate	34	78	4.5882	17	0.5879
Dolphins	62	159	5.1290	12	0.3030
Polbooks	105	441	8.4000	25	0.4875
Adjnoun	112	425	7.5893	49	0.1898
Football	115	613	10.6609	12	0.4032
Jazz	198	2742	27.6970	100	0.6334
Netscience	379	914	4.8232	34	0.7981
Email	1133	10903	9.6230	71	0.2550
Friendships	1858	12534	13.4919	272	0.1670

4.3. Evaluation Criteria

Here, we propose three evaluation criteria to evaluate the advantage and disadvantage of node importance ranking methods, which are the monotonicity-based evaluation criterion, complementary cumulative distribution function-based evaluation criterion and susceptible–infected–recovered epidemic-model-based evaluation criterion.

4.3.1. Monotonicity Based Evaluation Criterion

By taking the fact that a ranking method will be better if a few nodes are listed in the same order of consideration, the monotonicity relation [41] is applied to evaluate the discriminability of different methods, and the concrete formula is

$$M(R) = \left(1 - \frac{\sum_{r \in \Gamma} n_r(n_r - 1)^2}{n(n - 1)} \right)^2, \tag{25}$$

where R is the possible rank result, n_r represents the number of nodes that have been listed in the same order of R , and Γ is the index that represents the number of different orders. For example, if the rank result R is $v_1 \succ v_2 \approx v_3 \succ v_4 \succ v_5$, then $\Gamma = \{1, 2, 3, 4\}$, in which case nodes v_2 and v_3 are listed in the same order. To this, $n_1 = n_3 = n_4 = 1$ and $n_2 = 2$.

Obviously, the closer the value of $M(R)$ is to 1, the greater the monotonicity of the possible rank result [42]. When all nodes have a unique order, the value of $M(R)$ will be 1, and the possible rank result is completely monotonic.

4.3.2. Complementary Cumulative Distribution Function Based Evaluation Criterion

In addition to monotonicity, the complementary cumulative distribution function, abbreviated to CCDF for convenience, was utilized to further evaluate the ability that identify the importance of different nodes [43]. The mathematical formula of it is

$$CCDF(r) = \frac{n - \sum_{i=1}^r n_i}{n}. \quad (26)$$

Obviously, this formula can display the distribution of nodes in different orders. Having more nodes in the same order causes the function to rapidly drop to zero, while having fewer nodes in the same order will obtain a smoother descending slope.

4.3.3. Susceptible-Infected-Recovered Epidemic Model Based Evaluation Criterion

In order to assess the accuracy of SIWR method, we compare the possible rank result that generated by SIWR and other benchmark methods in terms of the susceptible–infected–recovered epidemic model, i.e., abbreviated to SIR for convenience [44,45].

Each node belongs to one of three states in SIR, which are susceptible, infected and recovered, respectively. At first, node $v_i \in V$ is selected as infected node, while others are in a susceptible state. After that, the infected node affects its neighbors with the infected probability $\beta = 1/(\langle d \rangle - 1)$, and then enters into a recovered state with the recovery probability $r = 1$. It should be pointed out that the infected probability and recovery probability have various forms in different articles. Here, we choose the same form as reference [46]. Finally, the total number of infected nodes is regarded as the propagation ability of node v_i when the whole process is finished. The stronger the propagation ability, the more important the node.

To increase accuracy, this process will be repeated hundreds of times, and the mean value will be considered the final result. Its mathematical expression is given as

$$F(t) = \frac{n_I}{N_{ite}}, \quad (27)$$

where n_I represents the total number of infected nodes and N_{ite} represents the number of repeated experiments.

5. Results Analysis

In this section, we conduct an experimental analysis of the SIWR method on nine real-world datasets. The concrete analysis includes monotonicity analysis, node distribution analysis, SIR analysis, robustness analysis and running time analysis. What is more, some comparing methods are used here to support the advantage of the SIWR method, which are the DC, MI, CC, EC, KS and IKS methods.

5.1. Monotonicity Analysis

By computing, the value of $M(R)$ with respect to the benchmark methods and SIWR method are listed in Table 5. Obviously, the ranking method SIWR shows excellent performance, especially on the Karate, Jazz, Netscience, Email and Friendships datasets. The interesting fact is that the good performance of the SIWR method increases along with the increasing of n , the nodes number of graph data.

Table 5. The monotonicity value of seven node importance ranking methods. The best results are highlighted in bold.

Dataset	M(DC)	M(MI)	M(CC)	M(EC)	M(KS)	M(IKS)	M(SIWR)
Karate	0.7079	0.9542	0.8993	0.9576	0.4958	0.6463	0.9577
Dolphins	0.8312	0.9905	0.9737	0.9979	0.3769	0.8841	0.9979
Polbooks	0.8252	1.0000	0.9846	1.0000	0.4949	0.8382	1.0000
Adjnoun	0.8661	0.9984	0.9837	0.9997	0.5990	0.8745	0.9997
Football	0.3636	0.9835	0.9488	1.0000	0.0003	0.9419	1.0000
Jazz	0.9659	0.9993	0.9878	0.9994	0.7944	0.9383	0.9995
Netscience	0.7642	0.9906	0.9928	0.9952	0.6421	0.7607	0.9954
Email	0.8874	0.9988	0.9988	0.9995	0.8088	0.8981	0.9999
Friendships	0.8859	0.9977	0.9982	0.9964	0.4388	0.4996	0.9991

On the Dolphins dataset, it can be seen from Table 5 that both of the EC and SIWR methods reach the maximum value at the same time. Certainly, the good performance of SIWR is obvious, especially for the KS method, and the bigger difference between them is 0.6210. What is more, the minimum difference between SIWR and the other methods, except EC, is 0.0074. For this, we can make a guess that for big graph data, the SIWR method would show more excellent performance.

On the Polbooks dataset, one can find that the MI, EC and SIWR methods reach the maximum value at the same time, which means that these three methods can completely identify the importance of different nodes and distribute each node to the unique order. What is more, the advantage is also obvious.

Due to the scale of the Adjnoun dataset being similar to that of the Polbooks dataset, most methods obtain similar monotonicity, except the KS method. Obviously, the $M(R)$ value of the KS method on the Adjnoun dataset is significantly larger than that of Polbooks compared to other methods. The main reason is that the maximum degree of Adjnoun is much larger than that of Polbooks. In addition, nodes with larger degree are scattered on the Adjnoun dataset.

On the Football dataset, both DC and KS methods perform poorly, especially the KS method, which obtains the minimum value 0.0003. This shows that the KS method can hardly identify the importance of different nodes on the Football dataset. The reason is related to the topological characteristics of this dataset, as we can find that the minimum degree of this dataset is 7 and the maximum degree is 12, but the average degree is as high as 10.6609. Due to most nodes having the same degree, neither DC nor KS can identify the importance of nodes commendably. In this case, the EC and SIWR methods still reach the maximum value. In terms of another perspective, it confirms the advantage of the SIWR method.

On the Email and Friendship datasets, since the scale of the dataset increases, the EC method that performs well on other datasets does not achieve good results. Obviously, it can be seen from Table 5 that the SIWR method reaches the maximum value on these two datasets.

All in all, the monotonicity values of the SIWR method are vastly superior to most methods. Videlicet, the rank result produced by SIWR method distributes a lower number of nodes to the same order. This is a very nice performance result for the node ranking, especially for the dataset with a certain property, such as the uniform degree distribution of nodes, large number of nodes with high degree, and so on.

5.2. Node Distribution Analysis

Figures 2–4 reflect the curves of CCDF of DC, MI, CC, EC, KS, IKS and SIWR methods on nine datasets. Herein, the vertical axis represents the concrete value of CCDF, and the horizontal axis represents the order number in the rank result.

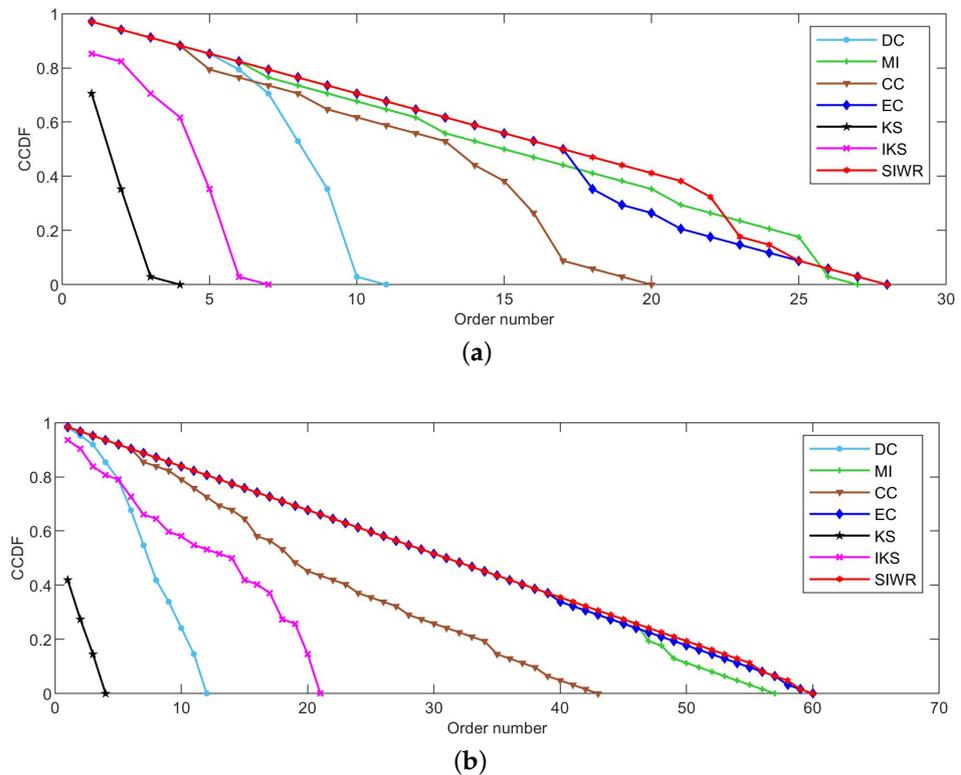


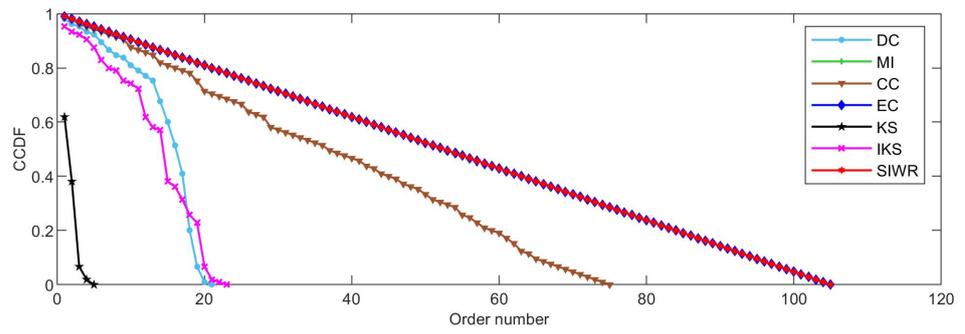
Figure 2. The curves of CCDF on (a) Karate and (b) Dolphins.

Figure 2a is the curve of CCDF on the Karate dataset. Obviously, there are four pentagrams on the curve induced by the KS method. The reason of it is that the index Γ is $\{1, 2, 3, 4\}$ for it. In addition, there are as many as 10 nodes in the first order of the rank result. That is to say, the importance of these 10 nodes are equal. For a dataset with only 34 number of nodes, is it a good rank result? It is not. However, for the SIWR method, it can be easily found that the index Γ is $\{1, 2, \dots, 28\}$. To this, the descending slope of curve of CCDF with respect to the SIWR method is smoother.

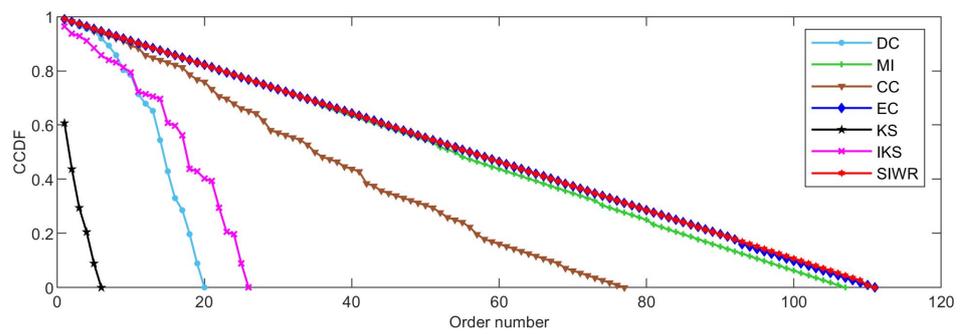
As shown in Figure 2b, the SIWR method shows good ranking performance. There are 62 nodes for the Dolphins dataset, but the order number reaches 60. Frankly speaking, almost every node is located in a unique order, i.e., $n_i = 1$ is true for $i \in \Gamma$ except $n_{56} = 2$ and $n_{59} = 2$.

As can be seen from Figure 3a, the SIWR method can divide the Polbooks dataset into 105 sortable classes. This is a perfect rank result as the node number of this dataset is also equal to 105. The KS method still has the worst ranking ability. The CC method is neither good nor bad. More interestingly, the facts reflected in Figure 3a are consistent with those of Table 5.

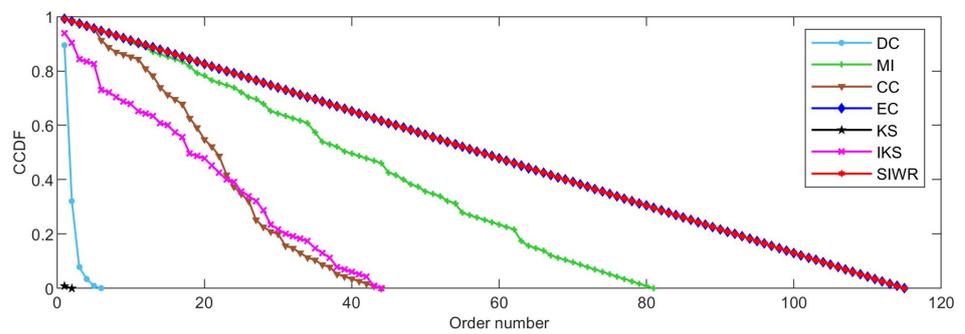
In terms of the distribution of curves from left to right, as well as that of the descending slope, it is not hard to find that the Adjnoun dataset shown in Figure 3b is similar to the Polbooks dataset. However, the IKS method has a smoother descending slope than the DC method on the Adjnoun dataset. Regrettably, the curve of CCDF with respect to the IKS method descends faster at the beginning. Due to the key nodes usually being listed in the front of the rank result, the IKS method cannot better identify the key nodes. In addition, both of EC and SIWR methods obtain the highest order numbers on the Adjnoun dataset. At the same time, the advantage is also obvious.



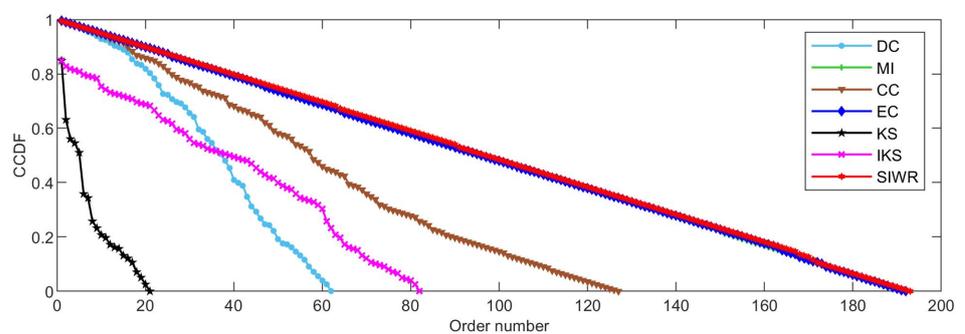
(a)



(b)



(c)



(d)

Figure 3. Cont.

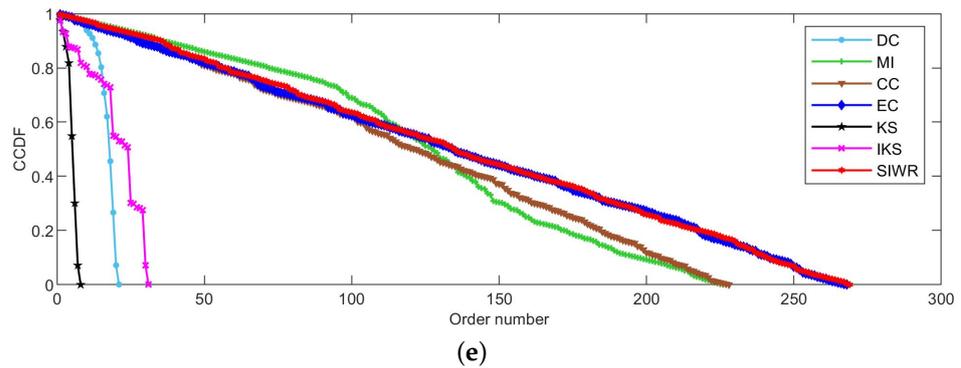


Figure 3. The curves of CCDF on (a) Polbooks, (b) Adjnoun, (c) Football, (d) Jazz and (e) Netscience.

The Football dataset contains 115 nodes, but the KS method simultaneously identifies 114 nodes as the most important nodes. This is a disastrous result. However, for the SIWR method, the order number shown in Figure 3c is 115, which is a perfect rank result. Additionally, its good rank ability is consistent with the monotonicity value of the SIWR method listed in Table 5.

As can be seen from Table 4, the edge number, average degree and maximum of the Jazz dataset are the largest among all the proposed datasets except the Email and Friendships datasets. For such datasets, the ranking methods that can make full use of edge information will have a great advantage. Based on the property of the SIWR method, one can be inferred that the descending slope of the SIWR method should be smoother, and this is verified by Figure 3d.

It can be seen from Figure 3e that the MI method obtains a smoother descending slope at the beginning. However, the slope of decline suddenly increases when the order number is between 100 and 150. The main reason is that the MI method distributes a large number of nodes with the same importance in this interval. In other words, the MI method cannot identify the importance of these nodes. On the whole, the SIWR and EC methods still show great advantages.

Obviously, Figure 4 tells us that the ranking ability of the KS, DC and IKS methods is significantly weaker than that of other methods. For the SIWR method, its overall ranking ability is quite good, as the descending slope of the CCDF curve is smooth. What is more, one can find that the value of CCDF is equal to 0, and the order number of the SIWR method comes up to 1106 in Figure 4a. In addition, the order number of SIWR method comes up to 1487, which is 15 higher than the EC method in Figure 4b. It is worth mentioning that the SIWR method obtains the maximum order number in all methods.

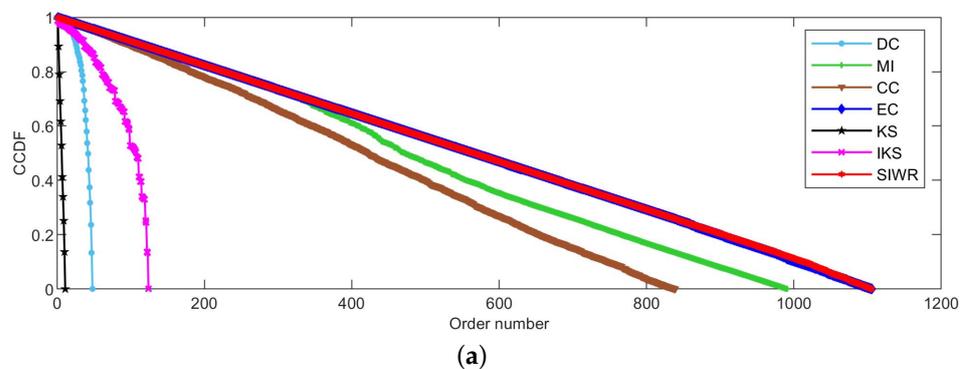


Figure 4. Cont.

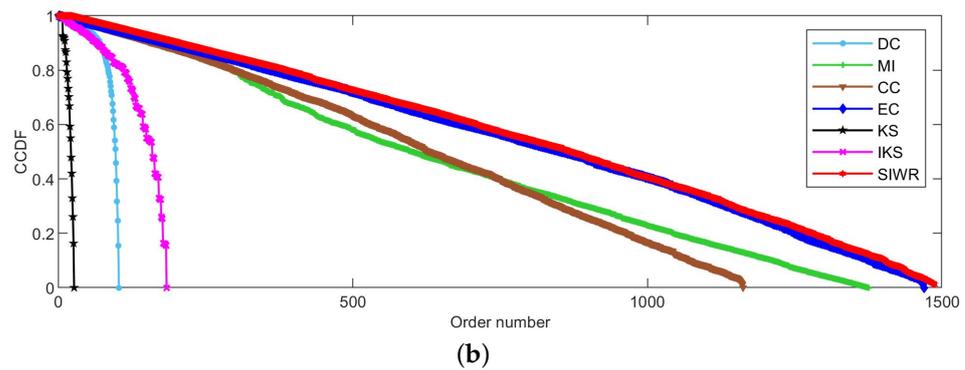


Figure 4. The curves of CCDF on (a) Email and (b) Friendships.

Based on above analysis, one can find that the curves of CCDF with respect to the SIWR method can maintain a smoother descending slope in most datasets. In other words, the SIWR method can lead to a good rank result, in which case little nodes are located at the same order.

5.3. SIR Analysis

In terms of SIR analysis, at first we make a rank for all nodes of each dataset by SIWR, DC, etc. After that, the nodes listed in front of the rank result are selected as seeds, and also are endowed the state of infection. Here we select 2, 4, 6, 8 and 10 nodes as seeds if $n \leq 1000$, and 10, 20, 30, 40 and 50 nodes as seeds once $n > 1000$. What is more, the KS method is excluded from analysis because a large number of nodes have the same order number once the KS method is applied to rank it.

The propagation ability of seeds obtained by DC, MI, CC, EC, IKS and SIWR methods on nine datasets is displayed in Figures 5–7 wherein the horizontal axis of each subfigure represents the number of seeds and the vertical axis of each subfigure represents the propagation ability of seeds.

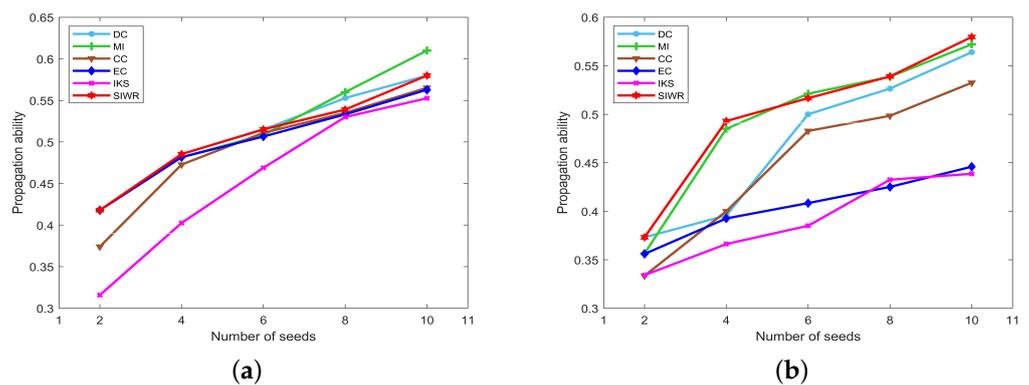


Figure 5. The curves of propagation ability on (a) Karate and (b) Dolphins.

It can be seen from Table 4 that the Karate dataset has only 34 nodes, but the clustering coefficient is large. This indicates that the distribution of these nodes is relatively concentrated, so seeds can obtain a large propagation range on all datasets. Obviously, one can find that the maximum propagation ability is as high as 0.61 from Figure 5a. The IKS method performs the worst, and the SIWR method has the obvious advantages when the number of seeds is equal to 2, 4 and 6.

As shown in Figure 5b, the propagation ability of seeds obtained by the SIWR method is much greater than that of other methods, except for the situation that the number of seeds is equal to 6. The bigger difference between SIWR and others is up to 0.1411. It

is worth mentioning that the key nodes obtained by the EC method have poor accuracy, although it performs well in both of the monotonicity and node distribution experiments.

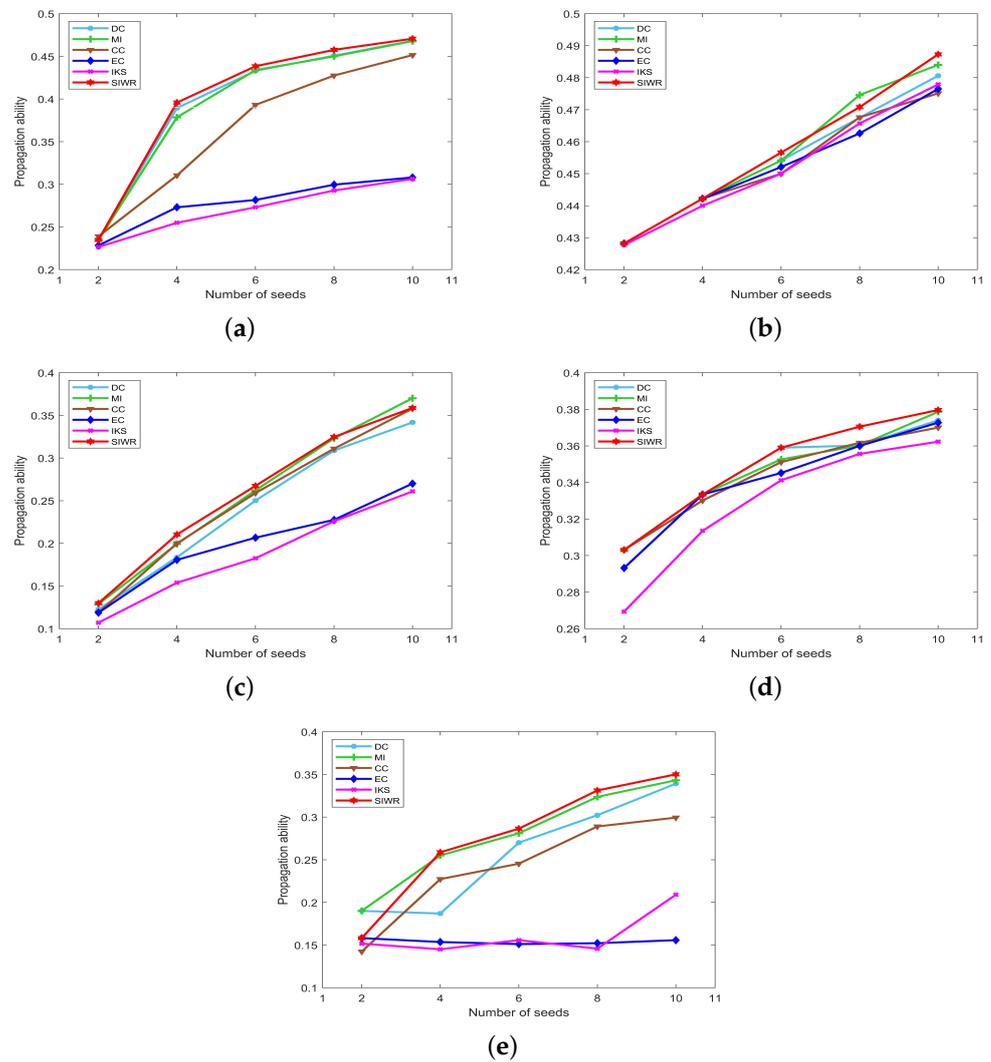


Figure 6. The curves of propagation ability on (a) Polbooks, (b) Adjnoun, (c) Football, (d) Jazz and (e) Netscience.

Similar situations to the Dolphins dataset appear on the Polbooks and Football datasets. The SIWR method achieves the highest propagation ability except at a certain point, while that of EC and IKS methods is much lower compared to others. In addition, from the previous two experiments, one can find that the SIWR method distributes each node to the unique order on these two datasets. At the same time, Figure 6a,c show that the seeds with respect to the SIWR method are more influential. That is, the rank result obtained by the SIWR method not only has higher monotonicity, but also is more accurate.

From Figure 6b,d, one can find that the curves of the propagation ability on the Adjnoun and Jazz datasets are concentrated. In particular, there are multiple methods that obtain the same propagation ability when the number of seeds is equal to 2 and 4, which means that these methods obtain the same key nodes. What is more, the advantage of SIWR method is still obvious. Certainly, the SIWR method exhibits the highest propagation ability for different numbers of seeds on the Jazz dataset.

As can be seen from Figure 6e, with the increasing number of seeds, the curves of the propagation ability corresponding to the EC and IKS methods do not change much, while that of the SIWR method shows an obvious upward trend. The maximum value of the SIWR method is 0.3501, which is 0.1943 higher than the EC method and 0.1411 higher than

the IKS method. This means that the number of nodes infected by the SIWR method is 73 higher than that of the EC method and 53 higher than that of the IKS method. Certainly, the advantage of the SIWR method is obvious.

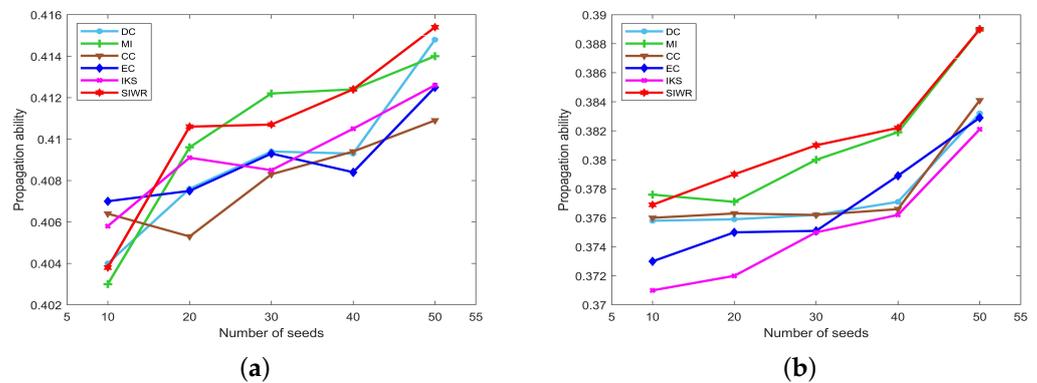


Figure 7. The curves of propagation ability on (a) Email and (b) Friendships.

From Figure 7, the curves of the propagation ability with respect to different methods are constantly fluctuating due to the scale of the dataset increasing. In this case, the SIWR method still maintains the relatively steady upward trend. Especially for the seeds with numbers of 20 or 50 nodes in Figure 7a and the seeds with number of 20 or 30 nodes in Figure 7b, the SIWR method outperforms the other methods obviously. Thus, we could deduce that the key nodes obtained by SIWR method are more accurate for large-scale graph data.

To summarize, the key nodes obtained by SIWR method show better propagation ability especially for large-scale datasets. Therefore, to some extent, the conclusion can be drawn that the SIWR method can obtain more accurate rank results and can be used in large-scale datasets.

5.4. Robustness Analysis

In order to analyze the robustness of the method, we randomly select nodes and remove them from the original datasets. The change rate of rank result is considered after the structure of datasets is changed. First, we randomly select 1% and 5% nodes and delete them from the original datasets. At the same time, the selected nodes are removed from the initial rank results. After that, the remaining nodes are ranked and the new rank result is obtained. Finally, we consider the proportion of nodes whose positions have changed by comparing the two rank results. The experiment will be repeated hundreds of times, and the mean value will be taken as the final change rate. Table 6 shows the rate of change after randomly removing 1% nodes from the original datasets.

Table 6. The rate of change after removing 1% nodes. The best results are highlighted in bold.

Dataset	DC	MI	CC	EC	IKS	SIWR
Karate	0.2767	0.3990	0.5479	0.4898	0.2716	0.2953
Dolphins	0.3678	0.5370	0.5383	0.4934	0.2827	0.4430
Polbooks	0.3960	0.4785	0.6530	0.6810	0.3789	0.4454
Adjnoun	0.3017	0.5280	0.4510	0.3357	0.3904	0.3827
Football	0.8359	0.8536	0.8512	0.8884	0.8439	0.8467
Jazz	0.5123	0.6280	0.5487	0.5392	0.3190	0.5365
Netscience	0.7150	0.6431	0.5652	0.5876	0.6787	0.5324
Email	0.9081	0.8901	0.8998	0.8945	0.8431	0.8421
Friendships	0.9103	0.8600	0.9138	0.8622	0.8953	0.8535

The Karate and Dolphins datasets contain a smaller number of nodes. Only removing one node will not change the rank results dramatically. However, in fact, the MI, CC and

EC methods do not perform well on these two datasets. In particular, the change rate of the CC method is as high as 54.79%, which means that only removing one node will cause more than half of the orders to change in the final rank results. In this case, the DC, IKS and SIWR methods are relatively stable and float between 25% and 45%.

The football datasets are the most special. This dataset has a relatively large average degree and clustering coefficient, which means that the nodes in this dataset are concentrated. The change of the local structure will affect the entire structure to a greater extent. Because of its special topological properties, the change rate of all methods is greater than 80% when only one node is removed. In this case, the SIWR method shows better robustness than the MI, CC and EC methods.

On the Email and Friendships datasets, 11 and 19 nodes were removed, respectively. The change rate of rank results obtained by all method increases significantly compared with other datasets. Obviously, the SIWR method obtains the minimum change rate even better than the DC and IKS methods.

Table 7 shows the rate of change after randomly removing 5% nodes from the original datasets. On the whole, the change rate of all methods increases significantly. The interesting phenomenon is that the advantage of the IKS method disappears and the DC method achieves the minimum rate of change on the Karate and Dolphins datasets. The rate of change obtained from the SIWR method is second only to the DC method on these two datasets.

Table 7. The rate of change after removing 5% nodes. The best results are highlighted in bold.

Dataset	DC	MI	CC	EC	IKS	SIWR
Karate	0.4255	0.5145	0.5776	0.6230	0.4770	0.4566
Dolphins	0.6781	0.7225	0.7135	0.7499	0.7243	0.7042
Polbooks	0.8169	0.7560	0.7862	0.8392	0.7540	0.8170
Adjnoun	0.7678	0.7890	0.7578	0.7117	0.7666	0.7531
Football	0.9170	0.9144	0.8981	0.9270	0.8875	0.9266
Jazz	0.7545	0.8400	0.7840	0.7428	0.6410	0.7537
Netscience	0.9268	0.8688	0.8585	0.8587	0.9131	0.8519
Email	0.9318	0.9249	0.9244	0.9360	0.9274	0.9112
Friendships	0.9327	0.9305	0.9254	0.9259	0.9293	0.9150

The advantage of the SIWR method is reflected in the Netscience, Email and Friendships datasets. The structure of the dataset is changed dramatically after removing 8, 57 and 93 nodes from these three datasets, respectively. In this case, the SIWR method obtains the minimum rate of change, which is consistent with Table 6. Therefore, we can conclude that the SIWR method has strong robustness and can be used in large-scale datasets.

In general, the minimum rate of change is concentrated in the DC, IKS and SIWR methods. However, in the previous experiment, we verified that the DC and IKS methods do not perform well in terms of identifying the importance of nodes. These two methods usually distribute the same score to a large number of nodes. As a result, the importance of different nodes cannot be correctly identified. Although the rank results are not changed significantly after removing a few nodes, these two methods still are unable to accurately identify the importance of different nodes. What is more, the SIWR method showed obvious advantages in previous experiments compared with DC and IKS methods. Table 2 shows that the SIWR method can distribute a lower number of nodes to the same order in all datasets. Therefore, although a few nodes are removed, our method can still obtain the rank result with a small rate of change and high accuracy, especially for the big datasets. To summarize, our method is more robust than the MI, CC and EC methods and has greater advantages in large-scale datasets.

5.5. Running Time Analysis

As we all know, a shorter running time means that the method is faster. Figure 8 shows that SIWR method takes less time than the CC, KS and IKS methods on Karate,

Dolphins, Polbooks and Friendships datasets. What is more, the running time of SIWR is also lower than that of the CC, EC and IKS methods on Adjnoun, Football, Jazz, Netscience and Friendships datasets. Since the SIWR method needs to set weight values for all edges in the graph data, the running time will increase when the number of edges is large. This is the reason that SIWR methods spend significantly more time on the Jazz dataset compared to Adjnoun and Football datasets when the number of nodes is similar.

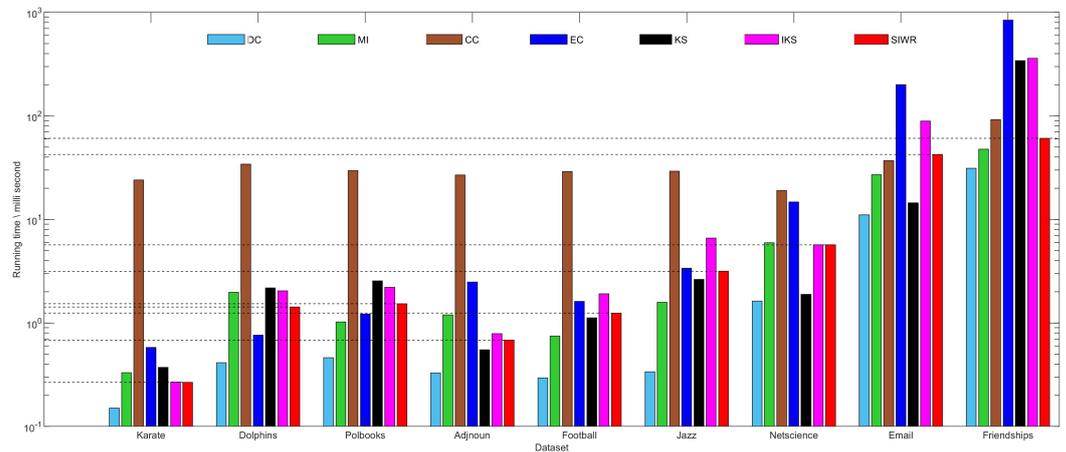


Figure 8. Running time of different methods on nine datasets.

Obviously, DC is the fastest method on all datasets. However, the rank result obtained by the DC method does not achieve better monotonicity and accuracy. The CC method needs to consider the problem of the shortest path in the graph data, so it is the slowest method on most datasets. In general, the running time of the SIWR method is in the middle position among all comparison methods.

6. Conclusions

This paper discussed the node importance ranking method of graph data from the perspective of edges. On one hand, the self-information that takes the nodes degree into account is regarded as the weight of edges, and it turned an unweighted graph datum into a weighted graph datum. On the other hand, we constructed the information entropy of nodes to measure the importance of each node.

A large number of theoretical derivation and experimental analyses demonstrated that the proposed method is more advantageous in aspects of monotonicity, node distribution and accuracy. However, it is not hard to see that this paper only discussed the undirected unweighted graph data. In reality, this is a special case. Therefore, a method that combines the topological properties and the theory of entropy will be considered in our future work. In addition, we will try our best to study graph data with more complicated cases, such as directed graph data, weighted graph data, and so on.

Author Contributions: Writing—original draft, S.L. and H.G.; Writing—review and editing, S.L. and H.G.; Software, S.L.; Formal analysis, H.G. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Natural Science Foundation of China (No. 61966039).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We are hugely grateful to the possible anonymous reviewers for their constructive comments with respect to the original manuscript. At the same time, we acknowledge all the network data used in this paper.

Conflicts of Interest: The author declares that there are no conflict of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

References

1. Liu, J.; Li, X.; Dong, J. A survey on network node ranking algorithms: Representative methods, extensions, and applications. *Sci. China Technol. Sci.* **2021**, *64*, 451–461. [[CrossRef](#)]
2. Borge-Holthoefer, J.; Moreno, Y. Absence of influential spreaders in rumor dynamics. *Phys. Rev. E* **2012**, *85*, 026116. [[CrossRef](#)] [[PubMed](#)]
3. Yao, X.; Gu, Y.; Gu, C.; Huang, H. Fast controlling of rumors with limited cost in social networks. *Comput. Commun.* **2022**, *182*, 41–51. [[CrossRef](#)]
4. Pastor-Satorras, R.; Castellano, C.; Van Mieghem, P.; Vespignani, A. Epidemic processes in complex networks. *Rev. Mod. Phys.* **2015**, *87*, 925. [[CrossRef](#)]
5. Wang, W.; Tang, M.; Stanley, H.E.; Braunstein, L.A. Unification of theoretical approaches for epidemic spreading on complex networks. *Rep. Prog. Phys.* **2017**, *80*, 036603. [[CrossRef](#)]
6. Cui, A.X.; Wang, W.; Tang, M.; Fu, Y.; Liang, X.; Do, Y. Efficient allocation of heterogeneous response times in information spreading process. *Chaos Interdiscip. J. Nonlinear Sci.* **2014**, *24*, 033113. [[CrossRef](#)]
7. Davis, J.T.; Perra, N.; Zhang, Q.; Moreno, Y.; Vespignani, A. Phase transitions in information spreading on structured populations. *Nat. Phys.* **2020**, *16*, 590–596. [[CrossRef](#)]
8. Wen, T.; Deng, Y. Identification of influencers in complex networks by local information dimensionality. *Inf. Sci.* **2020**, *512*, 549–562. [[CrossRef](#)]
9. Zareie, A.; Sheikahmadi, A.; Jalili, M. Influential node ranking in social networks based on neighborhood diversity. *Future Gener. Comput. Syst.* **2019**, *94*, 120–129. [[CrossRef](#)]
10. Yang, Y.Z.; Hu, M.; Huang, T.Y. Influential nodes identification in complex networks based on global and local information. *Chin. Phys. B* **2020**, *29*, 088903. [[CrossRef](#)]
11. Maji, G.; Dutta, A.; Malta, M.C.; Sen, S. Identifying and ranking super spreaders in real world complex networks without influence overlap. *Expert Syst. Appl.* **2021**, *179*, 115061. [[CrossRef](#)]
12. Liu, C.; Cao, T.; Zhou, L. Learning to rank complex network node based on the self-supervised graph convolution model. *Knowl.-Based Syst.* **2022**, *251*, 109220. [[CrossRef](#)]
13. Zareie, A.; Sheikahmadi, A. A hierarchical approach for influential node ranking in complex social networks. *Expert Syst. Appl.* **2018**, *93*, 200–211. [[CrossRef](#)]
14. Lu, P.; Zhang, Z.; Guo, Y.; Chen, Y. A novel centrality measure for identifying influential nodes based on minimum weighted degree decomposition. *Int. J. Mod. Phys. B* **2021**, *35*, 2150251. [[CrossRef](#)]
15. Zhang, J.X.; Chen, D.B.; Dong, Q.; Zhao, Z.D. Identifying a set of influential spreaders in complex networks. *Sci. Rep.* **2016**, *6*, 1–10. [[CrossRef](#)]
16. Sola, L.; Romance, M.; Criado, R.; Flores, J.; Garc?a del Amo, A.; Boccaletti, S. Eigenvector centrality of nodes in multiplex networks. *Chaos Interdiscip. J. Nonlinear Sci.* **2013**, *23*, 033131. [[CrossRef](#)]
17. Fu, Y.H.; Huang, C.Y.; Sun, C.T. Using global diversity and local topology features to identify influential network spreaders. *Phys. Stat. Mech. Appl.* **2015**, *433*, 344–355. [[CrossRef](#)]
18. Goldstein, R.; Vitevitch, M.S. The influence of closeness centrality on lexical processing. *Front. Psychol.* **2017**, *8*, 1683. [[CrossRef](#)]
19. Wen, T.; Pelusi, D.; Deng, Y. Vital spreaders identification in complex networks with multi-local dimension. *Knowl.-Based Syst.* **2020**, *195*, 105717. [[CrossRef](#)]
20. Carmi, S.; Havlin, S.; Kirkpatrick, S.; Shavitt, Y.; Shir, E. A model of Internet topology using k-shell decomposition. *Proc. Natl. Acad. Sci. USA* **2007**, *104*, 11150–11154. [[CrossRef](#)]
21. Zhu, D.; Wang, H.; Wang, R.; Duan, J.; Bai, J. Identification of Key Nodes in a Power Grid Based on Modified PageRank Algorithm. *Energies* **2022**, *15*, 797. [[CrossRef](#)]
22. Wang, T.; Chen, S.S.; Wang, X.X.; Wang, J.F. Label propagation algorithm based on node importance. *Phys. Stat. Mech. Appl.* **2020**, *551*, 124137. [[CrossRef](#)]
23. Yang, P.; Liu, X.; Xu, G. A dynamic weighted TOPSIS method for identifying influential nodes in complex networks. *Mod. Phys. Lett. B* **2018**, *32*, 1850216. [[CrossRef](#)]
24. Bouchon-Meunier, B.; Marsala, C. Entropy and monotonicity in artificial intelligence. *Int. J. Approx. Reason.* **2020**, *124*, 111–122. [[CrossRef](#)]
25. Fan, W.L.; Liu, Z.G.; Hu P. Identifying node importance based on information entropy in complex networks. *Phys. Scr.* **2013**, *88*, 065201.

26. Omar, Y.M.; Plapper, P. A survey of information entropy metrics for complex networks. *Entropy* **2020**, *22*, 1417. [[CrossRef](#)] [[PubMed](#)]
27. Guo, C.; Yang, L.; Chen, X.; Chen, D.; Gao, H.; Ma, J. Influential nodes identification in complex networks via information entropy. *Entropy* **2020**, *22*, 242. [[CrossRef](#)] [[PubMed](#)]
28. Zareie, A.; Sheikahmadi, A.; Fatemi, A. Influential nodes ranking in complex networks: An entropy-based approach. *Chaos Solitons Fractals* **2017**, *104*, 485–494. [[CrossRef](#)]
29. Ai, X. Node importance ranking of complex networks with entropy variation. *Entropy* **2017**, *19*, 303. [[CrossRef](#)]
30. Chen, X.; Zhou, J.; Liao, Z.; Liu, S.; Zhang, Y. A novel method to rank influential nodes in complex networks based on tsallis entropy. *Entropy* **2020**, *22*, 848. [[CrossRef](#)]
31. Fei, L.; Deng, Y. A new method to identify influential nodes based on relative entropy. *Chaos Solitons Fractals* **2017**, *104*, 257–267. [[CrossRef](#)]
32. Wang, Z.; Du, C.; Fan, J.; Xing, Y. Ranking influential nodes in social networks based on node position and neighborhood. *Neurocomputing* **2017**, *260*, 466–477. [[CrossRef](#)]
33. Jiang, Y.; Hu, A.; Huang, J. Importance-based entropy measures of complex networks robustness to attacks. *Clust. Comput.* **2019**, *22*, 3981–3988. [[CrossRef](#)]
34. Chen, B.; Wang, Z.; Luo, C. Integrated evaluation approach for node importance of complex networks based on relative entropy. *J. Syst. Eng. Electron.* **2016**, *27*, 1219–1226. [[CrossRef](#)]
35. Li, J.; Yin, C.; Wang, H.; Wang, J.; Zhao, N. Mining Algorithm of Relatively Important Nodes Based on Edge Importance Greedy Strategy. *Appl. Sci.* **2022**, *12*, 6099. [[CrossRef](#)]
36. Bao, Z.K.; Liu, J.G.; Zhang, H.F. Identifying multiple influential spreaders by a heuristic clustering algorithm. *Phys. Lett. A* **2017**, *381*, 976–983. [[CrossRef](#)]
37. Lu, P.; Zhang, Z. Critical nodes identification in complex networks via similarity coefficient. *Mod. Phys. Lett. B* **2022**, *36*, 2150620. [[CrossRef](#)]
38. Wang, M.; Li, W.; Guo, Y.; Peng, X.; Li, Y. Identifying influential spreaders in complex networks based on improved k-shell method. *Phys. Stat. Mech. Appl.* **2020**, *554*, 124229. [[CrossRef](#)]
39. Liu, Y.; Liu, S.; Yu, F.; Yang, X. Link prediction algorithm based on the initial information contribution of nodes. *Inf. Sci.* **2022**, *608*, 1591–1616. [[CrossRef](#)]
40. Gray, R.M. *Entropy and Information Theory*; Springer Science: Berlin/Heidelberg, Germany, 2011.
41. Langlois, R.N. Modularity in technology and organization. *J. Econ. Behav. Organ.* **2002**, *49*, 19–37. [[CrossRef](#)]
42. Boroujeni, R.J.; Soleimani, S. The role of influential nodes and their influence domain in community detection: An approximate method for maximizing modularity. *Expert Syst. Appl.* **2022**, *202*, 117452. [[CrossRef](#)]
43. Pianosi, F.; Wagener, T. A simple and efficient method for global sensitivity analysis based on cumulative distribution functions. *Environ. Model. Softw.* **2015**, *67*, 1–11. [[CrossRef](#)]
44. Barlow, N.S.; Weinstein, S.J. Accurate closed-form solution of the SIR epidemic model. *Phys. Nonlinear Phenom.* **2020**, *408*, 132540. [[CrossRef](#)] [[PubMed](#)]
45. Kudryashov, N.A.; Chmykhov, M.A.; Vigdorowitsch, M. Analytical features of the SIR model and their applications to COVID-19. *Appl. Math. Model.* **2021**, *90*, 466–473. [[CrossRef](#)]
46. Sheng, J.; Zhu, J.; Wang, Y.; Wang, B.; Hou, Z. Identifying influential nodes of complex networks based on trust-value. *Algorithms* **2020**, *13*, 280. [[CrossRef](#)]