

Article

Identifying Influential Spreaders Using Local Information

Zhe Li ^{1,*}  and Xinyu Huang ^{2,*} ¹ Software College, Shenyang University of Technology of China, Shenyang 110870, China² Software College, Northeastern University of China, Shenyang 110819, China

* Correspondence: lizhe@sut.edu.cn (Z.L.); huangxinyu@mail.neu.edu.cn (X.H.)

Abstract: The heterogeneous nature indicates that different nodes may play different roles in network structure and function. Identifying influential spreaders is crucial for understanding and controlling the spread processes of epidemic, information, innovations, and so on. So how to identify influential spreaders is an urgent and crucial issue of network science. In this paper, we propose a novel local-information-based method, which can obtain the degree information of nodes' higher-order neighbors by only considering the directly connected neighbors. Specifically, only a few iterations are needed to be executed, the degree information of nodes' higher-order neighbors can be obtained. In particular, our method has very low computational complexity, which is very close to the degree centrality, and our method is of great extensibility, with which more factors can be taken into account through proper modification. In comparison with the well-known state-of-the-art methods, experimental analyses of the Susceptible-Infected-Recovered (SIR) propagation dynamics on ten real-world networks evidence that our method generally performs very competitively.

Keywords: influential spreaders; local information; complex network; network information mining

MSC: 05C82



Citation: Li, Z.; Huang, X. Identifying Influential Spreaders Using Local Information. *Mathematics* **2023**, *11*, 1302. <https://doi.org/10.3390/math11061302>

Academic Editor: Andrea Scozzari

Received: 16 February 2023

Revised: 5 March 2023

Accepted: 7 March 2023

Published: 8 March 2023



Copyright: © 2023 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

The development of science and technology has greatly enriched human life, making people more engaged in exploring the nature of the world. Network science has provided powerful tools to abstract and understand complex mechanisms in many real-world scenarios [1,2]. Recently, the focus of network science has shifted from the discovery of macroscopic statistical regularities to the unfolding of mesoscopic structures organization, and further to the revelation of the significant role played by microscopic elements (i.e., nodes and links) [3]. The scale-free property [4] indicates that different nodes may play different roles in network function and structure. Vital nodes are some special ones that can affect the structure and function of the network to a greater extent. Although the number of vital nodes is generally very small, the influence is very large [5]. For example, an influential user on Twitter may significantly speed up the spread of information, a super spreader of epidemic may greatly increase the scale of epidemic, a deliberate attack on a few vital servers may bring down the entire network, and the failure of a few critical power grids may lead to catastrophic blackouts. Therefore, it is very important to identify vital nodes associated with some certain functional or structural objectives, which allows us to better control the outbreak of an epidemic [6–8], control rumor propagation [9,10], analyze drugs and proteins [11,12], analyze financial risks [13,14], analyze online social media systems [15], predict outstanding scientists or journals [16,17], prevent catastrophic disruptions of Internet or power grids [18,19], discover important species [20,21], affect political elections [22], and so on.

Most known methods make use of structural information [3]. Typical local-information-based representatives are degree centrality (DC) and H-index [23]. DC is the simplest local centrality only considering the number of directly connected neighbors of nodes, the larger

the DC is, the more directly connected neighbors the node has. H-index is a local centrality in which each node only needs the information about the degrees of its neighbors, the larger the H-index is, the more large-degree neighbors the node has. Typical global-information-based representatives are k -shell decomposition method [24] (KS), eigenvector centrality [25] (EC), betweenness centrality [26] (BC) and closeness centrality [27] (CC). KS is a global centrality describing the location of each node, the larger the KS is, the closer the node to the core of the network. EC is a global centrality which indicates that the influence of a node depends not only on the number of its neighbors, but also on the influence of each neighbor. BC is a global centrality describing the controllability of information flow of each node, the larger the BC is, the more shortest paths pass through the node. CC is a global centrality describing the distance from one node to all the other nodes in the network, the larger the CC is, the smaller the distance from the node to all the other nodes.

Although the global centralities usually perform better than the local centralities, they need the global topology information of the network, which makes it difficult to apply the global centralities in very large-scale dynamic networks [23]. Besides, the path-based global centralities (such as BC, CC) are very time-consuming. Fortunately, by fully considering the information contained in the fourth-order neighbors of each node, Chen et al. [28] proposed an effective local algorithm named LocalRank (LR). LR is not inferior in the comparison of the above global methods, which suggests that the well-designed local centralities have very promising potential. Recently, a large number of local-information-based algorithms have been proposed, and their performance is generally better than the classic centralities. By employing the probability model, Chen et al. [29] proposed a novel method named DynamicRank (DR). Inspired by gravity law, Li et al. [30] proposed a local gravity model (LGM), which considers both neighborhood and path information. From the perspective of "graph energy", Ma et al. [31] proposed a novel method named Quasi-Laplacian centrality (QC). Based on QC, Zhao et al. [32] proposed a variant algorithm named the third Laplacian energy centrality (LC). By combining LGM and QC, Zhang et al. [33] proposed an effective method named Laplacian gravity centrality (LGC). Inspired by the local tree-like structure raised from the Push-Republish model [34], Hao et al. [35] proposed a novel method named Local-Forest (LF).

Although the above methods have performed very well, it is necessary to consider the higher-order neighborhood of nodes for these methods (such as DR, LGM, LC, LGC, LF), which limits their application and is relatively time-consuming. In view of this, we propose an iterative refinement centrality that works by iterating the following process: all nodes collect the information of their directly connected neighbors and update the information of themselves. Therefore, we can only consider the directly connected neighbors without considering the higher-order neighborhood of nodes, and we can still obtain the information of higher-order neighbors by multiple iterations. The experimental results show that our method performs very competitively in comparison with the above methods.

The outline of the paper is as follows. Preliminaries are introduced in Section 2. Our method is proposed in Section 3. Experiments are shown in Section 4. Finally, discussion and conclusion are presented in Section 5 and Section 6, respectively.

2. Preliminaries

Given a simple undirected and unweighted network $G = \langle V, E \rangle$, where V is the set of nodes in G , and E represents the set of links connecting pairs of nodes. Denote $|V| = N$ and $|E| = M$, then the network includes N nodes and M links. The adjacent matrix of network G is denoted by $A = (a_{ij})_{N \times N}$, if there exists a link from node i to node j , $a_{ij} = 1$, otherwise, $a_{ij} = 0$.

2.1. Well-Known State-of-the-Art Methods

The degree centrality (DC) of node i is defined as the following Equation (1):

$$DC(i) = k(i), \quad (1)$$

where $k(i) = \sum_{j=1}^N a_{ij}$.

The H-index [23] of node i , denoted by $H(i)$, is defined as the maximal integer satisfying that there are at least $H(i)$ neighbors of node i whose degrees are all no less than $H(i)$.

The k -shell decomposition method [24] (KS) works by decomposing the network into different shells iteratively. The first step of KS is to remove the nodes with degree $k = 1$ from the network, and then remove all the nodes whose residual degree $k \leq 1$ until all the remaining nodes' residual degree $k > 1$. All of the nodes removed in the first step form 1-shell with k -shell values equal to 1. Repeat the process to get 2-shell, 3-shell, ... The decomposition process continues until there are no more nodes in the network.

The eigenvector centrality [25] (EC) of node i is defined as the following Equation (2):

$$EC(i) = c \sum_{j=1}^N a_{ij} EC(j), \tag{2}$$

where c is generally set to be the reciprocal of the largest eigenvalue of A .

The betweenness centrality [26] (BC) of node i is defined as the following Equation (3):

$$BC(i) = \sum_{s \neq i, s \neq t, i \neq t} \frac{g_{st}(i)}{g_{st}}, \tag{3}$$

where g_{st} is the number of shortest paths between node s and node t , and $g_{st}(i)$ is the number of shortest paths through node i between node s and node t .

The closeness centrality [27] (CC) of node i is defined as the following Equation (4):

$$CC(i) = \frac{N - 1}{\sum_{j \neq i} d(i, j)}, \tag{4}$$

where $d(i, j)$ is the shortest distance between node i and node j .

The LocalRank [28] (LR) of node i is defined as the following Equation (5):

$$LR(i) = \sum_{j \in \Lambda_i} \sum_{w \in \Lambda_j} R(w), \tag{5}$$

where Λ_i is the set of the nearest neighbors of node i , and $R(w)$ is the number of the nearest and the next nearest neighbors of node w .

The DynamicRank [29] (DR) of node i is defined as the following Equation (6):

$$DR(i) = \sum_{t=1}^3 \sum_{j \in \Gamma_t(i)} score(j, t), \tag{6}$$

where $\Gamma_t(i)$ is the set of t -th order neighbours of node i , and $score(j, t)$ can be obtained by the following Equation (7):

$$score(j, t) = 1 - \prod_{w \in \Gamma_{t-1}(i)} (1 - score(w, t - 1) * \beta), \tag{7}$$

where β the infected probability, and there exists a link between node j and node w .

The local gravity model [30] (LGM) of node i is defined as the following Equation (8):

$$LGM(i) = \sum_{d(i,j) \leq R, j \neq i} \frac{k(i)k(j)}{d^2(i, j)}, \tag{8}$$

where R is the truncation radius, and the optimal truncation radius R^* can be estimated by the following Equation (9):

$$R^* \approx \frac{1}{2} \langle d \rangle, \tag{9}$$

where $\langle d \rangle$ is the average distance of the network.

The Quasi-Laplacian centrality [31] (QC) of node i is defined as the following Equation (10):

$$QC(i) = k(i)^2 + k(i) + 2 \sum_{j \in \Lambda_i} k(j). \tag{10}$$

The Laplacian centrality [32] (LC) of node i is defined as the following Equation (11):

$$LC(i) = k(i)^3 + 3k(i)^2 - 2k(i) + \sum_{j \in \Lambda_i} (3k(j)^2 + 3k(j)) - 6\Delta_i, \tag{11}$$

where Δ_i is the number of triangles containing node i .

The Laplacian gravity centrality [33] (LGC) of node i is defined as the following Equation (12):

$$LGC(i) = \sum_{d(i,j) \leq \langle d \rangle / 2, j \neq i} \frac{QC(i)QC(j)}{d^2(i,j)}. \tag{12}$$

The Local-Forest [35] (LF) of node i is defined as the following Equation (13):

$$LF(i) = \sum_{j \in \Lambda_i} \sum_{w \in \Lambda_j} (k(w) - 2m(j)), \tag{13}$$

where $m(j)$ is the number of links between neighbors of node j .

2.2. SIR Model

In the SIR model [36], initially, one node selected as the seed is in the infected state (I), and the others are in the susceptible state (S). In each step, every infected node will infect its susceptible neighbors with probability β , and then each infected node changes to be recovered state (R) with probability λ . The influence of node i can be calculated by the following Equation (14):

$$F(i) = N_r / N, \tag{14}$$

where N_r is the number of recovered nodes when the propagation process is finished. For the sake of simplicity, λ is set to be 1, and then the corresponding epidemic threshold [37] can be calculated by the following Equation (15):

$$\beta_c \approx \frac{\langle k \rangle}{\langle k^2 \rangle - \langle k \rangle}, \tag{15}$$

where $\langle k \rangle$ and $\langle k^2 \rangle$ are the average degree and the second-order moment of the degree distribution.

2.3. The Kendall's Tau

The Kendall's Tau [38] is used to measure the strength of correlation between two sequences. $C = (c_1, c_2, \dots, c_N)$ and $D = (d_1, d_2, \dots, d_N)$ are two sequences with N elements. For any pair of two-tuples (c_i, d_i) and (c_j, d_j) ($i \neq j$), if both $c_i > c_j$ and $d_i > d_j$ or both $c_i < c_j$ and $d_i < d_j$, the pair of two-tuples is concordant. If both $c_i > c_j$ and $d_i < d_j$ or both $c_i < c_j$ and $d_i > d_j$, the pair of two-tuples is discordant. If $c_i = c_j$ or $d_i = d_j$, the pair of two-tuples is neither concordant nor discordant. The Kendall's Tau of two sequences C and D is defined as the following Equation (16):

$$\tau = \frac{2(n_+ - n_-)}{N(N - 1)}, \tag{16}$$

where n_+ and n_- are the number of concordant pairs and discordant pairs.

2.4. The Monotonicity

The monotonicity [39] of ranking list is used to measure the resolution of different ranking lists, and it is defined as the following Equation (17):

$$M(L) = \left[1 - \frac{\sum_{r \in L} U_r(U_r - 1)}{U(U - 1)}\right]^2, \tag{17}$$

where U is the size of list L , and U_r is the number of ties with the same rank r .

3. Methods

Most local-information-based algorithms with good performance usually take into account the third-order neighborhood or even higher-order neighborhood of nodes, which limits their application and relatively time-consuming. In view of this, we propose an iterative refinement centrality named degree information propagation (DP). It works by iterating the following process: all nodes collect the information of their directly connected neighbors and update the information of themselves. Initially, each node takes its degree as its information, so the zero-order information of node i , denoted by $I_0(i)$, can be defined as the following Equation (18):

$$I_0(i) = k(i). \tag{18}$$

In iteration 1, each node collects the zero-order information of its directly connected neighbors and update the information of itself, so the first-order information of node i , denoted by $I_1(i)$, can be defined as the following Equation (19):

$$I_1(i) = \sum_{j \in \Lambda_i} I_0(j). \tag{19}$$

In iteration 2, each node collects the first-order information of its directly connected neighbors and update the information of itself, so the second-order information of node i , denoted by $I_2(i)$, can be defined as the following Equation (20):

$$I_2(i) = \sum_{j \in \Lambda_i} I_1(j). \tag{20}$$

By analogy, in iteration l , each node collects the $l - 1$ th-order information of its directly connected neighbors and update the information of itself, so the l th-order information of node i , denoted by $I_l(i)$, can be defined as the following Equation (21):

$$I_l(i) = \sum_{j \in \Lambda_i} I_{l-1}(j). \tag{21}$$

Finally, for DP, the influence of node i can be calculated by the following Equation (22):

$$DP(i) = \sum_{l=1}^T I_l(i), \tag{22}$$

where T is the number of iterations.

An example network (shown in Figure 1a) is used to illustrate our method. As shown in Figure 1d,f,h, the more iterations, the richer information collected by node 2. In iteration 1, node 2 collects the degree information of node 1, node 3 and node 4. In iteration 2, node 2 collects the degree information of node 3, node 4, node 5, node 6 and itself. In iteration 3, node 2 collects the degree information of all nodes. Therefore, in our method, we can only consider the directly connected neighbors without considering the higher-order neighborhood of nodes, and can still obtain the degree information of higher-order neighbors by multiple iterations.

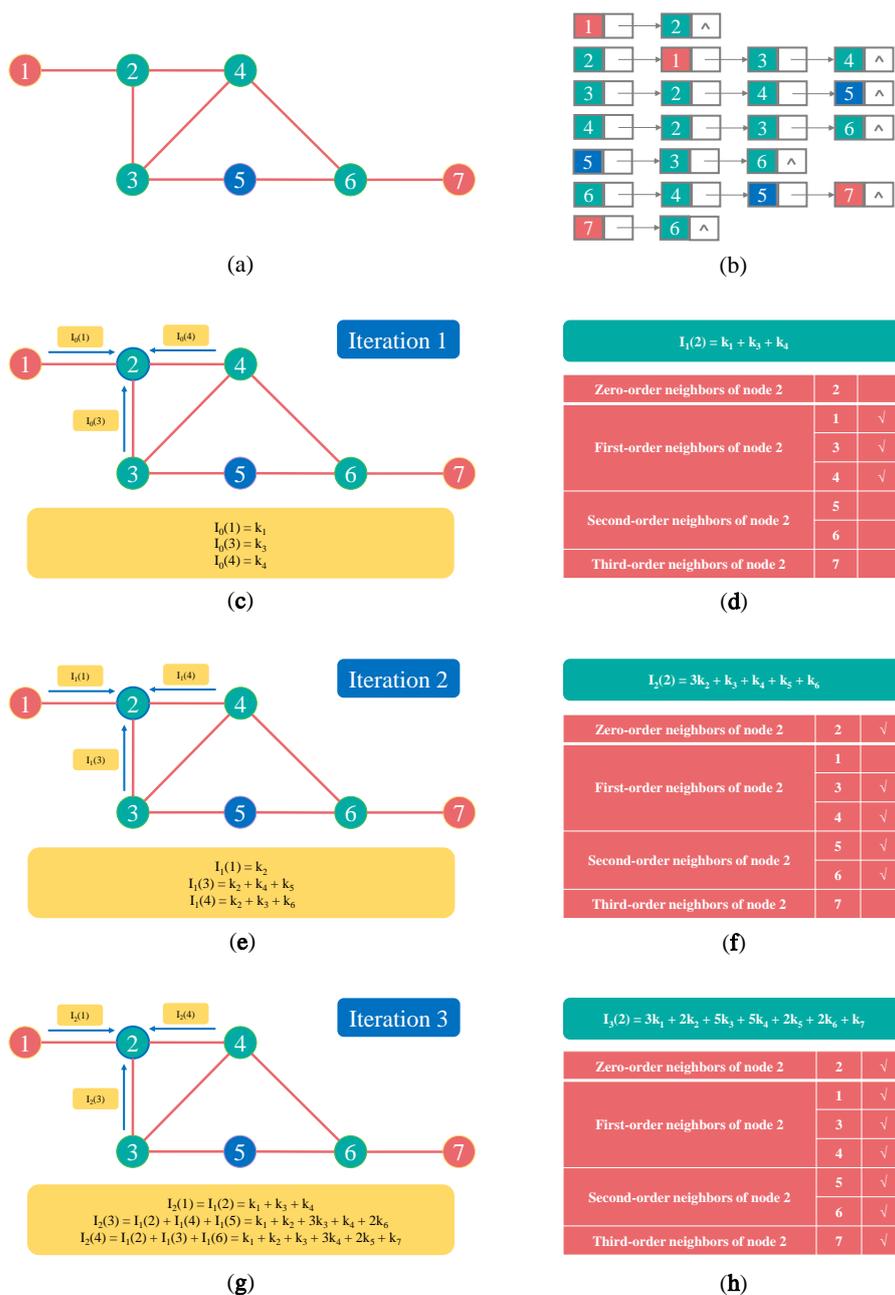


Figure 1. (a) An example network. (b) The adjacency list of the example network. (c,e,g) The process of each iteration of node 2. (d,f,h) The degree information of each order neighborhood collected by node 2 after each iteration.

However, unrestricted iteration will eventually reduce the role of the information collected in the previous iterations, mathematically, $I_l \geq I_{l-1}$.

Theorem 1. Let G be a simple undirected and unweighted connected network, for any node i in G , $I_l(i) \geq I_{l-1}(i)$.

Proof of Theorem 1. By mathematical induction, when $l = 1$, $I_1(i) = \sum_{j \in \Lambda_i} I_0(j) = \sum_{j \in \Lambda_i} k(j)$. As $k(j) \geq 1$, so $I_1(i) = \sum_{j \in \Lambda_i} k(j) \geq \sum_{j \in \Lambda_i} 1 = k(i) = I_0(i)$. When $l = p$, provided $I_p(i) \geq I_{p-1}(i)$. Then when $l = p + 1$, $I_{p+1}(i) = \sum_{j \in \Lambda_i} I_p(j) \geq \sum_{j \in \Lambda_i} I_{p-1}(j) = I_p(i)$. Therefore, $I_l(i) \geq I_{l-1}(i)$. □

In order to solve this problem, we propose a restricted version of DP named restricted degree information propagation (RDP). When $l \geq 1$, $I_l(i)$ can be redefined as the following Equation (23):

$$I_l(i) = \frac{1}{l^2} \sum_{j \in \Lambda_i} I_{l-1}(j). \tag{23}$$

Finally, for RDP, the influence of node i can be calculated by the following Equation (24):

$$RDP(i) = \sum_{l=1}^T I_l(i). \tag{24}$$

The results of DP ($T = 3$) and RDP ($T = 3$) of the example network are shown in Tables 1 and 2, respectively.

Table 1. The results of DP ($T = 3$) of the example network.

| DP | Node 1 | Node 2 | Node 3 | Node 4 | Node 5 | Node 6 | Node 7 |
|---------|--------|--------|--------|--------|--------|--------|--------|
| I_0 | 1 | 3 | 3 | 3 | 2 | 3 | 1 |
| I_1 | 3 | 7 | 8 | 9 | 6 | 6 | 3 |
| I_2 | 7 | 20 | 22 | 21 | 14 | 18 | 6 |
| I_3 | 20 | 50 | 55 | 60 | 40 | 41 | 18 |
| Results | 30 | 77 | 85 | 90 | 60 | 65 | 27 |

Table 2. The results of RDP ($T = 3$) of the example network.

| RDP | Node 1 | Node 2 | Node 3 | Node 4 | Node 5 | Node 6 | Node 7 |
|---------|--------|---------|---------|---------|---------|---------|--------|
| I_0 | 1.0000 | 3.0000 | 3.0000 | 3.0000 | 2.0000 | 3.0000 | 1.0000 |
| I_1 | 3.0000 | 7.0000 | 8.0000 | 9.0000 | 6.0000 | 6.0000 | 3.0000 |
| I_2 | 1.7500 | 5.0000 | 5.5000 | 5.2500 | 3.5000 | 4.5000 | 1.5000 |
| I_3 | 0.5556 | 1.3889 | 1.5278 | 1.6667 | 1.1111 | 1.1389 | 0.5000 |
| Results | 5.3056 | 13.3889 | 15.0278 | 15.9167 | 10.6111 | 11.6389 | 5.0000 |

The computational complexity of the algorithms used in this paper is shown in Table 3. For each iteration, the computational complexity for all nodes is $O(N\langle k \rangle)$. Fortunately, DP generally gets gratifying results after two iterations, and RDP generally gets gratifying results after three iterations (see more details in Section 4). Therefore, the computational complexity of DP and RDP is just $O(N\langle k \rangle)$.

Table 3. The computational complexity of different algorithms.

| Algorithms | Topology | Complexity |
|------------|----------|---------------------------|
| DC | Local | $O(N)$ |
| H-index | Local | $O(N + M)$ |
| DP/RDP | Local | $O(N\langle k \rangle)$ |
| QC | Local | $O(N\langle k \rangle)$ |
| LR | Local | $O(N\langle k \rangle^2)$ |
| LF | Local | $O(N\langle k \rangle^2)$ |
| LC | Local | $O(N\langle k \rangle^2)$ |
| DR | Local | $O(N\langle k \rangle^3)$ |
| LGM | Local | $O(N\langle k \rangle^R)$ |
| LGC | Local | $O(N\langle k \rangle^R)$ |
| KS | Global | $O(M)$ |
| EC | Global | $O(N + M)$ |
| BC | Global | $O(NM + N^2 \log N)$ |
| CC | Global | $O(NM + N^2 \log N)$ |

4. Experiments

4.1. Data Description

In this paper, we employ ten real-world networks from different fields to verify the performance of DP and RDP, listed as follows

- Technological network: Router [40];
- Infrastructure network: Power [41];
- Communication network: Email [42];
- Transportation network: USAir [43];
- Collaboration networks: NS [44] and Jazz [45];
- Social networks: Sex [46], WV [47], Facebook [48] and PB [49].

Table 4 shows the ten real-world networks' topological features, including the number of nodes and links, the average degree, the average distance and the epidemic threshold [37] of the SIR model [36].

Table 4. The topological features of ten real-world networks.

| Network | N | M | $\langle k \rangle$ | $\langle d \rangle$ | β_c |
|----------|--------|---------|---------------------|---------------------|-----------|
| Router | 5022 | 6258 | 2.4922 | 6.4488 | 0.0786 |
| Power | 4941 | 6594 | 2.6691 | 18.9892 | 0.3483 |
| Email | 1133 | 5451 | 9.6222 | 3.6060 | 0.0565 |
| USAir | 332 | 2126 | 12.8072 | 2.7381 | 0.0231 |
| NS | 379 | 914 | 4.8232 | 6.0419 | 0.1424 |
| Jazz | 198 | 2742 | 27.6970 | 2.2350 | 0.0266 |
| Sex | 15,810 | 38,540 | 4.8754 | 5.7846 | 0.0365 |
| WV | 7066 | 100,736 | 28.5129 | 3.2475 | 0.0069 |
| Facebook | 4039 | 88,234 | 43.6910 | 3.6925 | 0.0095 |
| PB | 1222 | 16,714 | 27.3552 | 2.7375 | 0.0125 |

4.2. Experimental Results

We use the simulation results produced by SIR model [36] as the standard rankings of the influence of nodes. Given the network and the spreading probability β , in order to obtain the stable simulation results, 1000 independent implementations need to be executed and averaged, and in each implementation, each node is selected as the seed once. Then, we use the Kendall's Tau (τ) between the standard rankings produced by SIR model and the rankings produced by the algorithm to measure the accuracy of the algorithm. The larger the value of τ , the stronger the correlation between the two sequences, and thus the better the performance of the algorithm.

Firstly, Table 5 shows the accuracies of DP and RDP with different iterations, where DP1 and RDP1 represent one iteration of DP and RDP, respectively. For each network, the best performed version of DP is emphasized by italic, the best performed version of RDP is emphasized by bold, and the best performed version of DP and RDP is emphasized by underline.

As shown in Table 5, DP generally gets gratifying results after two iterations, and RDP generally gets gratifying results after three iterations (the improvement of four iterations is not obvious, only four of the ten networks are slightly better than three iterations). In addition, except for Power, RDP performs better than DP, which suggests the effectiveness of our restriction strategy. Considering the universality and simplicity of our algorithms, the number of iterations of DP and RDP is set to 3.

Furthermore, 13 benchmark algorithms are selected in our experiments, respectively are: degree centrality (DC), H-index [23], k -shell decomposition method [24] (KS), eigenvector centrality [25] (EC), betweenness centrality [26] (BC), closeness centrality [27] (CC), LocalRank (LR) [28], DynamicRank (DR) [29], local gravity model (LGM) [30], Quasi-Laplacian centrality (QC) [31], the third Laplacian energy centrality (LC) [32], Laplacian gravity centrality (LGC) [33] and Local-Forest (LF) [35]. Figure 2 shows the accuracies of

DP/RDP and 13 benchmark algorithms for $\beta = \beta_c$. The T of DP and RDP is set to 3. The R of LGM and LGC is set to $\langle d \rangle / 2$. The β of DR is set to $2.5\beta_c$. Further, Figure 3 shows the accuracies of different β values which are not too far from β_c .

Table 5. The accuracies of DP and RDP with different iterations.

| Network | DP = 1 | DP = 2 | DP = 3 | DP = 4 | DP = 5 | RDP = 1 | RDP = 2 | RDP = 3 | RDP = 4 | RDP = 5 |
|----------|--------|--------|--------|--------|--------|---------|---------------|---------------|---------------|---------------|
| Router | 0.6792 | 0.8063 | 0.7991 | 0.7654 | 0.7563 | 0.6792 | 0.8071 | 0.8159 | 0.8164 | 0.8127 |
| Power | 0.6772 | 0.7688 | 0.8138 | 0.8534 | 0.8572 | 0.6772 | 0.7416 | 0.7625 | 0.7702 | 0.7719 |
| Email | 0.8932 | 0.9230 | 0.9214 | 0.9143 | 0.9073 | 0.8932 | 0.9212 | 0.9241 | 0.9217 | 0.9190 |
| USAir | 0.8984 | 0.9044 | 0.8956 | 0.8957 | 0.8948 | 0.8984 | 0.9046 | 0.8973 | 0.8966 | 0.8955 |
| NS | 0.8550 | 0.8991 | 0.8801 | 0.8509 | 0.8174 | 0.8550 | 0.8928 | 0.8995 | 0.8985 | 0.8974 |
| Jazz | 0.9043 | 0.9339 | 0.9340 | 0.9219 | 0.9102 | 0.9043 | 0.9324 | 0.9351 | 0.9277 | 0.9201 |
| Sex | 0.7520 | 0.7989 | 0.7964 | 0.7933 | 0.7621 | 0.7520 | 0.8029 | 0.8151 | 0.8184 | 0.8066 |
| WV | 0.8312 | 0.8379 | 0.8360 | 0.8348 | 0.8341 | 0.8312 | 0.8379 | 0.8361 | 0.8350 | 0.8343 |
| Facebook | 0.8003 | 0.8483 | 0.8599 | 0.8641 | 0.8382 | 0.8003 | 0.8474 | 0.8609 | 0.8670 | 0.8515 |
| PB | 0.8964 | 0.9213 | 0.9178 | 0.9150 | 0.9095 | 0.8964 | 0.9214 | 0.9185 | 0.9162 | 0.9118 |

As shown in Figure 2, the effective local-information-based algorithms (LR, DR, QC, LC, LF, LGM, LGC, DP and RDP) perform better than the classic algorithms (DC, H-index, KS, EC, BC and CC). Although EC performs well on some networks (especially in USAir and WV), it performs extremely poorly in these two networks (Power and NS). Furthermore, for the local-information-based algorithms, RDP generally performs better than the other algorithms. In ten real-world networks, RDP gets six best results (Email, NS, Jazz, WV, Facebook and PB), DR gets two best results (Router, USAir), and LGC gets two best results (Power and Sex). Considering that the complexity of these two algorithms is relatively high in local-information-based algorithms, while our method has the lowest complexity, which suggests that our method has both accuracy and efficiency. Furthermore, as shown in Figure 3, DP and RDP still perform very competitively compared with 13 benchmark algorithms for different β values, which suggests the robustness of our findings.

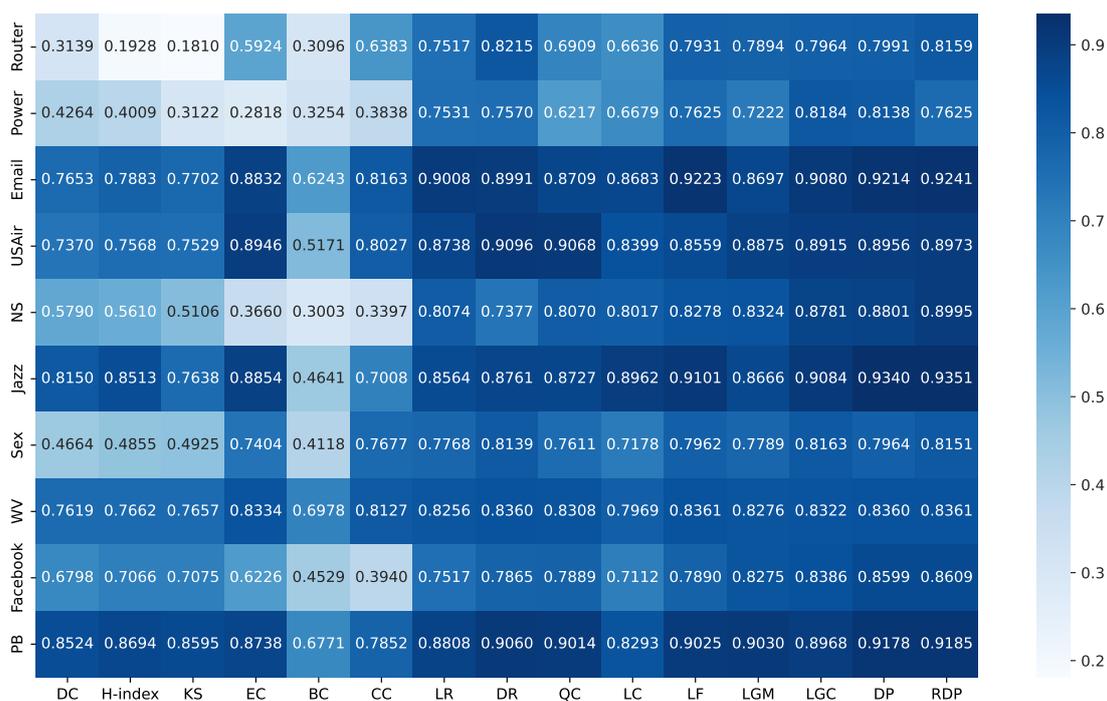


Figure 2. The accuracies of DP/RDP and 13 benchmark algorithms for $\beta = \beta_c$.

Figure 4 shows the runtime and accuracy of different algorithms in the ten real-world networks. The closer the point corresponding to the algorithm is to the upper left corner, the higher the accuracy and the shorter the runtime of the algorithm. It is obvious that DP and RDP perform very well.

Finally, we use the monotonicity [39] to measure the resolution of different algorithms. As shown in Figure 5, only six algorithms (i.e., EC, DR, LGM, LGC, DP and RDP) have monotonicity above 0.99 for all networks, which demonstrates that DP and RDP are remarkably high-resolution algorithms. Generally speaking, the higher the complexity of the algorithm, the higher the resolution. DR and RDP can still achieve gratifying results even with extremely low complexity, which shows the superiority of our method.

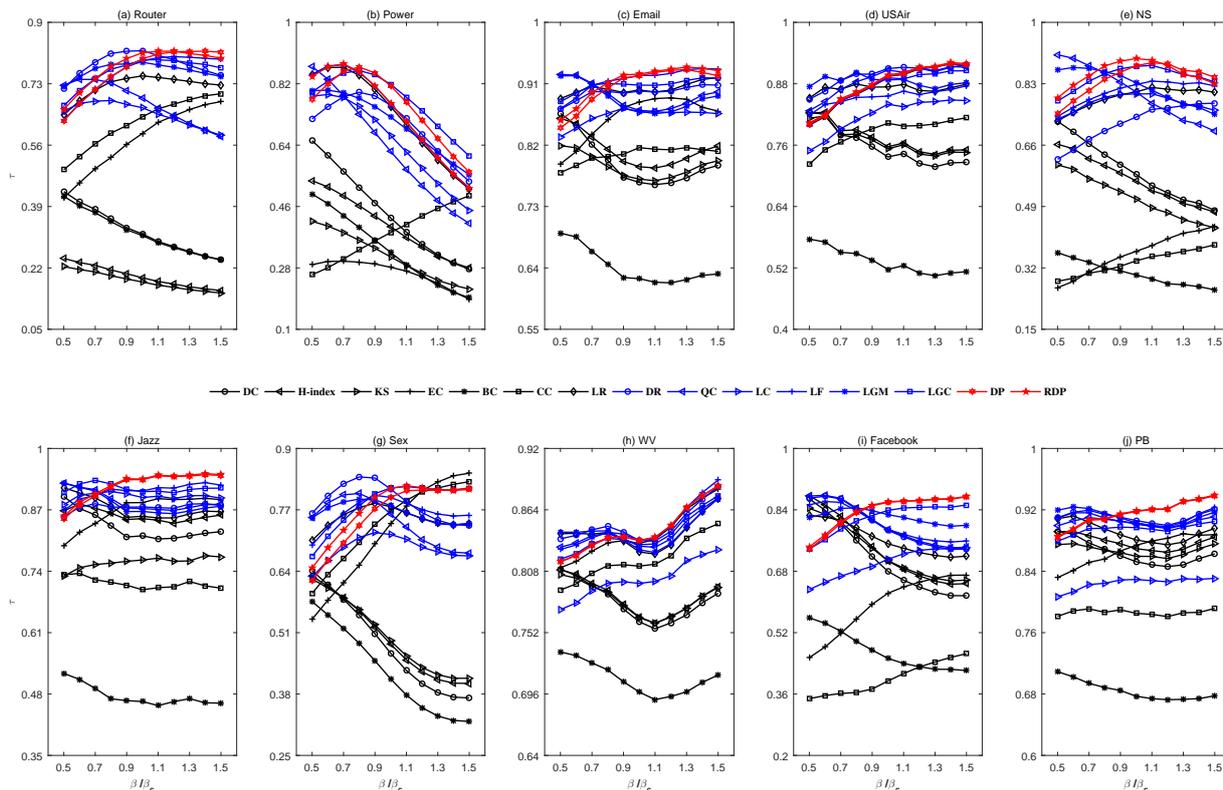


Figure 3. The algorithms’ accuracies measured by Kendall’s Tau for different β .

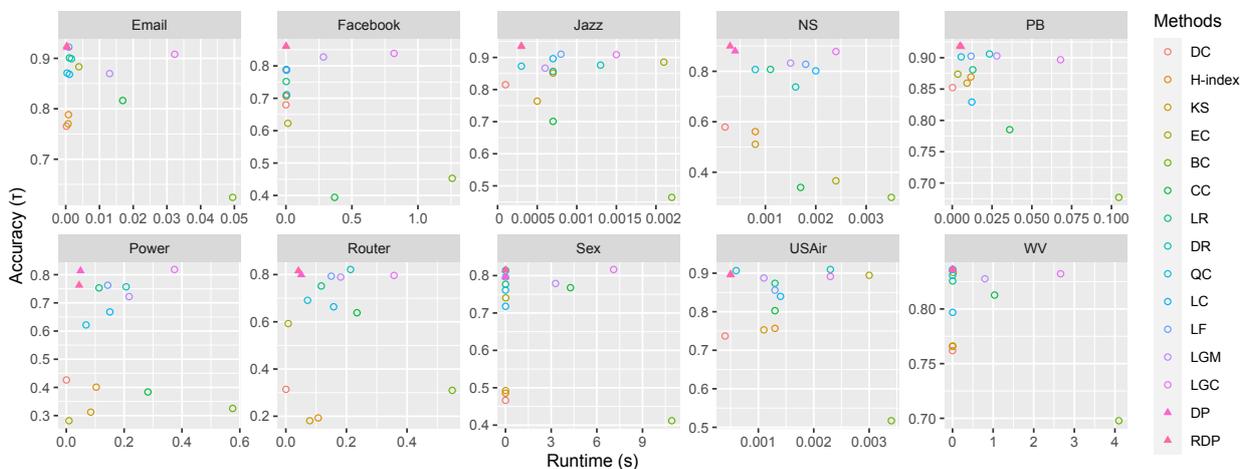


Figure 4. Comparison of runtime and accuracy on ten networks.

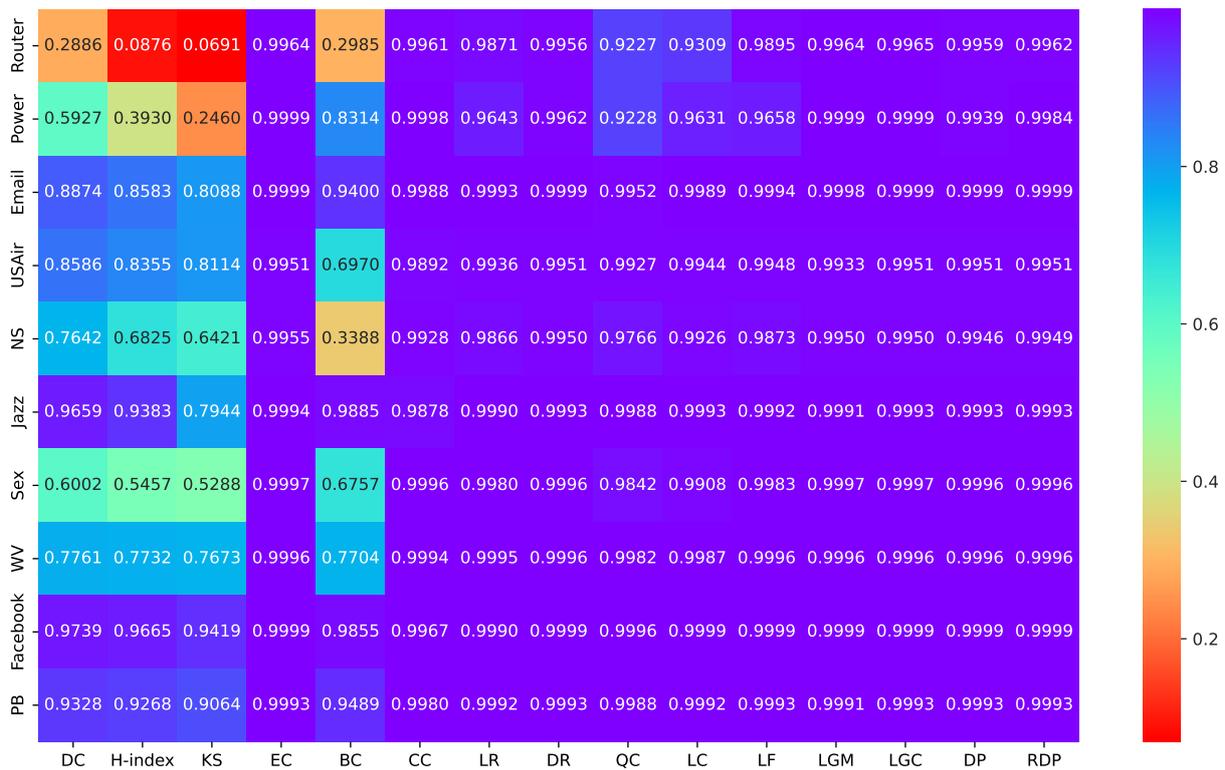


Figure 5. The monotonicity of different algorithms.

5. Discussion

Although our method has performed very well in the above experiments, there are still open issues for future studies. First of all, although we find that the effect of increasing the number of iterations to more than three is not obvious, whether there is an optimal number of iterations is still worth further verification. Secondly, our method is very simple and general, which can be further extended. The initial information of nodes in our method is degree that can be replaced by other centralities (such as H-index) or combinations of centralities (such as the combination of DC and H-index). Our method only uses degree information and some iterations to greatly improve the accuracy of DC. So, our method may also work for other centralities. Notice that, once the centrality based on global information (such as KS) is selected, our method will also become the global method accordingly. Furthermore, the number of iterations required may also change. Finally, the restriction strategy we proposed is relatively simple and rough, which can be further optimized. In view of these aspects, a universal information propagation model can be described as the following Equation (25):

$$UPM(i) = \sum_{l=1}^T R(l) \sum_{j \in \Lambda_i} I_{l-1}(j), \tag{25}$$

where $R(l)$ is the restriction function, and the initial information of node i can be described as the following Equation (26):

$$I_0(i) = S(i), \tag{26}$$

where $S(i)$ is the selected information of node i .

As a result, our suggestions for practical use are as follows:

- The number of iterations (i.e., T) is determined by the size of node neighborhood to be considered;

- The formula of node information (i.e., $S(i)$) can be set according to the needs of the problem, which can contain multi-characteristics of nodes;
- The restriction function (i.e., $R(l)$) can be set according to certain characteristics of the specific network, the growth rate of the results after iteration, or other specific needs.

6. Conclusions

In summary, we propose a novel iterative refinement centrality named degree information propagation (DP), which can obtain the degree information of higher-order neighbors by only considering the directly connected neighbors. Specifically, only a few iterations are needed to be executed, the degree information of nodes' higher-order neighbors can be obtained. Furthermore, we propose a restriction version of DP named restricted degree information propagation (RDP), and experimental results suggest our restriction strategy's effectiveness. In comparison with most local-information-based algorithms, our method is not only accurate, but also extremely low in complexity. No matter whether DP or RDP, it generally achieves the desired results within three iterations. In addition, our method is of great extensibility, with which more factors can be taken into account through proper modification.

Author Contributions: Conceptualization, Z.L.; methodology, Z.L.; software, Z.L. and X.H.; writing—original draft preparation, Z.L. and X.H.; writing—review and editing, Z.L. and X.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: All relevant data are available at <https://github.com/MLIF/Network-Data>, accessed on 10 February 2023.

Acknowledgments: The authors greatly appreciate the reviews' suggestions and the editor's encouragement.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Newman, M.E.J. *Networks*, 2nd ed.; Oxford University Press: Oxford, UK, 2018; pp. 1–11.
2. Wang, X.F.; Li, X.; Chen, G.R. *Network Science: An Introduction*; Higher Education Press: Beijing, China, 2012; pp. 3–27.
3. Lü, L.; Chen, D.; Ren, X.L.; Zhang, Q.M.; Zhang, Y.C.; Zhou, T. Vital nodes identification in complex networks. *Phys. Rep.* **2016**, *650*, 1–63. [[CrossRef](#)]
4. Barabási, A.L. Scale-free networks: A decade and beyond. *Science* **2009**, *325*, 412–413. [[CrossRef](#)]
5. Albert, R.; Jeong, H.; Barabási, A.L. Error and attack tolerance of complex networks. *Nature* **2000**, *406*, 378–382. [[CrossRef](#)] [[PubMed](#)]
6. Pastor-Satorras, R.; Castellano, C.; Van Mieghem, P.; Vespignani, A. Epidemic processes in complex networks. *Rev. Mod. Phys.* **2015**, *87*, 925–979. [[CrossRef](#)]
7. 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](#)] [[PubMed](#)]
8. Malik, H.A.M.; Abid, F.; Wahiddin, M.R.; Bhatti, Z. Robustness of dengue complex network under targeted versus random attack. *Complexity* **2017**, *2017*, 2515928. [[CrossRef](#)]
9. Borge-Holthoefer, J.; Moreno, Y. Absence of influential spreaders in rumor dynamics. *Phys. Rev. E* **2012**, *85*, 026116. [[CrossRef](#)]
10. Cui, A.X.; Wang, W.; Tang, M.; Fu, Y.; Liang, X.; Do, Y. Efficient allocation of heterogeneous response times in information spreading process. *Chaos* **2014**, *24*, 033113. [[CrossRef](#)]
11. Csermely, P.; Korcsmáros, T.; Kiss, H.J.M.; London, G.; Nussinov, R. Structure and dynamics of molecular networks: A novel paradigm of drug discovery: A comprehensive review. *Pharmacol. Ther.* **2013**, *138*, 333–408. [[CrossRef](#)]
12. Sun, P.; Quan, Y.; Miao, Q.; Chi, J. Identifying influential genes in protein-protein interaction networks. *Inf. Sci.* **2018**, *454*, 229–241. [[CrossRef](#)]
13. Puliga, M.; Caldarelli, G.; Battiston, S. Credit default swaps networks and systemic risk. *Sci. Rep.* **2014**, *4*, 6822. [[CrossRef](#)] [[PubMed](#)]
14. Bardoscia, M.; Barucca, P.; Battiston, S.; Caccioli, F.; Cimini, G.; Garlaschelli, D.; Saracco, F.; Squartini, T.; Caldarelli, G. The physics of financial networks. *Nat. Rev. Phys.* **2021**, *3*, 490–507. [[CrossRef](#)]

15. Malik, H.A.M. Complex network formation and analysis of online social media systems. *Cmes-Comp. Model. Eng.* **2022**, *130*, 1737–1750.
16. Ding, Y.; Yan, E.; Frazho, A.; Caverlee, J. PageRank for ranking authors in co-citation networks. *J. Am. Soc. Inform. Sci. Technol.* **2009**, *60*, 2229–2243. [[CrossRef](#)]
17. Su, C.; Pan, Y.; Zhen, Y.; Ma, Z.; Yuan, J.; Guo, H.; Yu, Z.; Ma, C.; Wu, Y. PrestigeRank: A new evaluation method for papers and journals. *J. Inform.* **2011**, *5*, 1–13. [[CrossRef](#)]
18. Albert, R.; Albert, I.; Nakarado, G.L. Structural vulnerability of the North American power grid. *Phys. Rev. E* **2004**, *69*, 025103. [[CrossRef](#)]
19. Motter, A.E. Cascade control and defense in complex networks. *Phys. Rev. Lett.* **2004**, *93*, 098701. [[CrossRef](#)]
20. Bellingeri, M.; Cassi, D.; Vincenzi, S. Increasing the extinction risk of highly connected species causes a sharp robust-to-fragile transition in empirical food webs. *Ecol. Model.* **2013**, *251*, 1–8. [[CrossRef](#)]
21. Bellingeri, M.; Bodini, A. Food web's backbones and energy delivery in ecosystems. *Adv. Ecol.* **2016**, *125*, 586–594. [[CrossRef](#)]
22. Monica, O.; Wahida, F.W.; Fakhruroja, H. The Relations between Influencers in Social Media and the Election Winning Party 2019. In Proceedings of the 2019 International Conference on ICT for Smart Society, Bandung, Indonesia, 19–20 November 2019; pp. 1–5.
23. Lü, L.; Zhou, T.; Zhang, Q.M.; Stanley, H.E. The H-index of a network node and its relation to degree and coreness. *Nat. Commun.* **2016**, *7*, 10168. [[CrossRef](#)]
24. Kitsak, M.; Gallos, L.K.; Havlin, S.; Liljeros, F.; Muchnik, L.; Stanley, H.E. Identification of influential spreaders in complex networks. *Nat. Phys.* **2010**, *6*, 888–893. [[CrossRef](#)]
25. Bonacich, P. Factoring and weighting approaches to status scores and clique identification. *Math. Sociol.* **1972**, *2*, 113–120. [[CrossRef](#)]
26. Freeman, L.C. A set of measures of centrality based on betweenness. *Sociometry* **1977**, *40*, 35–41. [[CrossRef](#)]
27. Freeman, L.C. Centrality in social networks conceptual clarification. *Soc. Netw.* **1979**, *1*, 215–239. [[CrossRef](#)]
28. Chen, D.; Lü, L.; Shang, M.S.; Zhang, Y.C.; Zhou, T. Identifying influential nodes in complex networks. *Phys. A* **2012**, *391*, 1777–1787. [[CrossRef](#)]
29. Chen, D.; Sun, H.L.; Tang, Q.; Tian, S.Z.; Xie, M. Identifying influential spreaders in complex networks by propagation probability dynamics. *Chaos* **2019**, *29*, 033120. [[CrossRef](#)] [[PubMed](#)]
30. Li, Z.; Ren, T.; Ma, X.Q.; Liu, S.M.; Zhang, Y.X.; Zhou, T. Identifying influential spreaders by gravity model. *Sci. Rep.* **2019**, *9*, 8387. [[CrossRef](#)]
31. Ma, Y.; Cao, Z.; Qi, X. Quasi-Laplacian centrality: A new vertex centrality measurement based on Quasi-Laplacian energy of networks. *Phys. A* **2019**, *527*, 121130. [[CrossRef](#)]
32. Zhao, S.; Sun, S. Identification of node centrality based on Laplacian energy of networks. *Phys. A* **2023**, *609*, 128353. [[CrossRef](#)]
33. Zhang, Q.; Shuai, B.; Lü, M. A novel method to identify influential nodes in complex networks based on gravity centrality. *Inf. Sci.* **2022**, *618*, 98–117. [[CrossRef](#)]
34. Zhao, J.; Wu, J.; Xu, K. Weak ties: Subtle role of information diffusion in online social networks. *Phys. Rev. E* **2010**, *82*, 016105. [[CrossRef](#)] [[PubMed](#)]
35. Hao, Y.; Tang, S.; Liu, L.; Zheng, H.; Wang, X.; Zheng, Z. Local-forest method for superspreaders identification in online social networks. *Entropy* **2022**, *24*, 1279. [[CrossRef](#)]
36. Hethcote, H.W. The mathematics of infectious diseases. *SIAM Rev.* **2009**, *42*, 599–653. [[CrossRef](#)]
37. Castellano, C.; Pastor-Satorras, R. Thresholds for epidemic spreading in networks. *Phys. Rev. Lett.* **2010**, *105*, 218701. [[CrossRef](#)]
38. Kendall, M. A new measure of rank correlation. *Biometrika* **1938**, *30*, 81–89. [[CrossRef](#)]
39. Bae, J.; Kim, S. Identifying and ranking influential spreaders in complex networks by neighborhood coreness. *Phys. A* **2014**, *395*, 549–559. [[CrossRef](#)]
40. Spring, N.; Mahajan, R.; Wetherall, D.; Anderson, T. Measuring ISP topologies with rocketfuel. *IEEE/ACM Trans. Netw.* **2004**, *12*, 2–16. [[CrossRef](#)]
41. Watts, D.J.; Strogatz, S.H. Collective dynamics of ‘small-world’ networks. *Nature* **1998**, *393*, 440–442. [[CrossRef](#)]
42. Guimerà, R.; Danon, L.; Díaz-Guilera, A.; Giralt, F.; Arenas, A. Self-similar community structure in a network of human interactions. *Phys. Rev. E* **2003**, *68*, 065103. [[CrossRef](#)]
43. Pajek Datasets. Available online: <http://vlado.fmf.uni-lj.si/pub/networks/data/> (accessed on 1 February 2023).
44. Newman, M.E.J. Finding community structure in networks using the eigenvectors of matrices. *Phys. Rev. E* **2006**, *74*, 036104. [[CrossRef](#)]
45. Gleiser, P.; Danon, L. Community structure in Jazz. *Adv. Complex Syst.* **2003**, *6*, 565–573. [[CrossRef](#)]
46. Rocha, L.E.; Liljeros, F.; Holme, P. Simulated epidemics in an empirical spatiotemporal network of 50,185 sexual contacts. *PLoS Comput. Biol.* **2011**, *7*, e1001109. [[CrossRef](#)] [[PubMed](#)]
47. Leskovec, J.; Huttenlocher, D.; Kleinberg, J. Predicting positive and negative links in online social networks. In Proceedings of the 19th international conference on World Wide Web, Raleigh, North Carolina, USA, 26–30 April 2010; pp. 641–650.

48. Mcauley, J.J.; Leskovec, J. Learning to discover social circles in ego networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 548–556.
49. Adamic, L.A.; Glance, N. The political blogosphere and the 2004 U.S. election: Divided they blog. In Proceedings of the 3rd International Workshop on Link Discovery, Chicago, IL, USA, 21–25 August 2005; pp. 36–43.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.