Received August 17, 2019, accepted September 2, 2019, date of publication September 9, 2019, date of current version September 25, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2939804

Identifying Influential Nodes in Complex Networks Based on Local Neighbor Contribution

JINYING DAI[1], BIN WANG[1], JINFANG SHENG[1], ZEJUN SUN[1,2], FAIZA RIAZ KHAWAJA[1], AMAN ULLAH[1], DAWIT AKLILU DEJENE[1], AND GUIHUA DUAN[1]
[1]School of Computer Science and Engineering, Central South University, Changsha 410083, China
[2]Department of Network Center, Pingdingshan University, Pingdingshan 467000, China

Corresponding author: Jinfang Sheng (jfsheng@csu.edu.cn)

This work was supported by the National Science and Technology Major Project of China under Grant 2017ZX06002005.

ABSTRACT The identification of influential nodes in complex networks has been widely used to suppress rumor dissemination and control the spread of epidemics and diseases. However, achieving high accuracy and comprehensiveness in node influence ranking is time-consuming, and there are issues in using different measures on the same subject. The identification of influential nodes is very important for the maintenance of the entire network because they determine the stability and integrity of the entire network, which has strong practical application value in real life. Accordingly, a method based on local neighbor contribution (LNC) is proposed. LNC combines the influence of the nodes themselves with the contribution of the nearest and the next nearest neighbor nodes, thus further quantifying node influence in complex networks. LNC is applicable to networks of various scales, and its time complexity is considerably low. We evaluate the performance of LNC through extensive simulation experiments on seven real-world networks and two synthetic networks. We employ the SIR model to examine the spreading efficiency of each node and compare LNC with degree centrality, betweenness centrality, closeness centrality, eigenvector centrality, PageRank, Hyperlink-Induced Topic Search(HITS), ProfitLeader, Gravity and Weighted Formal Concept Analysis(WFCA). It is demonstrated that LNC ranks nodes effectively and outperforms several state-of-the-art algorithms.

INDEX TERMS Complex networks, influential nodes, local structure, neighbor contribution.

I. INTRODUCTION

Complex networks are powerful methods for representing and studying the interactions among objects in the real world, it is an abstraction of complex systems. The topology of complex networks determines their node influence [1]. Recently, complex network mining has attracted significant attention [2]–[4]. In several studies, a node with greater propagation capability is regarded as influential, that is, it can spread a message to a significant number of network users [5]. Influential nodes contain more global or local network information compared with other nodes. Therefore, determining the propagation capability of nodes and identifying influential nodes are highly important for successful message propagation in social networks [6]. In addition to its theoretical significance, influential node mining in complex networks has various practical applications. For example, as the scale of the national power grid continues to expand, its structure becomes more complex, and the disconnection of several main trunks would lead to the collapse of the entire network [7]. We can predict and control hidden problems in the power grid and thus avoid economic loss only if we understand the network structure in advance. Identification of influential nodes has wide application in various areas, as it can be used to hold back the spread of viruses [8], suppress disease difussion [9], isolate disease sources [10], distinguish key personnel or information [11], [12] and rank web pages according to their importance and relevance to a query. With the development of complex network science and the continuous expansion of study fields, complex networks have been widely used in Economics [13], Chemistry [14], Biology [15], [16], and other fields [17], [18]. Determining influential nodes has great theoretical significance for optimizing network structure, enhancing the

The associate editor coordinating the review of this manuscript and approving it for publication was Alberto Cano.

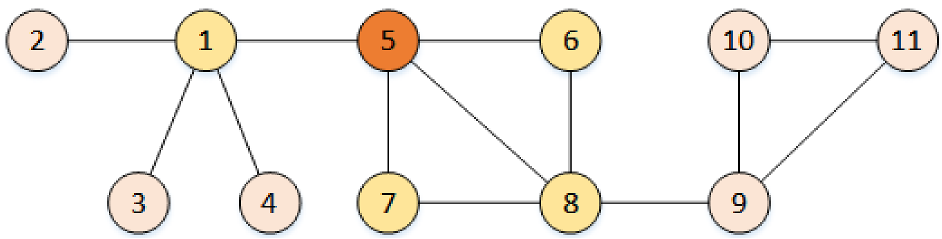

FIGURE 1. Example network containing 11 nodes and 13 edges. The network will be disconnected if v_1, v_5, v_8, and v_9 are removed. Their ranks are consistent if the node removal method is used to determine node influence. However, intuitively, there are significant differences in the influence of these nodes.

robustness of network architecture, and understanding the dynamics of information dissemination [19], [20]. Designing fast and effective influential node mining methods for large-scale dynamic networks is an urgent priority.

In the past few decades, several methods for identifying influential nodes in complex networks have been proposed, such as degree centrality [21], closeness centrality [22], betweenness centrality [23], PageRank [24], Leader Rank [25], H-index [26], and HITS [27]. However, many methods have common shortcomings. Simpler methods are inaccurate, whereas accurate methods have high computational complexity. For instance, methods based on global or local structure. Typical methods based on local information focus on the most essential attributes of a node and cannot fully reflect the influence of the node on the network. Degree centrality is a typical method based on local information. It is a direct indicator of single node importance, but it ignores a key factor: a node with few high-impact neighbors is more important than a node with a large number of low-impact neighbors [28], [29]. That is, degree centrality considers only the influence of the nearest neighbors of a node. The methods based on global information consider the global structure of the network, but they relatively difficult to apply in some specific networks. For example, betweenness centrality and closeness centrality are not suitable for large-scale networks owing to their high time complexity. PageRank, a web influence indicator based on global information, performs well in directed networks. The expeditious and accurate identification of influential nodes is an important research topic, particularly when the network scale reaches tens of millions or even billions of nodes. In short, accurate and efficient mining of influential nodes is still an open question [30].

In this paper, we propose a new method, called LNC, for detecting influential nodes in complex networks of various scales. Although LNC is based on local information, the difference from traditional methods is that it considers the factors affecting node influence from different perspectives, its effectiveness and accuracy will be discussed in Section III. We first introduce the basic principle of the proposed method.

A. BASIC IDEA

The influence of a node in a network is mainly measured by its ability to influence other nodes. If the entire society in a certain country is seen as a complex network, state leaders, successful scholars, great scientists, or famous celebrities, who are well known in daily life, can be regarded as influential nodes in the network and promote the development of the country in different respects. What makes these individuals stand out is not only their own superior abilities but also their surroundings [31]. Therefore, the influence of a node can be determined by the node itself as well as the influence of other nodes on it. Accordingly, we propose the LNC method to rank the influence of nodes in complex networks. There are two main factors in this method. One is the influence of each node itself, which is mainly measured by its ability to influence other nodes. Furthermore, we consider the contribution of other nodes in the network on a specific node. Neither the contribution of other nodes nor the influence of the node itself alone can fully reflect the importance of the node in the entire network, thus, their combination is necessary.

TABLE 1. Use LNC method to get the influence rank and calculate the influence of each node.

Node	5	8	1	9	6	7	11	10	4	3	2
Influence	32.4	29.7	18.9	9.6	3.2	3.2	2.0	2.0	0.4	0.4	0.4

To illustrate the basic principle of LNC further, we consider an example in detail. Fig. 1 shows a network G that contains 11 nodes and 13 edges. First, we analyze the influence of each node in the network. According to the ranking methods based on node removal, the network cannot be connected if the nodes v_1, v_5, v_8, and v_9 are removed. That is, the number of spanning trees is 0 if these nodes are removed, and thus it appears that these four nodes have the same influence. However, intuitively, they differ significantly in their influence rank. First, v_5 and v_8 are located at the core of the network, thus, they should be the most influential. Furthermore, the degree of v_1 is larger than that of v_9, and the removal of v_1 has greater impact on network connectivity, thus v_1 is more important than v_9. Finally, the peripheral nodes v_2, v_3, and v_4 are on the edge of the network and have little effect, thus, they are the least influential. Table 1 shows that the results by LNC are consistent with the theoretical results. The specific calculation will be described in Section III.

Several factors should be considered for accurately determining node influence in complex networks. In the LNC method, the contributions of the nearest and the next nearest neighbor nodes are indispensable. The nearest neighbor nodes have a direct influence on a node, whereas the next nearest neighbor nodes have indirect effects that cannot be ignored. We will further evaluate the performance of the proposed method in real-world complex networks in Section IV.

B. CONTRIBUTIONS

The main contributions of LNC are described as follows.

- **Intuitive and effective influential node detection:** We consider the factors that affect the importance of nodes from two different perspectives, and node influence is measured through the combination of the influence contributions of the nodes themselves, the nearest neighbor nodes, and the next nearest neighbor nodes. This increases identification accuracy to some extent.
- **Scalability:** Compared with other methods, LNC can effectively and accurately identify influential nodes and greatly reduce computational cost (Fig. 2–Fig. 4, Table 4–Table 6). It is computationally simple and has low time complexity (the specific time complexity will be explained in Section III), thus, it is suitable for large-scale networks.
- **Parameter-Free:** LNC does not rely on prior knowledge and parameter adjustments but can automatically identify influential nodes.

The remainder of this paper is organized as follows. In Section II, we provide a brief survey of related work. Section III presents the LNC method in detail. In Section IV, we present a performance evaluation of LNC based on nine networks in terms of several widely used metrics. We finally conclude the paper in Section V.

II. RELATED WORK

In the past few decades, several methods (e.g., degree centrality [21], K-shell [32], closeness centrality [22], betweenness centrality [23], eigenvector centrality [33], PageRank [24], nodal contraction, and betweenness centrality with weight) have been proposed for identifying influential nodes in complex networks, and all these methods have their advantages and disadvantages. We provide a brief overview of these methods below.

Ranking methods based on neighbor nodes. These indicators are simple and intuitive, and they have low time complexity. Degree centrality and K-shell are two representative methods based on neighbor nodes. Degree centrality measures the influence of a node by the number of neighbor nodes. It is the simplest indicator for characterizing influential nodes. Its disadvantage is the lack of consideration of the global network structure and the influence of the surrounding nodes, therefore, in several cases, it is not sufficiently accurate. K-shell is a coarse-grained ranking method [34], in which a node is usually considered to have a higher influence if it is situated in the core position of the network even if its degree is small. The influence of large-degree nodes on the edge is often limited [35]. Although this method has low time complexity, it is not suitable for certain types of networks, such as rule or BA networks [36], and the ranking result is coarse-graining because it is difficult to distinguish the influence of nodes in the same layer.

Ranking methods based on the shortest path. These methods assume that the information in a network flows only through the shortest path. Representative methods are closeness centrality and betweenness centrality. Closeness centrality reflects the degree of closeness between any two nodes in the network. It uses the relative distance between each pair of nodes to quantify their centrality in the entire network [37]. Betweenness centrality assumes that information flows along the shortest path [38] and measures saliency by the number of shortest paths that pass through a node. Closeness centrality and betweenness centrality are based on global structure and can effectively identify influential nodes. However, their computational complexity is high, and thus they cannot be applied to large-scale or complex networks [39].

Ranking methods based on eigenvectors. These methods consider not only the number of neighbor nodes but also their influence. Eigenvector centrality and PageRank are representative methods based on eigenvectors. Eigenvector centrality can be efficiently calculated using a power iteration approach, but it may become trapped in a zero status owing to the presence of several nodes without in-degree [26]. PageRank is a well-known web page ranking algorithm that is used in the Google search engine. It ranks based on the link structure of web pages and assumes that the influence of a web page is determined by both the quality and the number of the pages linked to it. PageRank has been widely used in various areas. However, it is sensitive to random network disturbances and exhibits topic drifts in special network structures [26].

Ranking methods based on node contraction. These methods consider the influence of a node to be equivalent to the destructiveness of the network after the node is removed. The network is significantly more cohesive after a node contracts if the node is influential. In this method, the influence of a node is determined by the number of its neighbors and its location in the network. As the average path length should be calculated for each node contraction, the time complexity is high. Thus, this method is not suitable for large-scale networks [40].

Ranking methods based on node centrality in weighted networks. These methods use edge weights so that the structure and function of a network may be understood more comprehensively. A representative method is betweenness centrality with weight. In a weighted network, the path length between nodes is determined by edge weights. Specifically, path length is measured by the reciprocal of the edge weight.

In conclusion, several methods have been proposed for identifying influential nodes in complex networks. To some extent, owing to the different structure of real-world complex systems, each method has its advantages and disadvantages. Effective and efficient identification of influential nodes remains a non-trivial task. Here, we propose an effective method that can be applied to networks of various scales.

III. THE LNC MODEL

We first introduce some basic concepts and definitions concerning LNC in Section 3.1. The calculation of LNC is presented in Section 3.2, and Section 3.3 analyzes the time complexity of LNC in detail.

A. PRELIMINARIES

Before explaining the proposed algorithm, we will formalize some of the basic definitions that will be used in the following sections.

Definition 1 (Degree Centrality): Given a network $G = (V, E)$, degree refers to the number of relationships between a node and other nodes in the network. The degree of a node v_i is divided by the maximum number of possible connections with other nodes to obtain the proportion of nodes directly related to v_i, which is the degree centrality. It is denoted by $DC(v_i)$, and is defined as follows:

$$DC(v_i) = \frac{d(v_i)}{(n-1)} \tag{1}$$

where n is the total number of nodes, and $d(v_i)$ represents the degree of v_i. In real-world applications, each node has a different influence. Degree centrality describes the direct influence of v_i, and greater degree implies that the node is more important.

Definition 2 (Contribution Probability): Given a network $G = (V, E)$, a node v_i is randomly connected to any of its neighbors, and its degree represents all possible connections. We take the reciprocal of the degree is defined as the contribution probability of the node v_i. It is denoted by $P(v_i)$, and is defined as follows:

$$P(v_i) = 1/d(v_i) \tag{2}$$

where the degree of v_i is used as an indicator of its influence.

Definition 3: (Cluster Degree) Given a network $G = (V, E)$, the degree sum of all neighbor nodes of the node v_i is called cluster degree and is defined as follows:

$$D(v_i) = \sum_{v_j=\eta(v_i)} d(v_j) \tag{3}$$

where $\eta(v_i)$ is the set of the nearest neighbors of v_i.

Definition 4: (Contribution) Given a network $G = (V, E)$, the influence of a node depends partly on its surroundings. The contribution of the nearest and the next nearest neighbor nodes is called contribution and is defined as follows:

$$neiCon(v_i) = D(v_i) \sum_{j=1}^{k} P(v_j) DC(v_j) \tag{4}$$

where k denotes the number of the nearest neighbor nodes and the next nearest neighbor nodes, and v_j represents the neighbor nodes of the node v_i.

Definition 5: (Own Influence) Given a network $G = (V, E)$, here, we assume that information transfers between nodes with equal probability. Therefore, the random selectivity probability describing the contribution ability of the node v_i is $P(v_i)$. The influence of the node v_i itself is denoted by $ownCon(v_i)$ and defined as follows:

$$ownCon(v_i) = d(v_i) \sum_{j=1}^{k} C(k, 1) P(v_i)^1 (1 - P(v_i))^{(k-1)} \tag{5}$$

Each node may choose to connect with any neighbor node.

B. THE LNC MODEL

The proposed method mainly considers two key factors affecting node influence. One is an indicator of the node's own influence, and the other is the contribution of the nearest and the next nearest neighbor nodes. In Fig. 1, if we take the node v_5 as an example, then according to the proposed algorithm, the influence of v_5 depends on the contribution of the nearest and the next nearest neighbors as well as the influence of v_5 itself. The method for identifying influential nodes is to simulate various complex systems in reality by analyzing the network topology and various characteristic node attributes. We consider detecting influential nodes by LNC. The graph G in Fig. 1 is a synthetic network containing 11 nodes and 13 edges. The main process of LNC can be divided into four steps. First, we compute the sum of neighbor node degrees. Subsequently, we calculate the contribution of the neighbor and the next neighbor nodes. Then, we calculate each node's own influence. Finally, we calculate the influence of each node in the network.

1) COMPUTATION OF THE SUM OF NEIGHBOR NODE DEGREES

To control the time complexity of LNC, we select degree as the fundamental node influence indicator, and the contribution probability is set to be the reciprocal of the node degree. Here, we assume that the information in the network flows randomly. The contribution probability and the degrees of all neighbor nodes are obtained by formula(2). We now present the influence calculation for the node v_5 in detail. First, v_5 has four nearest neighbor nodes, that is, v_1, v_6, v_7, v_8, and six next nearest neighbors, that is, v_2, v_3, v_4, v_6, v_7, v_9. Thus, by formula(3), $D(v_5) = \sum_{v_j=\eta(v_5)} d(v_j) = d(v_1) + d(v_6) + d(v_7) + d(v_8) = 12$. This serves as a fundamental measure of the influence of neighbor nodes.

2) CALCULATION OF THE CONTRIBUTION OF THE NEAREST AND THE NEXT NEAREST NEIGHBOR NODES

The contribution of each neighbor node to a certain node is measured by the influence of these nodes themselves. In this part, we consider the nearest and the next neighbor nodes of a node in the network. The degree of a nearest neighbor node is the number of the next nearest neighbor nodes. We calculate the degree of each nearest neighbor node and take its reciprocal as the propagation probability. By formula(4), we calculate the contribution of each neighbor node separately and then obtain the total contribution. By analyzing these two aspects, we obtain the influence contribution made by all neighbors to node v_5. Therefore,

$$neiCon(v_5) = D(v_5)\sum_{j=1}^{n} P(v_j)DC(v_j) = 19.2$$

3) CALCULATION OF EACH NODE'S OWN INFLUENCE

To set a unified standard for the influence of a node, we still consider degree the basic influence value. We observed that we can determine the influence of a node by its ability to influence other nodes. This implies that an influential node randomly passes information to neighbor nodes. It is also assumed that the information is spread among nodes with equal probability. The influence of a node is calculated by its influence on other nodes. We consider the node v_5 in Fig. 1. It can randomly affect any neighbor node, and we assume that the propagation probability of a node randomly selecting its neighbor node is the reciprocal of the node degree. It can be seen that v_1, v_6, v_7, v_8 are neighbor nodes of v_5, and therefore by formula(5), the influence of the node itself is

$$ownCon(v_5) = d(v_5)\sum_{j=1}^{4} C(4,1)P(v_5)^1(1-P(v_5))^3 = 1.6875$$

4) CALCULATION OF THE INFLUENCE OF THE NODES IN THE NETWORK

The contribution of the nearest and the next nearest neighbor nodes is combined with the influence of the node itself. Then, the influence of all nodes in the network is obtained. It is defined as follows:

$$Influ(v_i) = neiCon(v_i)ownCon(v_i) \tag{6}$$

The influence of v_5 is as follows, and the influence of the other nodes is shown in Table 1.

$$Influ(v_5) = neiCon(v_5)ownCon(v_5) = 32.4$$

An implementation of the LNC algorithm is shown in Algorithm 1.

C. TIME COMPLEXITY

One of the advantages of LNC is its low time complexity, which has three main components. In the first step, to calculate the influence of all nodes in the network, the algorithm should identify the nearest and the next neighbor nodes of each node. The time complexity for computing the nearest neighbor nodes is $O(<k>n)$, where $<k>$ is the average number of nearest neighbor nodes, and n is the total number of nodes in the network. In the second step, the nearest and the next neighbor nodes are considered. LNC should calculate the degrees of the nearest neighbor nodes and their contributions. Thus, the time complexity of this part is $O(<k>n)$. In the third step, the influence of a node itself is measured based on its degree. The time complexity of obtaining the degree of each node in the network is $O(n)$. Hence, the computational complexity of LNC is $O(<k>n)$. We note that $k << n$, and thus the LNC algorithm can handle large-scale networks.

Algorithm 1 LNC

Input:
Graph:$G = (V, E)$
// The contributions of neighbor nodes
for each node v in V **do**
  for each node u in $N(v)$ **do**
    compute $DC(u)$ using (1)
    compute $P(u)$ using (2)
    compute $D(v)$ using (3)
    compute $neiCon(v)$ using (4)
  end for
end for
// The influence of nodes themselves.
for each node v in V **do**
  n = G.degree(v)
  compute $P(v)$ using (2)
  compute $ownCon(v)$ using (5)
end for
compute $Influ(v)$ using (6)
// Return *Influ*
Output: $Influ(v)$

IV. EXPERIMENTS

In this section, we present the results of experiments conducted using seven real-world networks and two synthetic networks to demonstrate the performance of LNC based on comparisons with several other methods. Before presenting the experimental results, we briefly introduce the methods used in the comparisons.

A. COMPARISON METHOD DESCRIPTION

Degree Centrality(DC) is the simplest indicator for describing node influence. Nodes with high degree have higher influence than nodes with lower degree. For example, in Fig. 1, v_5, as the initially infected node, spreads information faster and more widely than v_6. However, degree centrality considers only limited information and is not effective in some cases.

Betweenness Centrality(BC) assumes that the information flow propagates along the shortest path. On the shortest path between all node pairs, the influence of a node is proportional to the number of shortest paths passing through it. Betweenness centrality calculates node influence based on global information. However, this is complicated, as it requires not only calculating the shortest path length between each pair of nodes but also recording these shortest paths.

Closeness Centrality(CC) identifies influential nodes based on global information. Closeness centrality uses the relative distance between each pair of nodes to determine node centrality, which is widely used in research. Closeness centrality effectively resolves the issue of node contraction and reduces the complexity of directly calculating betweenness centrality. It can also be understood as the use of the average propagation time of network information for determining node influence, however, it has high time complexity.

Eigenvector Centrality(EC) assumes that the influence of a node depends on both the number of neighbor nodes (the degree of the node) and the influence of each neighbor node. It evaluates the influence of a node by the information of the other nodes connected to it. This approach has attracted great attention both in theory and in practice.

HITS(Hyperlink-Induced Topic Search) uses different metrics simultaneously. HITS evaluates the influence of each node using authority and hub. The authority value measures the original creativity of the node with respect to the information, and the hub value reflects the role of the node in the information transmission. They interact with each other and converge through iteration.

PageRank ranks web pages based on their link structure. It assumes that the influence of a web page depends on the number and quality of other pages pointing to it. If a page has a large number of high-quality pages pointing to it, then its quality is also high. It performs well in directed networks but cannot be applied in undirected networks [41].

ProfitLeader is the latest of these comparison methods and was proposed in May 2018. This algorithm ranks key nodes in networks by quantifying the profit that a node can make. Its calculation is relatively simple, and it is suitable for large-scale networks [31].

Gravity Centrality(GC) viewing the k-shell value of each node as its mass and the shortest path distance between two nodes as their distance, then inspired by the idea of the gravity formula, the author proposed a gravity centrality index to identify the influential spreaders in complex networks.

Weighted Formal Concept Analysis(WFCA) is a typical computational intelligence technique. This model converts the binary relationships between nodes in a given network into a knowledge hierarchy, and employs WFCA to aggregate the nodes in terms of their attributes. The more nodes aggregated, the more important each attribute becomes.

B. DATA DESCRIPTION

In this section, we evaluate the proposed method on two synthetic networks and seven real-world networks to demonstrate its performance. These data sets are selected from different fields, and their network structures and scales are also various. From the experiment on the example network, we can see that LNC exhibits high performance. Furthermore, we use several well-known real-world networks with various sizes and characteristics to assess the performance of these methods. The statistics for them are summarized in Table 2. We now briefly introduce these networks, they are all publicly available from http://konect.uni-koblenz.de/networks/arenas-email (Karate, Email, Friendship, Powergrid, Caida, Douban) and http://networkrepository.com (Ca-Csphd).

TABLE 2. Statistics of seven real-world networks and two synthetic networks: node number $|V|$, edge number $|E|$, the average degree $< K >$, maximum degree K_{max}, and clustering coefficient $< CC >$.

Data Sets	$\|V\|$	$\|E\|$	$< K >$	K_{max}	$< CC >$
Karate	34	78	4.59	17	0.5706
Email	1133	5451	9.62	71	0.2202
Friendship	1858	12534	5.76	85	0.167
Ca-Csphd	1882	1740	1.849	46	0.005
Random	3000	47475	30.32	54	0.0102
Powergride	4941	6594	2.669	19	0.0801
Caida	26475	53381	4.033	2628	0.208
BA	30000	119984	7.992	668	0.0025
Douban	154908	327162	4.224	287	0.016

1) SYNTHETIC NETWORKS

Many complex networks exist in the real world but we don't know the ground truth details about all of them. Therefore, in order to evaluate LNC and the methods used in the comparison, we built comprehensive networks with different scales.

a: RANDOM NETWORK

This is a synthetic random network with 3000 nodes connected by probability 0.01, in other words, the average degree of random network is 3, and it contains 47,475 edges.

b: BA NETWORK

This is a synthetic scale-free network with 30000 nodes, and 4 edges added each time, and it contains 119984 edges.

2) REAL-WORLD NETWORKS

There are many real-world networks in our life. In this section, we use seven real networks to verify the effectiveness of the LNC algorithm.

a: KARATE NETWORK [42]

This is a well-known network that has been widely used for influential node mining in complex networks. It is a friendship network among the 34 members of a karate club at an American university. It consists of 34 nodes and 78 edges, where nodes denote members of Zachary's karate club at an American university and edges represent the friendship between members.

b: EMAIL NETWORK [43]

This is an email communication network from a university in Tarragona, Spain. It contains 1133 nodes and 5452 edges. Nodes represent users, and each edge indicates that at least one email was sent.

c: FRIENDSHIP NETWORK [44]

This network represents the friendship between any two users on hamsterster.com. Nodes represent users and edges denote the closeness between users.

d: POWERGRID NETWORK [45]

This undirected network contains information about the power grid of the western states of the United States of America. It has 4941 nodes and 6594 edges. A node is either a generator, a transformer, or a substation, and an edge is a power supply line.

e: CA-CSPHD NETWORK [46]
The network consists of 1882 nodes and 1740 edges. This is a popular network and it has been used widely for complex network influential nodes mining.

f: CAIDA NETWORK [47], [48]
This is an undirected network of autonomous systems of the Internet connected with each other from the CAIDA project, collected in 2007. Nodes are autonomous systems, and edges denote communication.

g: DOUBAN NETWORK [49]
This is the social network of Douban, a Chinese online recommendation site. The network is undirected and unweighted. It is a large-scale and complex network, it consists of 154908 nodes and 327163 edges.

C. EVALUATION METRICS
In this study, we employ the SIR model [50] to investigate the spreading influence of ranked nodes. There are three states in the SIR model: (i) Susceptible (S) denotes susceptible individuals who are not yet infected. (ii) Infected (I) represents infected individuals, who may spread the disease to susceptible individuals. (iii) Recovered (R) denotes recovered individuals, who can never be infected again. When an individual has experienced a complete infection cycle, it will never be re-infected, thus, the individual's state can be ignored. To measure the spreading capability of the nodes, in each implementation, only one node is selected to be infected, whereas the other nodes are set as susceptible at each independent run. The seed node infects its neighbor nodes with a certain probability, and infected nodes recover with some other probability. Each loop is treated as a time step t, and $F(t)$ indicates the number of nodes infected and recovered at time t, which is used to assess the influence of the initially infected node. Obviously, the cumulative number of infected nodes gradually converges with time and eventually reaches a steady state. The numbers of infected and recovered nodes indicate the impact capability of the seed node.

To evaluate the performance of the methods, Kendall τ [51] is introduced to measure the correlation of the spreading influence of the nodes by the ten methods. Kendall τ as a rank correlation coefficient is usually used to measure the correlation between two ranking lists. We assume that two queues X and Y with the same number of elements n, $X = (x_1, x_2, ..., x_n)$ and $Y = (y_1, y_2, ..., y_n)$. Any pair of pairs (x_i, y_i) and (x_j, y_j) $(i \neq j)$ are said to be concordant if the ranks for both elements agree, that is, if both $x_i > x_j$ and $y_i > y_j$ or $x_i < x_j$ and $y_i < y_j$. They are said to be discordant if $x_i > x_j$ and $y_i < y_j$ or if $x_i < x_j$ and $y_i > y_j$. If $x_i = x_j$ or $y_i = y_j$, the pair is neither consistent nor inconsistent.

TABLE 3. **Ranking by LNC and other methods, where the last two columns are ranked by the SIR model. Here, PL, PR, Gr, LNC' and SIR' denote ProfitLeader, PageRank, Gravity, LNC value, and SIR value, respectively.**

BC	CC	DC	EC	HITS	PR	PL	Gr	WFCA	LNC	LNC'	SIR	SIR'
5	5	1	8	1	1	5	8	5	5	32.4	5	1.536
1	8	5	5	5	5	8	5	1	8	29.7	8	1.501
8	1	8	6	8	8	1	9	9	1	18.9	1	1.462
9	6	9	7	9	9	9	6	10	9	9.6	9	1.361
2	7	6	9	6	10	10	7	11	6	3.2	6	1.296
3	9	7	1	7	11	11	10	6	7	3.2	7	1.282
4	2	10	10	10	6	6	11	7	10	2.0	11	1.257
6	3	11	11	11	7	7	1	2	11	2.0	10	1.236
7	4	2	2	3	2	3	2	3	2	0.4	4	1.142
10	10	3	3	4	3	2	3	4	3	0.4	3	1.118
11	11	4	4	2	4	4	4	8	4	0.4	2	1.104

Kendall τ coefficient is defined as follows:

$$\tau(X, Y) = \frac{n_c - n_d}{0.5n(n-1)} \tag{7}$$

Here, n_c and n_d indicate the number of concordant and discordant pairs, respectively. This coefficient reflects the correlation and matching between two methods. In general, τ is in $[-1, 1]$, where $\tau > 0$ indicates positive correlation, whereas $\tau < 0$ indicates negative correlation. That is, higher t values imply a more accurate ranking list.

D. PERFORMANCE EVALUATION
In this experiment, to distinguish node influence and verify the effectiveness of the proposed method, the SIR model was used to verify the accuracy of LNC. Using the network in Fig. 1 as an example, Table 3 shows the ranking results for each node by the ten methods and lists the results obtained by LNC and the SIR model. Kendall τ was used to detect the veracity and reliability of LNC. Fig. 2 shows the Kendall τ of the LNC method, where the ranking lists are generated by the BC, CC, DC, EC, HITS, PageRank, ProfitLeader, Gravity and WFCA. It can be seen that most Kendall τ for the LNC method is between 0.8 and 1(such as Karate, Ca-CSphd, Friendship, Random, Email, Powergrid, Caida and Douban), which is the largest in most networks, indicating that the ranking lists generated by LNC and the SIR spreading process are essentially identical. Fig. 2 shows that performance varies among the methods, and LNC performs well on networks of different scales. We note that the performance of the BC method is nearly always the worst on all networks because BC is generally based on the definition of the shortest path, but information in most networks does not flow along the shortest paths. In addition, we have noticed that on the BA network, Kendall τ of all methods is less than 0. Although LNC algorithm is not the best, it still has certain advantages compared with some algorithms.

To investigate the performance of the LNC method further, we consider the spreading influence of the ranked nodes in the SIR model. To distinguish the influential nodes more clearly, we choose a relatively small λ for the large data sets (Powergrid, Caida, BA and Douban). Specifically we set $\lambda = 0.01$ because with a larger value, the propagation would

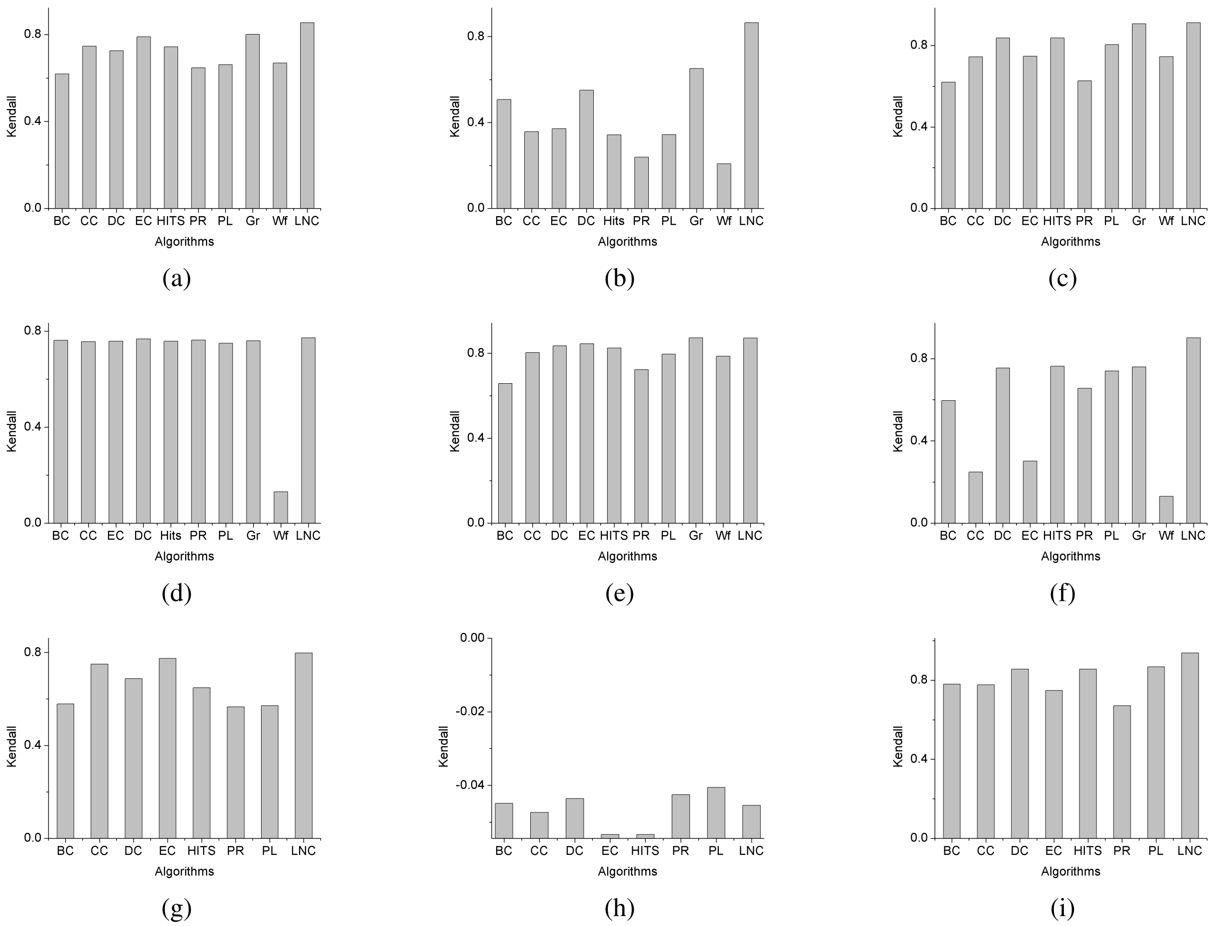

FIGURE 2. Kendall τ is obtained by comparing the rankings by the ten methods and the SIR model. Networks with propagation probability $\lambda = 0.1$ include (a) Karate. (b) Ca-Csphd. (c) Friendship. (d) Random. (e) Email. Those with $\lambda = 0.01$ are (f) Powergrid. (g) Caida. (h) BA. (i) Douban. Here, PL, PR, Gr and Wf denotes ProfitLeader, PageRank, Gravity and WFCA, respectively.

TABLE 4. Rankings by the ten methods. Owing to space limitations, only the top 10 nodes of two networks are shown: (a) Friendship (b) Email, where PL and PR denote ProfitLeader and PageRank, respectively.

(a)

BC	CC	DC	EC	HITS	PR	PL	Gravity	WFCA	LNC
237	237	237	237	237	237	237	237	237	237
137	137	238	238	238	238	238	238	238	238
169	238	168	168	168	169	168	168	356	44
238	177	137	356	356	44	137	137	168	168
251	176	169	44	44	137	356	44	137	137
44	118	45	177	177	168	44	177	244	45
296	178	46	137	137	45	244	176	177	46
168	168	176	244	244	46	177	46	649	176
23	3	177	176	176	65	458	3	44	177
65	117	47	3	3	87	46	356	158	169

(b)

BC	CC	DC	EC	HITS	PR	PL	Gravity	WFCA	LNC
333	333	105	105	105	105	105	105	299	105
105	23	333	16	333	23	16	333	434	333
23	105	16	196	42	333	42	42	552	42
578	42	23	204	16	41	196	23	389	16
76	41	42	42	23	42	3	76	726	23
233	76	41	49	41	16	333	41	756	41
135	233	196	56	196	233	299	233	571	196
41	52	233	116	233	355	49	196	886	233
355	135	21	333	21	21	41	52	888	76
42	378	76	3	76	24	46	3	788	21

occur across nearly the entire network [32], in which case it would be difficult to distinguish the influence of different nodes. For the small data sets (Karate, Email, Friendship, Ca-CSphd, and Random), we set $\lambda = 0.1$ to evaluate the influence of each node so that we can obtain the propagation efficiency of all nodes in the network, otherwise, we set the recovery probability $\mu = 1$ and the time step $t = 500$. First, we compute the influence of each node using the various methods and rank them in descending order. Table 4 presents the top 10 ranked nodes. Owing to space limitations, we only show the top 10 nodes of two networks: Friendship and Email. It can be seen that most top 10 nodes of LNC are also obtained by other methods. Therefore, the validity of the LNC method is verified. Secondly, each node is considered a seed node for influencing other nodes. Finally, we obtain the number of nodes successfully infected by the seed nodes by computing the average over 1000 turns. The results are shown in Fig. 3.

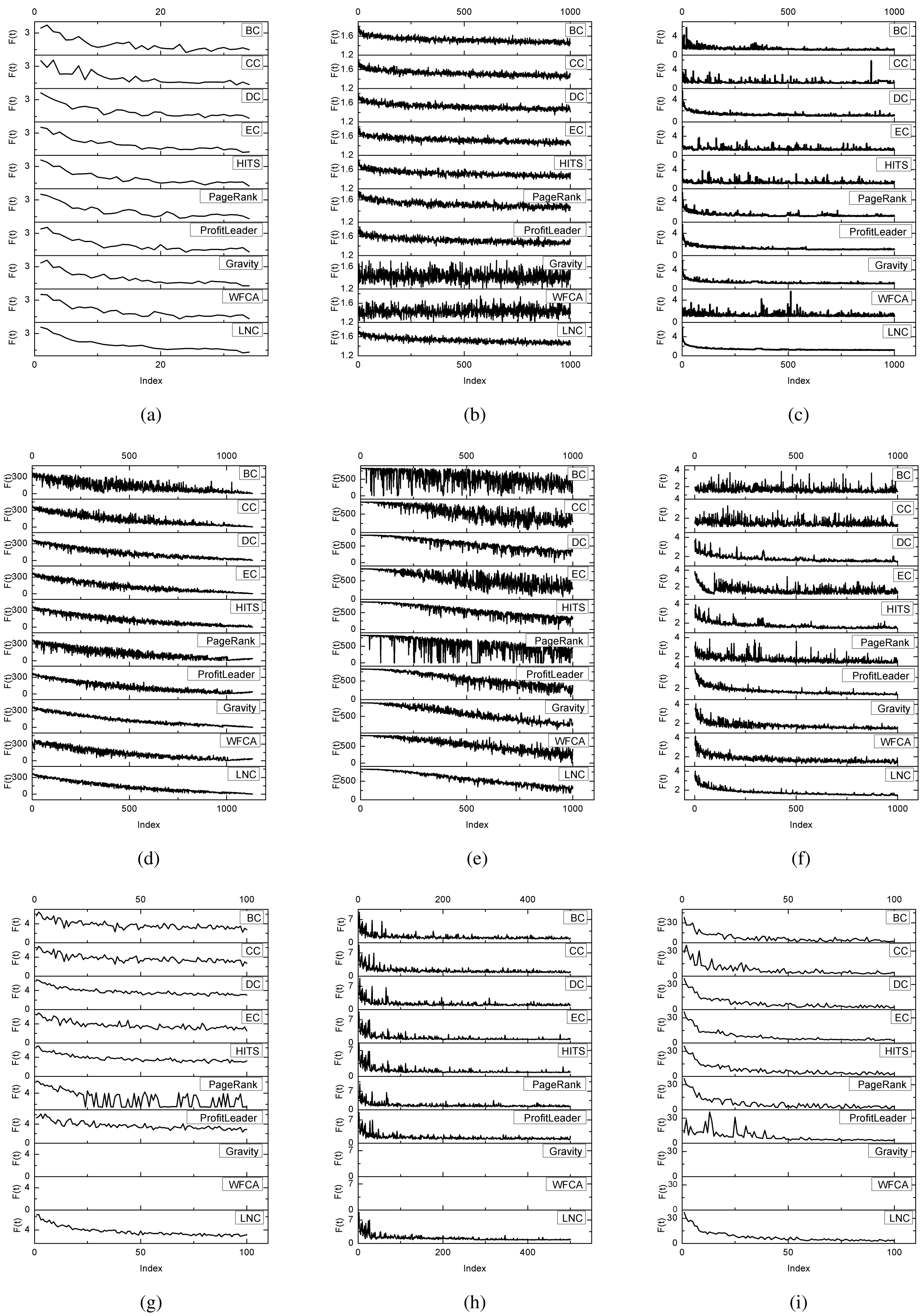

FIGURE 3. Propagation influence for the rankings by the ten methods, F(t) denotes the number of infected and recovered nodes at time t, and ranked index represents the order of the ranking. (a) Karate. (b) Random. (c) Ca-CSphd. (d) Email. (e) Friendship. (f) Powergird. (g) Douban. (h) BA. (i) Caida.

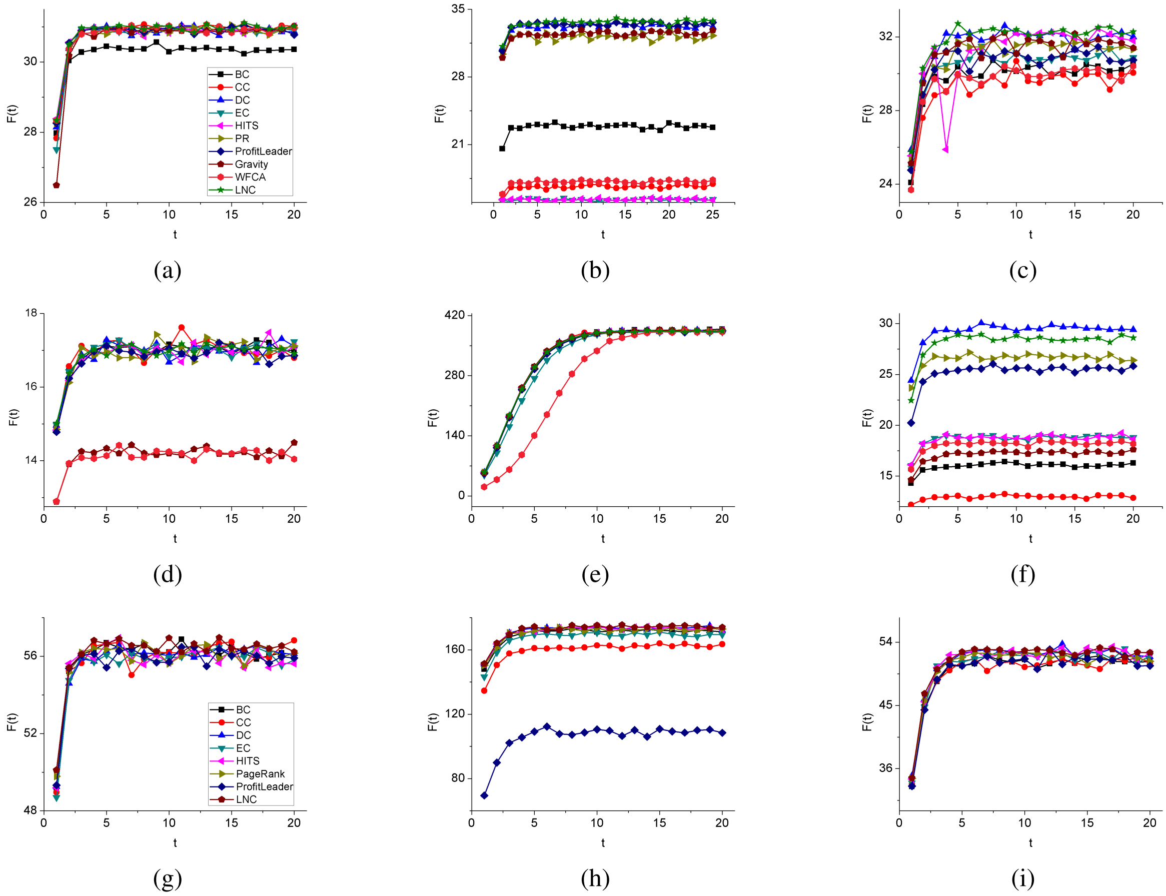

FIGURE 4. Propagation influence of top 10 nodes for the rankings by the ten methods. F(t) denotes the number of infected and recovered nodes at time t, and t varies from 1 to 20. (a) Karate. (b) Ca-CSphd. (c) Friendship. (d) Random. (e) Email. (f) Powergrid. (g) BA. (h) Caida. (i) Douban.

Normally, a more important node infects more nodes, thus, an effective algorithm should generate a curve that decreases as node influence decreases. As can be seen from Fig. 3, LNC performs best on Karate, Email, Friendship, Powergrid, Douban, and Caida networks. For the Email network, Gravity has the best performance, but LNC still has an advantage over most methods, and all curves are smooth with only a slight fluctuation. For Random and BA networks, all methods have similar effects, but LNC is still outstanding. As shown in Fig. 2, the Kendall τ of the Random network is still the highest, and the infectivity of the top 10 nodes is relatively strong compared with other methods. Table 5 further presents the ranking results on the Karate network for the ten methods, where each node is treated as a seed node for infecting its neighbor nodes. It can be seen that LNC performs best compared to other methods.

TABLE 5. Ranking by the SIR model according to node influence by the ten methods. Here, PL, PR and Gr denotes ProfitLeader, PageRank and Gravity respectively.

t	BC	CC	DC	EC	HITS	PR	PL	Gr	WFCA	LNC
1	3.44	3.50	3.61	3.53	3.58	3.52	3.39	3.34	3.50	3.56
2	3.71	2.90	3.31	3.41	3.39	3.39	3.54	3.58	3.47	3.41
3	3.11	3.56	3.02	2.89	3.0	3.15	3.02	2.92	2.98	3.02
4	3.03	2.30	2.81	3.09	3.03	2.95	2.93	3.00	3.00	2.87
5	2.37	2.32	2.61	2.71	2.63	2.67	2.64	2.59	2.61	2.59
6	2.39	2.30	2.16	2.37	2.29	2.25	2.35	2.26	2.20	2.38
7	2.68	3.05	2.33	2.26	2.33	2.27	2.47	2.33	2.38	2.25
8	2.23	1.79	2.40	2.15	2.35	2.09	2.27	2.29	2.31	2.25
9	1.86	2.72	2.31	2.32	2.38	2.35	2.27	1.99	2.39	2.29
10	1.70	2.29	2.22	2.07	2.05	2.22	2.00	2.14	2.21	2.12
11	1.80	2.03	1.64	2.13	1.97	1.75	1.69	2.32	1.85	2.01
12	1.95	1.85	1.81	2.24	1.88	1.76	1.81	1.90	2.04	1.96
13	2.12	1.87	2.09	1.96	2.08	1.84	1.86	1.96	2.07	1.90
14	2.04	1.97	2.07	1.93	1.64	2.04	2.00	2.07	1.74	1.95
15	2.18	1.59	1.86	1.92	1.72	2.14	1.98	1.75	1.78	1.76

Furthermore, we compare the influence of the top 10 nodes that are selected by LNC and the other methods. All top 10 nodes are used as seed nodes and the time step t ranges from 1 to 15. Table 6 shows that LNC has the highest propagation capability for all nodes and all methods. Clearly, the infection node $F(t)$ increases as t increases, and the propagation in most networks reaches a steady state at $t = 10$. This is because the top 10 influential nodes infect other nodes with high efficiency, and stability is attained after a certain threshold is reached [52]. In addition, Fig. 4 shows the influence of the top 10 nodes in the nine networks, and it can be seen that the LNC method has high spreading efficiency. LNC has the best performance on most networks, that is, Ca-CSphd, Friendship, Douban, and Caida.

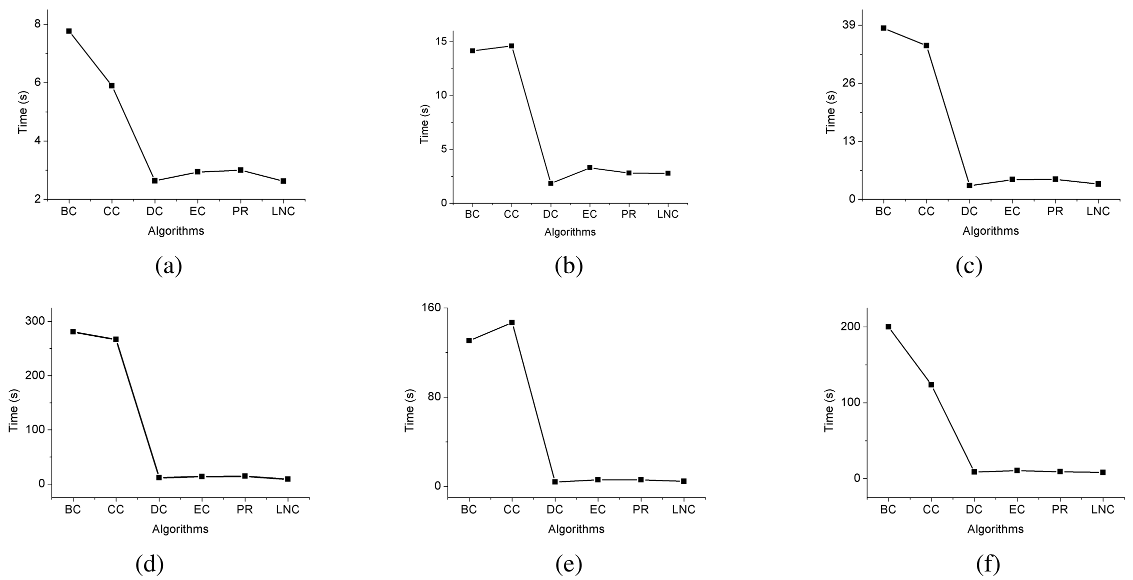

FIGURE 5. LNC method compared with other methods in terms of running time. (a) Ca-CSphd. (b) Email. (c) Friendship. (d) Web-Spam. (e) Random. (f) Powergrid.

TABLE 6. Influences $F(t)$ for top 10 nodes by LNC and the other methods in Friendship network. Here, PR, PL and Gr denotes PageRank, ProfitLeader and Gravity, respectively.

t	BC	CC	DC	EC	HITS	PR	PL	Gr	WFCA	LNC
1	24.08	23.69	25.87	24.88	25.74	25.33	24.76	26.49	28.34	25.76
2	28.34	27.60	29.69	28.64	29.64	28.87	28.84	30.18	30.37	30.30
3	29.83	28.83	30.92	30.33	31.18	30.35	30.21	30.84	30.79	31.45
4	29.61	29.05	32.19	30.47	31.89	30.23	31.22	30.72	30.85	31.68
5	30.37	29.98	32.04	30.61	31.78	31.74	31.24	30.95	30.91	32.72
6	29.76	28.86	32.21	30.80	32.14	31.46	30.12	30.96	30.87	32.16
7	29.86	29.34	31.79	31.51	31.72	31.46	31.07	30.83	31.04	32.34
8	30.72	29.84	31.94	30.82	32.33	31.85	30.86	30.95	30.93	32.45
9	30.17	29.36	32.60	30.55	32.11	31.07	31.22	30.92	30.99	32.28
10	30.12	30.68	32.19	31.14	32.12	31.55	31.09	30.89	31.03	32.40
11	30.34	29.60	32.13	30.69	32.28	31.65	30.85	30.89	30.91	32.11
12	30.49	29.51	32.05	30.78	31.91	31.78	30.46	30.86	30.87	32.09
13	29.83	29.87	32.24	30.76	32.20	31.41	30.88	30.87	31.00	32.26
14	30.19	29.94	32.13	30.99	32.33	31.62	30.98	30.87	30.84	32.39
15	29.98	29.45	31.27	30.87	32.32	31.68	31.33	30.85	30.93	32.06

Moreover, Fig. 5 shows a comparison of the time efficiency of LNC with that of the other methods. We chose to use six test data with more than 1000 nodes in the experiment, and the time required to sort the nodes is used as a measure of the running time of these methods. We compare the LNC method with several representative methods, namely, DC, CC, BC, EC, and PageRank. As can be seen from Fig. 5, LNC has the least runtime in Web-Spam and Powergrid networks, and is above centrality only in Email, Ca-CSphd, Random and Friendship. Thus, we verified the high efficiency of LNC.

V. CONCLUSION

In this study, we considered the problem of detecting influential nodes based on the local neighbor contribution. The proposed method considers local information regarding a given network. This is a new measurement method for identifying influential nodes, which has low time and computational complexity. Accordingly, this method can be applied to large and complex networks. We considered both network complexity and the influence of the nearest and the next nearest neighbor nodes. To evaluate the performance of the proposed method, we applied it to nine networks and used the SIR model to simulate the spreading process by employing Kendall τ to measure the correlation between the ranking lists generated by the simulation and the other identification methods. Experimental results regarding monotonicity, correctness, and efficiency demonstrated that the proposed method exhibits excellent performance on both artificial and real-world networks. The new method can discriminate the node influence more accurately and provides a more reasonable ranking list than other measures.

REFERENCES

[1] Z. Sun, B. Wang, J. Sheng, Y. Hu, Y. Wang, and J. Shao, "Identifying influential nodes in complex networks based on weighted formal concept analysis," *IEEE Access*, vol. 5, pp. 3777–3789, 2017.

[2] J. Shao, Z. Han, Q. Yang, and T. Zhou, "Community detection based on distance dynamics," in *Proc. ACM 21th SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2015, pp. 1075–1084.

[3] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proc. Nat. Acad. Sci. USA*, vol. 99, no. 12, pp. 7821–7826, Apr. 2002.

[4] D. Chen, L. Lü, M.-S. Shang, Y.-C. Zhang, and T. Zhou, "Identifying influential nodes in complex networks," *Phys. A, Stat. Mech. Appl.*, vol. 391, no. 4, pp. 1777–1787, 2012.

[5] A. Zareie and A. Sheikhahmadi, "A hierarchical approach for influential node ranking in complex social networks," *Expert Syst. Appl.*, vol. 93, pp. 200–211, Mar. 2018.

[6] Y. Liu, M. Tang, T. Zhou, and Y. Do, "Identify influential spreaders in complex networks, the role of neighborhood," *Phys. A, Stat. Mech. Appl.*, vol. 452, pp. 289–298, Jun. 2016.

[7] J. Zhong, F. Zhang, and Z. Li, "Identification of vital nodes in complex network via belief propagation and node reinsertion," *IEEE Access*, vol. 6, pp. 29200–29210, 2018.

[8] T. Bian and Y. Deng, "Identifying influential nodes in complex networks: A node information dimension approach," *Chaos*, vol. 28, no. 4, pp. 043109-1–043109-103, 2018.

[9] C. Li, L. Wang, S. Sun, and C. Xia, "Identification of influential spreaders based on classified neighbors in real-world complex networks," *Appl. Math. Comput.*, vol. 320, pp. 512–523, Mar. 2018.
[10] A. Farooq, G. J. Joyia, M. Uzair, and U. Akram, "Detection of influential nodes using social networks analysis based on network metrics," in *Proc. Int. Conf. Comput., Math. Eng. Technol. (iCoMET)*, Mar. 2018, pp. 1–6.
[11] S. Basu and U. Maulik, "Mining important nodes in complex networks using nonlinear PCA," in *Proc. IEEE Calcutta Conf. (CALCON)*, Dec. 2017, pp. 469–473.
[12] Y. Wang, S. Wang, and Y. Deng, "A modified efficiency centrality to identify influential nodes in weighted networks," *Pramana*, vol. 92, no. 4, p. 68, 2019.
[13] K. Yada, H. Motoda, T. Washio, and A. Miyawaki, "Consumer behavior analysis by graph mining technique," *New Math. Natural Comput.*, vol. 2, no. 1, pp. 59–68, 2006.
[14] G. Klopman, "Artificial intelligence approach to structure-activity studies. computer automated structure evaluation of biological activity of organic molecules," *J. Amer. Chem. Soc.*, vol. 106, no. 24, pp. 7315–7321, 1984.
[15] S. Parthasarathy, S. Tatikonda, and D. Ucar, "A survey of graph mining techniques for biological datasets," in *Managing and Mining Graph Data*. Springer, 2010, pp. 547–580.
[16] H. Rahmani, H. Blockeel, and A. Bender, "Using a human drug network for generating novel hypotheses about drugs," *Intell. Data Anal.*, vol. 20, no. 1, pp. 183–197, 2016.
[17] C. Phua, V. Lee, K. Smith, and R. Gayler, "A comprehensive survey of data mining-based fraud detection research," 2010, *arXiv:1009.6119*. [Online]. Available: https://arxiv.org/abs/1009.6119
[18] L. Xiaojun, H. Song, and X. Weikun, "The analysis of logistics influence of the important node cities of beijing-tianjin-hebei," *Int. J. Bus. Econ. Res.*, vol. 6, no. 5, p. 88, 2017.
[19] X. Wen, C. Tu, and M. Wu, "Fast ranking nodes importance in complex networks based on LS-SVM method," *Phys. A, Stat. Mech. Appl.*, vol. 506, pp. 11–23, Jun. 2016.
[20] B. Wei and Y. Deng, "A cluster-growing dimension of complex networks: From the view of node closeness centrality," *Phys. A, Stat. Mech. Appl.*, vol. 522, pp. 80–87, May 2019.
[21] P. Bonacich, "Factoring and weighting approaches to status scores and clique identification," *J. Math. Sociol.*, vol. 2, no. 1, pp. 113–120, 1972.
[22] G. Sabidussi, "The centrality index of a graph," *Psychometrika*, vol. 31, no. 4, pp. 581–603, Dec. 1966.
[23] L. C. Freeman, "Centrality in social networks conceptual clarification," *Soc. Netw.*, vol. 1, no. 3, pp. 215–239, 1979.
[24] S. Brin and L. Page, "Reprint of: The anatomy of a large-scale hypertextual Web search engine," *Comput. Netw.*, vol. 56, no. 18, pp. 3825–3833, Dec. 2012.
[25] L. Lü, Y.-C. Zhang, C. H. Yeung, and T. Zhou, "Leaders in social networks, the delicious case," *PLoS ONE*, vol. 6, no. 6, Jun. 2011, Art. no. e21202.
[26] L. Lü, D. Chen, X.-L. Ren, Q.-M. Zhang, Y.-C. Zhang, and T. Zhou, "Vital nodes identification in complex networks," *Phys. Rep.*, vol. 650, pp. 1–63, Sep. 2016.
[27] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *J. ACM* , vol. 46, no. 5, pp. 604–632, Sep. 1999.
[28] C. Salavati, A. Abdollahpouri, and Z. Manbari, "Ranking nodes in complex networks based on local structure and improving closeness centrality," *Neurocomputing*, vol. 336, pp. 36–45, Apr. 2019.
[29] L. Fei, Q. Zhang, and Y. Deng, "Identifying influential nodes in complex networks based on the inverse-square law," *Phys. A, Stat. Mech. Appl.*, vol. 512, pp. 1044–1059, Dec. 2018.
[30] L. Yin and Y. Deng, "Toward uncertainty of weighted networks: An entropy-based model," *Phys. A, Stat. Mech. Appl.*, vol. 508, pp. 176–186, Oct. 2018.
[31] Z. Yu, J. Shao, Q. Yang, and Z. Sun, "Profitleader: Identifying leaders in networks with profit capacity," *World Wide Web*, vol. 22, pp. 533–553, Mar. 2019.
[32] M. Kitsak, L. K. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H. E. Stanley, and H. A. Makse, "Identification of influential spreaders in complex networks," *Nature Phys.*, vol. 6, pp. 888–893, Aug. 2010.
[33] E. Estrada and J. A. Rodríguez-Velázquez, "Subgraph centrality in complex networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 71, no. 5, May 2005, Art. no. 056103.
[34] T. Bian and Y. Deng, "A new evidential methodology of identifying influential nodes in complex networks," *Chaos, Solitons Fractals*, vol. 103, pp. 101–110, Oct. 2017.
[35] S. Wang, Y. Du, and Y. Deng, "A new measure of identifying influential nodes: Efficiency centrality," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 47, pp. 151–163, Jun. 2017.
[36] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.
[37] M. Li, Q. Zhang, and Y. Deng, "Evidential identification of influential nodes in network of networks," *Chaos, Solitons Fractals*, vol. 117, pp. 283–296, Dec. 2018.
[38] Z. Wang, X. Pei, Y. Wang, and Y. Yao, "Ranking the key nodes with temporal degree deviation centrality on complex networks," in *Proc. IEEE 29th Chin. Control Decis. Conf. (CCDC)*, May 2017, pp. 1484–1489.
[39] S. Xu and P. Wang, "Identifying important nodes by adaptive LeaderRank," *Phys. A, Stat. Mech. Appl.*, vol. 469, pp. 654–664, Mar. 2017.
[40] J. Wu and Y. Yue-Jin, "Finding the most vital node by node contraction in communication networks," in *Proc. Int. Conf. Commun., Circuits Syst.*, May 2005, p. 1286.
[41] J. Hu, Y. Du, H. Mo, D. Wei, and Y. Deng, "A modified weighted TOPSIS to identify influential nodes in complex networks," *Phys. A, Stat. Mech. Appl.*, vol. 444, pp. 73–85, Feb. 2016.
[42] W. W. Zachary, "An information flow model for conflict and fission in small groups," *J. Anthropol. Res.*, vol. 33, no. 4, pp. 452–473, 1977.
[43] (Sep. 2016). *U. Rovira I Virgili Network Dataset—KONECT*. [Online]. Available: http://konect.uni-koblenz.de/networks/arenas-email
[44] J. Kunegis, "KONECT: The Koblenz network collection," in *Proc. Int. Conf. World Wide Web Companion*, 2013, pp. 1343–1350. [Online]. Available: http://userpages.uni-koblenz.de/~kunegis/paper/kunegis-koblenz-network-collection.pdf
[45] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, pp. 440–442, Jun. 1998.
[46] R. A. Rossi and N. K. Ahmed, "The network data repository with interactive graph analytics and visualization," in *Proc. 29th AAAI Conf. Artif. Intell.*, 2015, pp. 4292–4293. [Online]. Available: http://networkrepository.com
[47] J. Leskovec, J. Kleinberg, and C. Faloutsos, "Graph evolution: Densification and shrinking diameters," *ACM Trans. Knowl. Discovery Data*, vol. 1, no. 1, pp. 1–40, 2007.
[48] (Sep. 2016). *Caida Network Dataset—KONECT*. [Online]. Available: http://konect.uni-koblenz.de/networks/as-caida20071105
[49] R. Zafarani and H. Liu. (2009). *Social Computing Data Repository at ASU*. [Online]. Available: http://socialcomputing.asu.edu
[50] R. M. Anderson, R. M. May, and B. Anderson, *Infectious Diseases of Humans: Dynamics and Control*. London, U.K.: Oxford Univ. Press, 1992.
[51] M. G. Kendall, "A new measure of rank correlation," *Biometrika*, vol. 30, no. 1/2, pp. 81–93, Jun. 1938.
[52] S. Peng, Y. Zhou, L. Cao, S. Yu, J. Niu, and W. Jia, "Influence analysis in social networks: A survey," *J. Netw. Comput. Appl.*, vol. 106, pp. 17–32, Mar. 2018.

JINYING DAI received the B.S. degree in software engineering from Henan University, in 2016. She is currently pursuing the M.S. degree with the School of Computer Science and Engineering, Central South University, Changsha, China. Her research interests include software engineering and complex networks.

BIN WANG received the M.Sc. degree in mining engineering and the Ph.D. degree in computer science and technology from Central South University, China, in 1999 and 2003, respectively, where he is currently a Professor with the School of Computer Science and Engineering. His research interests include transparent computing and software engineering.

JINFANG SHENG received the M.Sc. degree in computer science and technology and the Ph.D. degree in control theory and control engineering from Central South University, China, in 1996 and 2007, respectively, where she is currently an Associate Professor with the School of Computer Science and Engineering. Her research interests include transparent computing and big data processing.

ZEJUN SUN received the B.Sc. degree from Henan Polytechnic University, in 2003, and the M.Sc. degree from Xidian University, China, in 2008, both in computer science. He is currently pursuing the Ph.D. degree with Central South University, China. His research interests include data mining, complex network structure mining, and machine learning.

FAIZA RIAZ KHAWAJA received the B.S. degree in software engineering from Isra University, Hyderabad, Pakistan, in 2016. She is currently pursuing the M.S. degree in computer science and technology with Central South University, Changsha, China. She was a Teaching Assistant with Isra University for one year. Her research interests include software engineering and complex networks.

AMAN ULLAH received the B.S. degree in computer science from Gomal University, Dera Ismail Khan, Pakistan, in 2012, and the M.S. degree in software engineering from Abasyn University, Peshawar, Pakistan, in 2017. He is currently pursuing the Ph.D. degree with Central South University, Hunan, China. His research interests include software engineering, software cost estimation, artificial intelligent, and complex networks.

DAWIT AKLILU DEJENE received the B.S. degree (Hons.) in computer science from Unity University, Addis Ababa, Ethiopia, in 2016. He is currently pursuing the M.S. degree with the School of Computer Science and Engineering, Central South University, Changsha, Hunan. His research interests include cloud computing and network security.

GUIHUA DUAN received the Ph.D. degree in computer science and technology from Central South University, in 2010, where she is currently an Associate Professor with the School of Computer Science and Engineering. Her research interests include big data processing and network security.