IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 5, 2017, accepted February 28, 2017, date of publication March 7, 2017, date of current version April 24, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2679038

Identifying Influential Nodes in Complex Networks Based on Weighted Formal Concept Analysis

ZEJUN SUN[1,2], BIN WANG[1], JINFANG SHENG[1], YIXIANG HU[1], YIHAN WANG[1], AND JUNMING SHAO[3]

[1]School of Information Science and Engineering, Central South University, Changsha 10083, China
[2]Department of Network Center, Pingdingshan University, Pingdingshan 467000, China
[3]Big Data Research Center, University of Electronic Science and Technology of China, Chengdu 611731, China

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

This work was supported in part by the National Natural Science Foundation of China under Grant 61403062, in part by the Science-Technology Foundation for Young Scientist of Sichuan Province under Grant 2016JQ0007, in part by the Natural Science Foundation of Fujian Province of China under Grant 2015J01271, and in part by the Education Hall of Young Teachers' Scientific Research Project of Fujian Province of China under Grant JAT160469.

ABSTRACT The identification of influential nodes is essential to research regarding network attacks, information dissemination, and epidemic spreading. Thus, techniques for identifying influential nodes in complex networks have been the subject of increasing attention. During recent decades, many methods have been proposed from various viewpoints, each with its own advantages and disadvantages. In this paper, an efficient algorithm is proposed for identifying influential nodes, using weighted formal concept analysis (WFCA), which is a typical computational intelligence technique. We call this a WFCA-based influential nodes identification algorithm. The basic idea is to quantify the importance of nodes via WFCA. Specifically, 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. WFCA not only works on undirected or directed networks, but is also applicable to attributed networks. To evaluate the performance of WFCA, we employ the SIR model to examine the spreading efficiency of each node, and compare the WFCA algorithm with PageRank, HITS, K-shell, H-index, eigenvector centrality, closeness centrality, and betweenness centrality on several real-world networks. Extensive experiments demonstrate that the WFCA algorithm ranks nodes effectively, and outperforms several state-of-the-art algorithms.

INDEX TERMS Influential nodes, weighted formal concept analysis, complex networks, SIR model.

I. INTRODUCTION

During recent decades, complex network mining has been the subject of significant attention [1]–[3]. This can help in understanding complex network functions, as well as discovering the regularity of the dynamic evolution of complex networks and predicting their behavior [4]–[6]. In a complex network, each node may have a different status or role and the roles of different nodes in the structure and function may be largely different. Some nodes affect the structure and function of the network to a greater extent than others, and these are called influential nodes [7]. The study of influential nodes has important practical value. In the example of epidemic spreading, if we know the influential nodes in a given network, this may help to predict the spread of the disease, and to control the disease before an epidemic outbreak occurs [8], [9]. In the criminal networks, the importance of ranking in favor of discriminating the ringleaders, backbone members and followers, quickly locate the leader of criminal gangs [10]. Identifying influential nodes in networks also can do much good to many applications such as effective vaccinations strategies [11], saving human lives [12], and the resolution of social dilemma [13], all these relying on proper identification of influential nodes. In fact, identifying influential nodes has played an important role in the analysis of social networks, biological networks, information networks, and transportation systems [14]–[17]. Thus, techniques for identifying

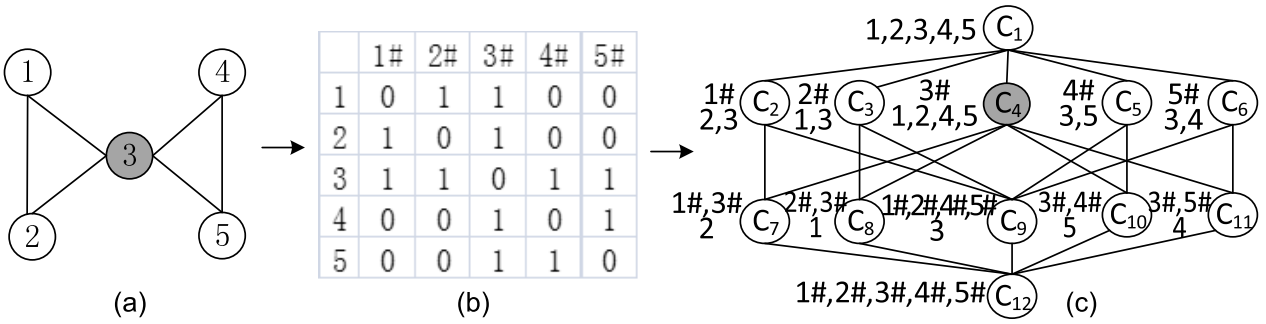


FIGURE 1. Toy example of WFCA. (a) A toy network G. (b) The formal context of the network G. (c) The concept lattices of the network G. WFCA clusters objects by attributes. For example, the concept C_4 clusters concepts C_7, C_8, C_{10}, and C_{11}, which have the common attribute {3#}. The objects 1, 2, 4, 5 are gathered to the concept C_4. Therefore, the more objects that are aggregated, the more important the corresponding attribute becomes.

influential nodes in complex networks have been the subject of increasing attention. So far, many methods for identifying influential nodes have been proposed, such as degree centrality [18], closeness centrality [19], betweennesscentrality [20], PageRank [21], Leader Rank [22], H-index [23], and HITS [24]. Although many algorithms have been proposed, these methods all have their own advantages and limitations. For example, degree centrality has a comparatively low time complexity, but has a low accuracy. The PageRank algorithm is stable in scale-free networks, but it is sensitive to disturbances of random networks [23]. Leader Rank performs well in directed complex networks, but cannot be applied in undirected complex networks.

In this paper, we propose a new method, called WFCA, to identify influential nodes in complex networks. We consider the problem of detecting influential nodes from the new viewpoint of weighted formal concept analysis. This takes into account the relationships of nodes in terms of the overall structure of the network, and converts binary relationships into a hierarchy structure. We will describe WFCA in Section 3, but first let us elaborate on the basic idea.

A. BASIC IDEAS

Formal concept analysis (FCA) [25] provides a powerful approach to data mining and knowledge representation. The concept of FCA consists of objects and attributes. FCA uses binary object-attribute relations to construct a knowledge hierarchy that reflects the intrinsic relationships between objects and attributes. Here, we consider a given complex network as a formal context. Nodes of the complex network represent the objects and attributes of the formal context. An edge between nodes denotes a binary relation, where an attribute belongs to an object. When building upon the concepts of FCA, the calculation of important nodes is involved in three major steps. First, we construct the adjacency matrix of a network as the formal context. Subsequently, we compute the concepts of the formal context. Finally, we calculate the weight of each node based on the concepts.

To further illustrate the basic idea, let us consider a toy network as an example. Here, Fig. 1(a) illustrates a network G that contains five nodes and six edges. Fig. 1(b) presents the formal context of the network G, and Fig. 1(c) shows the concept lattices of the formal context. The method of calculating the concept lattice will be described in Section 3.2. In Fig. 1(c), there are 12 concepts, denoted by C_1 to C_{12}. The numbers (1, 2, 3, 4, 5) represent objects, and (1#, 2#, 3#, 4#, 5#) denote attributes. Both of these represent the nodes (1, 2, 3, 4, 5). Formal concept analysis aggregates these objects by attributes to form a hierarchy, where C_{12} is located in the bottom, $C_7 - C_{11}$ are located in the first layer, $C_2 - C_6$ are located in the second layer, and C_1 is located in the top layer. As the level increases, more objects are added, while the attributes gradually reduce. The hierarchy structure not only reflects the intrinsic links between nodes, but also characterizes the generalization-instantiation relationships between concepts. For example, because the concepts C_7, C_8, C_{10}, and C_{11} have the common attribute {3#}, objects with this attribute will be gathered to form the concept C_4. In Fig. 1(c), we can observe that the more objects are aggregated, the more important the corresponding attribute becomes. The weights of the attributes in each concept are obtained intuitively, by using the number of objects divided by the number of attributes.

B. CONTRIBUTIONS

WFCA has several attractive benefits for identifying influential nodes in complex networks, most importantly:

- **A new viewpoint**: We consider the problem of detecting influential nodes from a new viewpoint: **weighted formal concept analysis**. This takes into account global information regarding the network, and converts binary relationships between nodes in a network into a hierarchy. This hierarchy not only reflects the intrinsic links between nodes, but also characterizes the generalization-instantiation relationships between concepts of nodes. This allows the clustering of object nodes based on identical attributes nodes. Therefore, the more objects that are aggregated, the more important the corresponding attribute becomes (see Fig. 1(c)).
- **High performance**: Compared with several representatives of influential node detection algorithms, the WFCA

method is shown to be more effective (cf. Fig. 5-Fig. 7, Table5, Table6).
- **Flexibility**: Our method can not only be used in undirected or directed networks, but is also applicable to attributed networks.

The remainder of this paper is organized as follows. In the next section, we provide a brief overview of related work. Section 3 describes the main idea of WFCA, and then presents the algorithm in detail. Section 4 compares WFCA with several representative methods on eight real-world networks. Finally, our conclusions are presented in Section 5.

II. RELATED WORK

In recent years, many approaches have been proposed for identifying influential nodes (e.g., degree centrality [18], K-shell [26], closeness centrality [19], betweenness centrality [20], eigenvector centrality [27], PageRank [21], Leader Rank [22], H-index [23], and HITS [24]). Here, we only provide a brief survey regarding the identification of influential nodes.

A. STRUCTURE-BASED APPROACHES

The influence of a node is significantly affected by the network topology. In fact, the majority of approaches for identifying influential nodes only consider structural information. Existing Structure-based measures can be divided into two categories: one is based on the neighborhood of each node (such as the degree centrality, K-shell and H-index methods), while the other is based on paths between nodes (such as closeness centrality and betweenness centrality). Degree centrality characterizes nodes with larger numbers of neighbors as having a larger influence. Thus, it is the most simple index for characterizing influential nodes. However, although it is simple and easy to calculate, it suffers from poor accuracy, owing to the lack of consideration of the global network structure. K-shell determines the importance of a node according to its location in the network. Although it has a low time complexity, it is not suitable for some specific types of networks, such as rule or BA networks [28], and the sorting also results in coarse-graining. In contrast, closeness centrality and betweenness centrality both have high computational complexity, and so they are not suitable for application to large-scale networks.

B. EIGENVECTOR-BASED APPROACHES

Methods of this type not only consider the number of neighboring nodes, but also take into account their influences (such as eigenvector centrality, PageRank, Leader Rank, and HITS). Eigenvector centrality can be efficiently calculated using a power iteration approach, but it may become trapped in a zero status, because of the presence of many nodes without in-degree [23]. PageRank is a famous ranking algorithm that is used in the Google search engine, and it is a variant of the eigenvector centrality method. It supposes that the importance of a web page is determined by both the quantity and the quality of the pages linked to it. PageRank performs well in scale-free networks, and has been widely employed in many fields. However, it is sensitive to disturbances of random networks, and it exhibits topic drifts in special network structures [23]. The HITS algorithm considers each node in the network as performing two roles: authority and hub. Similarly, HITS also exhibits a topic drift phenomenon [29]. Leader Rank performs well in directed complex networks, but cannot be applied in undirected complex networks.

In general, each algorithm has its own advantages and disadvantages. Effectively and efficiently identifying influential nodes remains a non-trivial task. Here, we provide an effective method for identifying influential nodes, which can not only be applied to directed or undirected networks, but is also suitable for attributed networks.

TABLE 1. Example of a formal context.

	a	b	c	d	e	f	g
1		×	×			×	
2	×	×	×	×		×	
3		×	×		×	×	
4		×	×			×	×
5		×				×	×
6		×			×	×	

III. IDENTIFICATION OF INFLUENTIAL NODES BASED ON WFCA

A. PRELIMINARIES

In this section, we first introduce some basic concepts and definitions concerning FCA. FCA is a powerful data analysis technique, and was first proposed by Rudolf Wille in 1982 [25]. In the past several decades, FCA has been widely applied in software engineering [30], text processing [31], data mining [32], ontology engineering [33], and other fields [34], [35]. FCA considers entities that consist of objects and attributes, where objects have attributes, and attributes belong to objects. The following are some definitions relating to FCA.

Definition 1: A formal context is a triplet of sets $\mathbb{K} := (O, A, I)$, where O is a set of objects, A is a set of attributes, and $I \subseteq O \times A$ is a binary relation between O and A. The object o is an element of the object set O, the attribute a is an element of the attribute set A, and oIa or $(o, a) \in I$ indicates that the object o has the attribute a.

Table 1 presents a formal context. It can be represented as a two-dimensional table (or matrix), where the rows represent objects and the columns are attributes. The cross cell of a row and column in the table represents the incidence relation I, and all relations between objects and attributes could be written in a table. In Table 1, the symbol × indicates that an object has the corresponding attribute. For example, object 2 has attributes (a,b,c,d,f).

Definition 2: For a set T of objects from O, denoted as $T \subseteq O$, we define

$$T^{\Uparrow} = \{a \in A | \forall o \in T, oIa\}. \quad (1)$$

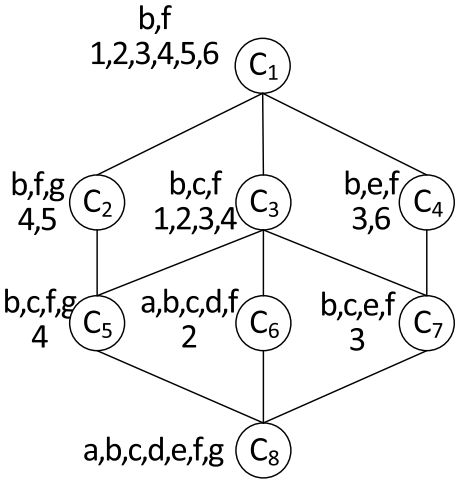


FIGURE 2. The concept lattice of the formal context in Table1.

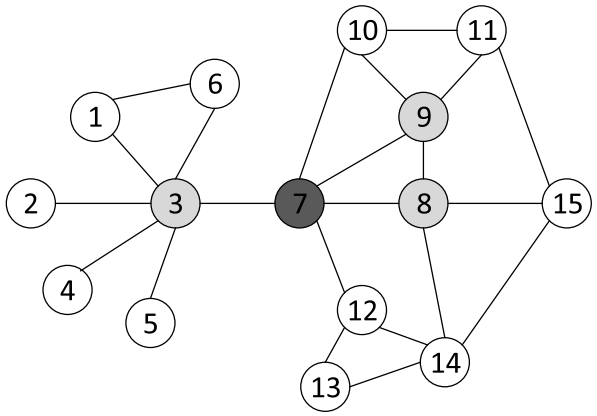


FIGURE 3. A synthetic network consisting of 15 nodes and 22 edges.

TABLE 2. A formal context of Fig. 3. The rows (1, 2, ..., 15) and columns (1#, 2#, ..., 15#) represent the nodes (1, 2, ..., 15).

	1#	2#	3#	4#	5#	6#	7#	8#	9#	10#	11#	12#	13#	14#	15#
1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0
2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
3	1	1	0	1	1	1	1	0	0	0	0	0	0	0	0
4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	1	0	0	0	0	1	1	1	0	1	0	0	0
8	0	0	0	0	0	0	1	0	1	0	0	0	0	1	1
9	0	0	0	0	0	0	1	1	0	1	1	0	0	0	0
10	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0
11	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1
12	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0
13	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0
14	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
15	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0

Here, $T^{\Uparrow}$ denotes the set of attributes shared by all of the objects in T. Similarly, we can define the set of objects shared by all of the attributes in P as $P^{\Downarrow}$. Here, for a set P of attributes from A, denoted as $P \subseteq A$, we define

$$P^{\Downarrow} = \{o \in O | \forall a \in P, oIa\}. \quad (2)$$

Definition 3: A binary group (T, P) is a formal concept of a formal context $\mathbb{K} := (O, A, I)$, with $T \subseteq O$, $P \subseteq A$, $T^{\Uparrow} = P$, and $P^{\Downarrow} = T$. We call T and P the *extent* and the *intent* of (T, P), respectively. The collection of all formal concepts of a formal context $\mathbb{K}$ is denoted as $\mathfrak{B}(O, A, I)$.

For example, $(\{1, 2, 3, 4, 5, 6\}, \{b, f\})$, $(\{1, 2, 3, 4\}, \{b, c, f\})$, and $(\{4, 5\}, \{b, f, g\})$ are formal concepts in Table 1, while $(\{1, 2, 3, 4\}, \{b, c\})$ is not a formal concept. Although the result of $\{b, c\}^{\Downarrow}$ is $\{1, 2, 3, 4\}$, the result of $\{1, 2, 3, 4\}^{\Uparrow}$ is $\{b, c, f\}$. Because $\{1, 2, 3, 4\}^{\Uparrow} \neq \{b, c\}$, it follows that $(\{1, 2, 3, 4\}, \{b, c\})$ is not a formal concept.

Proposition 4: Let $\mathbb{K} := (O, A, I)$ be a formal context, where $T, T_1, T_2 \subseteq O$ are sets of objects and $P, P_1, P_2 \subseteq A$ are sets of attributes. Then, the following properties hold:

1) $T_1 \subseteq T_2 \Rightarrow T_2^{\Uparrow} \subseteq T_1^{\Uparrow}$
2) $P_1 \subseteq P_2 \Rightarrow P_2^{\Downarrow} \subseteq P_1^{\Downarrow}$
3) $T \subseteq T^{\Uparrow\Downarrow}$
4) $P \subseteq P^{\Downarrow\Uparrow}$
5) $T^{\Uparrow} = T^{\Uparrow\Downarrow\Uparrow}$
6) $P^{\Downarrow} = P^{\Downarrow\Uparrow\Downarrow}$
7) $T \subseteq P^{\Downarrow} \Leftrightarrow P \subseteq T^{\Uparrow} \Leftrightarrow T \times P \subseteq I$

Definition 5: If (T_1, P_1) and (T_2, P_2) are two concepts of a formal context $\mathbb{K}$, where $T_1 \subseteq T_2$ (which is equivalent to $P_2 \subseteq P_1$ by property 1), then (T_1, P_1) is called a *subconcept* of (T_2, P_2), and (T_2, P_2) is called a *superconcept* of (T_1, P_1). This can be denoted by $(T_1, P_1) \leq (T_2, P_2)$.

The relation $\leq$ is called a hierarchical order of concept, and represents a partial order. All of the concepts can be combined using hierarchical ordering, with the result being called a *concept lattice*, which denoted by $\underline{\mathfrak{B}}(O, A, I)$. A concept lattice can be represented by a Hasse diagram. For example, Fig. 2 presents the Hasse diagram of the concept lattice of Table 1. All of the concepts are listed as follows:

1) $(\{1, 2, 3, 4, 5, 6\}, \{b, f\})$
2) $(\{4, 5\}, \{b, f, g\})$
3) $(\{1, 2, 3, 4\}, \{b, c, f\})$
4) $(\{3, 6\}, \{b, e, f\})$
5) $(\{4\}, \{b, c, f, g\})$
6) $(\{2\}, \{a, b, c, d, f\})$
7) $(\{3\}, \{b, c, e, f\})$
8) $(\{\}, \{a, b, c, d, e, f, g\})$

Definition 6: Let (O, A, W, I) be a multi-valued formal context. This is composed of sets O, A, W and a ternary relation I (with $I \subseteq O \times A \times W$. Then,

$(o, a, w) \in I$ and $(o, a, v) \in I \Rightarrow w = v$,

where $(o, a, w) \in I$ can be understood to mean that the attribute a of the object g has the value w. If w has n elements, then (O, A, W, I) is called an n-valued context.

Definition 7: Let $G = (V, E)$ be a graph of a complex network, where V is the set of nodes, E is the set of edges, $e = \{u, v\}$ denotes an edge from between the nodes u and v, and $e \subseteq E$.

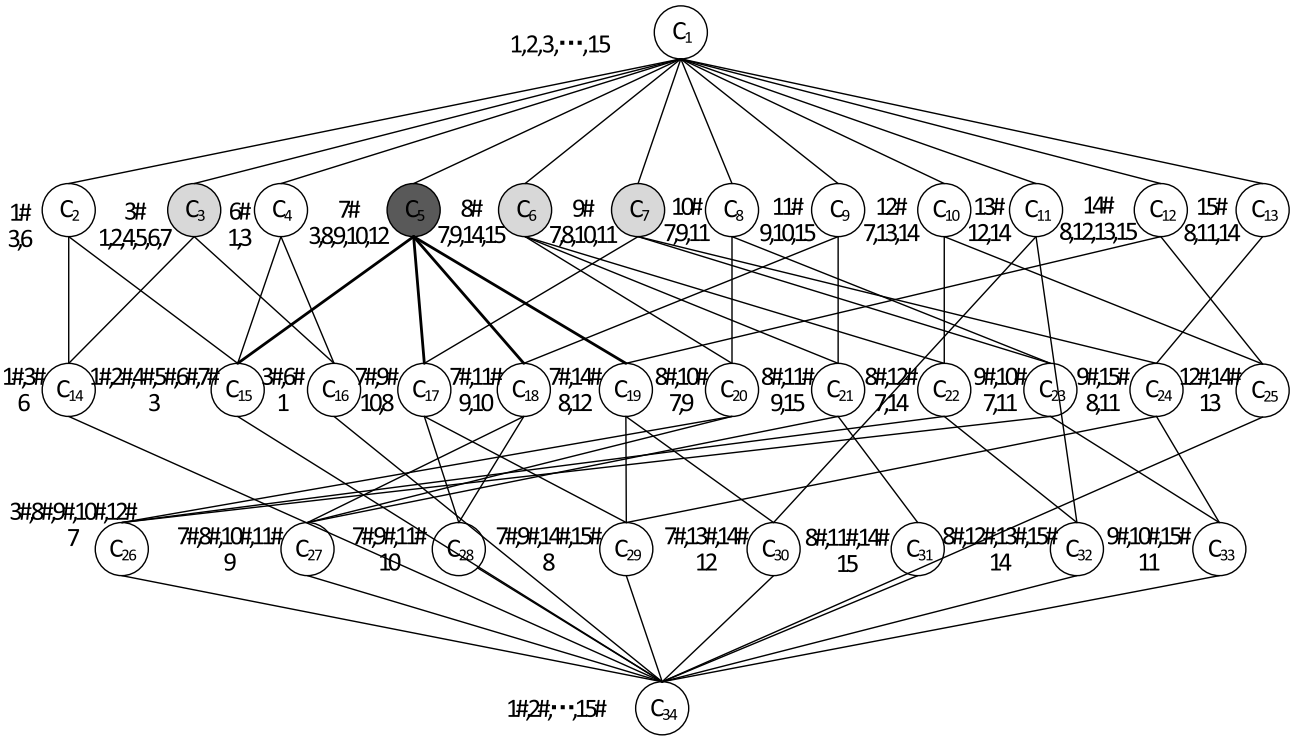


FIGURE 4. Concept lattice of Table 2.

TABLE 3. The list is ranked by WFCA and several other algorithms, where the last two columns are ranked by the SIR model.

PR	HITS	EC	CC	BC	K-shell	H-index	WFCA	WFCA vlaue	SIR	SIR value
3	7	7	7	3	8	7	7	9.333333	7	2.671
7	9	9	3	7	9	8	9	8.116667	9	2.39
14	8	8	8	8	7	9	8	8.033333	8	2.388
9	10	10	9	12	10	10	3	7.2	3	2.377
8	14	14	12	14	11	11	14	6.416667	14	2.189
12	11	3	10	9	14	14	11	5.916667	10	2.079
11	15	11	14	10	15	15	10	5.783333	15	2.025
15	12	15	15	15	3	3	12	4.95	12	2.019
10	3	12	11	11	12	12	15	4.833333	11	1.932
1	13	13	1	1	13	13	1	2.666667	6	1.676
6	1	1	6	6	1	6	6	2.666667	1	1.666
13	6	6	2	13	6	1	13	2.583333	13	1.664
2	2	2	4	2	4	4	2	0.166667	5	1.46
4	4	4	5	4	5	5	4	0.166667	4	1.418
5	5	5	13	5	2	2	5	0.166667	2	1.415

B. WEIGHTED FORMAL CONCEPT ANALYSIS

We consider the issue of detecting influential nodes from the perspective of WFCA. From the graph G, we can obtain the adjacency matrix. Let $M = (m_{u,v})$ denote the adjacency matrix, and let $|V|$ be the number of vertices of G. For example, let e be an edge linking the vertex u to the vertex v. Then, if G is a directed graph, $m_{u,v} = 1$ and $m_{v,u} = 0$. Otherwise, $m_{u,v} = 1$ and $m_{v,u} = 1$. Furthermore, if there is no edge between u and v, then $m_{u,v} = 0$ and $m_{v,u} = 0$. We consider the adjacency matrix as a formal context $\mathbb{K}$, where each row is an object and each column is an attribute.

Fig. 3 illustrates a synthetic network containing 15 nodes and 22 edges. Now, based on the concepts and definitions of WFCA, we can rank the nodes by importance. The main idea of a node importance ranking using WFCA is to first cluster nodes in a hierarchical tree, and then to compute all concepts of the network. Finally, we calculate the weight of each node, and rank the nodes by weight. Specifically,

1) Construct the adjacency matrix of the graph G (directed or undirected).
2) Convert the adjacency matrix into a formal context.
3) Compute all of the concepts of the formal context.
4) Calculate the weight of each node based on the concepts $W_i = \sum_{k=1}^{n} \frac{O_{ik}}{A_{ik}}$.
5) Rank the weights of the nodes.

Algorithm 1 WFCA

Input:
$G = (V, E)$;
Output:
Ranked nodes;
1: //Initialize the formal context K and weight W, where n is the number of nodes.
2: **if** G is attributed network **then**
3: $s \leftarrow PlainScaling(K)$;
4: **else**
5: $s = n$;
6: **end if**
7: $K[n][s] = 0$;
8: $W[n] = 0$;
9: //Construct a formal context K, which is a Boolean matrix.
10: **if** G is an attributed network **then**
11: **for** each node v in V **do**
12: **for** each attribute a **do**
13: **if** attribute a belong to node v **then**
14: $K[v][a] = 1$;
15: **end if**
16: **end for**
17: **end for**
18: **else**
19: **for** each edge $e = \{u, v\} \in E$ **do**
20: $K[u][v] = 1$;
21: **if** G is undirected **then**
22: $K[v][u] = 1$;
23: **end if**
24: **end for**
25: **end if**
26: // Compute all concepts C.
27: $C \leftarrow In - Close()$;
28: //Calculate the weight of each node.
29: **for** each concept C_i in C **do**
30: $O_i = countobject(C_i)$;
31: $A_i = countattribute(C_i)$;
32: **for** each attribute node j in C_i **do**
33: $W[j] += \frac{O_i}{A_i}$;
34: **end for**
35: **end for**
36: // Rank the weights of all nodes
37: **return** $Rank(W)$;

To illustrate the procedure, we first construct the adjacency matrix for the network in Fig. 3, as shown in Table 2, where the rows (1, 2, …, 15) and the columns (1#, 2#, …, 15#) represent the nodes (1, 2, …, 15). Each cell value is set as 1 or 0, where 1 indicates that an edge exists between two nodes, and 0 signals the opposite.

Next, based on the Proposition 1 and Definition 4, we can calculate all of the concepts of the formal context, as illustrated in Section 3.3. Fig. 4 presents the Hasse diagram, which contains all of the concepts of the formal context. Finally, we can calculate the weight of each node based on the concepts, where the weight of node i is denoted by W_i. Formally, let C_i be a set of concepts $(C_{i1}, C_{i2}, \ldots, C_{in})$ where the attributes contain the node i. The attribute number of C_{i1} is denoted as A_{i1}, and the object number of C_{i1} is denoted by O_{i1}. Then, the weight of node i (W_i) is computed as follows:

$$W_i = \sum_{k=1}^{n} \frac{O_{ik}}{A_{ik}}. \tag{3}$$

For example, we calculate the weight of node 1. Here, 1# can be considered as an attribute, and so 1# and the node 1 represent the same node. In the collection of all concepts of Fig. 4, there are three concepts containing the attribute 1#: ({3, 6}, {1#, }), ({6}, {1#, 3#}), and ({3}, {1#, 2#, 4#, 5#, 6#, 7#}). According to formula (3), we obtain $W_1 = \frac{2}{1} + \frac{1}{2} + \frac{1}{6} = 2.666$. Similarly, we can also calculate the weight of node 7#, as $W_7 = \frac{5}{1} + \frac{1}{6} + \frac{2}{2} + \frac{2}{2} + \frac{2}{2} + \frac{1}{4} + \frac{1}{3} + \frac{1}{4} + \frac{1}{3} = 9.333$.

After calculating the weights of all nodes, we rank these weights. Table 3 presents the results for several algorithms, including PageRank, eigenvector centrality, closeness centrality, betweeness centrality, HITS, K-shell and H-index, which are all realized using Gephi and NetworkX. In Table 3, the nodes 3, 7, 8, and 9 are found to be more important than the others by all of the algorithms. PageRank and betweeness determined node 3 to be the most important node, but eigenvector, closeness, HITS and H-index found the node 7 to be the most important. From Fig. 3, we can see that the degree of node 3 is six, and the degree of node 7 is five. Although node 3 has a larger degree than node 7, the neighborhood of node 3 looks less important than that of 7. This is because most of the neighboring nodes of node 3 are leaf nodes. However, those of node 7 are hub nodes. The WFCA method takes into account global information regarding the network, instead of local information, and identifies node 7 as the most influential node. The SIR model is employed to examine the spreading influence of each node. We set the infection rate of the SIR model with a probability of $\lambda = 0.1$, and run this 1000 times. The experiment results for the SIR model are presented in the last two columns in Table 3. The rankings of the SIR model are basically the same as for the WFCA method, which

TABLE 4. Some statistical properties of eight real-world networks: node number $|V|$, edge number $|E|$, average degree $\langle k \rangle$, maximum degree k_{max} and clustering coefficient $\langle C \rangle$.

Data Sets	$\|V\|$	$\|E\|$	$\langle k \rangle$	k_{max}	$\langle C \rangle$
Aviation	1226	2615	4.27	37	0.04
Protein	2239	6452	5.76	314	0.023
Blogs	1224	19025	31.08	147	0.21
Powergird	4941	6594	2.67	10	0.107
Euroroad	1174	1417	2.41	5	0.02
Friendships	1858	12534	5.76	85	0.167
Ca-AstroPh	18771	198050	21.34	236	0.677
DBLP	2723	20248	7.44	103	0.323

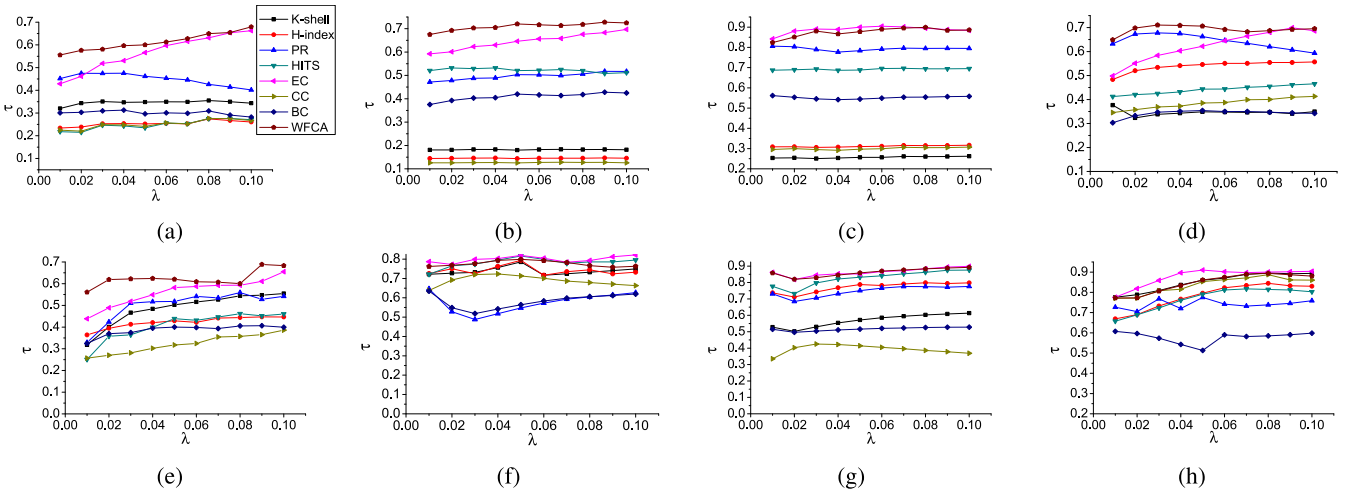


FIGURE 5. The value of Kendall's tau τ is obtained by comparing the ranking lists which generated by the eight algorithms and the ranking list generated by SIR model on eight real networks. The spreading probability λ varies from 0.01 to 0.10, and the results are obtained by averaging over 1000 independent runs. (a) Aviation. (b) Protein. (c) Blogs. (d) Powergrid. (e) Euroroad. (f) Friendships. (g) Ca-Astroph. (h) DBLP.

demonstrates that the WFCA approach identifies the influential nodes effectively. We will evaluate the performance of the WFCA algorithm further for real-world complex networks in Section 4.

C. THE WFCA ALGORITHM

In this section, we present the WFCA algorithm in detail. The WFCA algorithm consists of the following three main steps.

1) **Formal context initialization and construction**. First, we standardize the formal context K using the traditional method, plain scaling [36] (see Algorithm 1), to transform the multi-valued formal context into a one-valued formal context. For attributed networks, the formal context is denoted by $K(G) = (V, A, I)$ (cf. Definition 1), where V represents the objects of the formal context, A represents the attributes, and I represents the binary relationships between objects and attributes. For non-attributed networks, K is a modified adjacency matrix, which is denoted by $K(G) = (V, V, I)$, where the first and second Vs represent the objects and attributes, respectively. Here, we use a two-dimensional array structure to represent the formal context.
2) **Compute all concepts:** C. Based on Definition 3, we calculate all of the concepts of a formal context using a combination of rows or columns. Over recent years, many efficient algorithms have been proposed for constructing the formal concept, including AddIntent [37], FastAddIntent [38], and FCbO [39]. Here, we employ the In-Close approach [40], which is an incremental algorithm that uses a recursion to generate the combinations of attributes or objects in lexicographical sequences, while avoiding repeated generations of a concept.
3) **Node weight calculation**. Building upon the obtained concepts, we compute the weights of nodes according to the formula 3. First, we calculate the sub-weight $(\frac{O_i}{A_i})$ of each attribute of every concept. Finally, we obtain the total weight of each attribute by summing over all sub-weights.

D. RUNTIME COMPLEXITY

In WFCA algorithm, one of the important work is to compute the concepts of the formal context. However, applying FCA method to large formal context could bring many challenges, because the concepts can grow exponentially in the worst case and calculating all the concepts is an NP-complete problem [41]. The high computational complexity is actually the main weak point of FCA. However, we do not need to generate all the concept lattices in the actual calculation. So, we can use the lexicographic approach for implicitly searching, prune the recursion and avoid generating a concept repetitively. The time complexity of each step of WFCA is estimated below, where n is the number of vertices, L is the number of all concepts, and $|a|$ is the number of attribute. The time complexity of WFCA algorithm is mainly composed of three parts, and we give our main argument about asymptotic time complexity as follows.

1) **The complexity of formal context initialization and construction**. The function $PlainScaling(K)$ [36] standardize the formal context K and the time complexity is $O(n|a|)$. The construction of formal context is implemented in two loops. So, in attributed network the time complexity is $O(n|a|)$. Otherwise, the time complexity is $O(n^2)$.
2) **The complexity of computing concepts**. The concepts are calculated by In-Close [40] method and the complexity is $O(n^2L)$.
3) **The complexity of calculating node weight**. To calculate the node weight, there are two loops. The times of the first loop is the length L of the concept lattice, and that of second loop is the number of attribute whose the max

TABLE 5. The ranking list is generated by the eight algorithms. Owing to space limitations, we only show the top-10 nodes of two networks, including one directed network and one undirected network, where K-s, H-i denotes K-shell and H-index, respectively. (a) Aviation. (b) Euroroad.

(a)

PR	HITS	EC	CC	BC	K-s	H-i	WFCA
312	68	116	68	68	113	3	82
61	52	34	52	52	44	6	312
105	47	110	1149	212	604	10	116
19	44	46	113	312	852	44	46
842	113	10	116	135	68	46	110
187	102	66	110	213	52	47	34
578	89	72	44	523	523	52	10
86	109	134	47	220	89	68	134
52	34	124	51	148	212	43	52
311	187	73	132	689	213	51	68

(b)

PR	HITS	EC	CC	BC	K-s	H-i	WFCA
284	7	7	401	402	7	7	284
137	43	43	402	284	8	8	7
236	454	499	403	277	9	9	236
39	411	107	432	453	10	39	137
107	39	181	1019	452	12	43	107
7	453	454	253	403	23	50	39
204	8	39	452	401	39	107	181
768	42	180	404	404	42	144	43
164	881	411	232	837	43	253	499
181	23	8	284	836	49	401	401

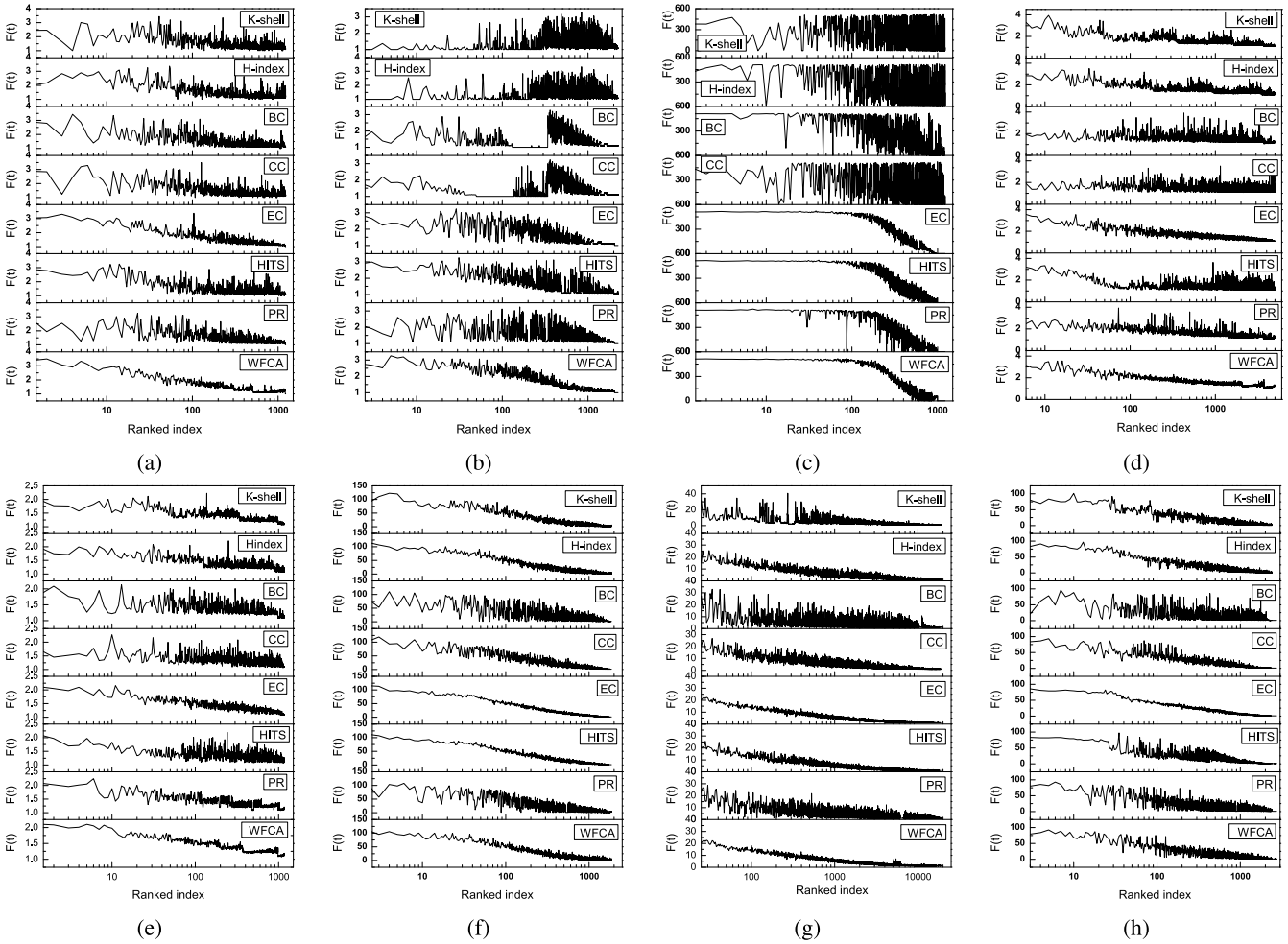


FIGURE 6. The propagation influence of the ranking lists is generated by the eight algorithms, F(t) denotes the number of infected and recovered nodes at time t (t=500), and ranked index represents the order of ranking list. (a) Aviation. (b) Protein. (c) Blogs. (d) Powergrid. (e) Euroroad. (f) Friendships. (g) Ca-Astroph. (h) DBLP.

value is n. So, the complexity of calculating node weight is $O(nL)$.

In summary, the time complexity of WFCA algorithm is the sum of the three parts, namely $O(n^2 + n^2L + nL)$. As we know, $n^2 \leq n^2L$, $nL \leq n^2L$. Therefore, the time complexity is $O(n^2L)$. We can see, the time complexity of WFCA is not very low, but its accuracy is better than the other algorithms(cf. Fig. 5-7).

TABLE 6. The influences F(t) is obtained by top-10 different nodes of ranking lists between WFCA and other algorithms in Aviation network.

t	WFCA	PR	WFCA	HITS	WFCA	EC	WFCA	CC	WFCA	BC	WFCA	H-index	WFCA	K-shell
1	27.1	24.2	28.5	21.5	27.8	21.9	28.8	20.8	26.3	22.2	30.4	22.0	30.2	22.1
2	39.8	30.7	41.0	29.2	37.7	31.9	41.1	28.1	38.0	29.8	43.5	28.7	43.3	28.9
3	46.4	35.2	50.2	35.0	44.3	39.6	47.7	34.1	44.9	35.0	51.2	34.2	50.3	33.9
4	50.3	38.9	54.5	39.5	48.5	43.9	51.7	37.5	49.6	38.1	55.3	37.1	54.8	36.5
5	52.2	40.0	57.9	42.0	51.2	46.5	54.3	39.9	51.1	40.4	59.2	38.2	56.6	38.5
6	52.8	42.3	59.3	43.2	53.6	48.3	55.7	41.2	51.6	41.8	59.1	40.7	57.8	40.0
7	52.7	43.5	61.9	43.8	54.2	49.1	56.7	42.4	51.8	42.4	60.6	41.6	57.7	40.6
8	53.5	44.0	60.8	44.2	54.9	48.3	56.2	41.7	51.7	43.3	61.1	41.5	58.9	41.5
9	54.7	44.9	61.8	44.8	56.1	49.7	56.7	42.4	52.4	43.7	60.6	41.8	59.6	41.5
10	54.6	45.1	61.7	44.5	57.5	49.6	56.2	42.5	52.8	43.7	61.5	42.3	59.3	41.7
11	54.6	44.9	62.0	44.3	57.4	49.9	56.9	42.3	53.9	44.4	61.1	41.4	60.0	42.3
12	54.3	44.9	63.2	44.2	58.3	49.1	57.5	43.0	53.8	44.1	60.4	41.8	59.0	42.5
13	53.7	47.5	61.7	44.0	57.8	50.0	57.2	43.7	53.1	45.2	60.8	42.7	59.6	42.2
14	54.5	47.0	62.5	43.3	57.7	50.9	57.7	43.6	52.5	44.6	60.0	42.4	59.2	43.2
15	55.0	46.5	62.3	44.4	57.4	51.5	58.0	43.6	54.3	45.6	62.3	41.4	58.6	41.0
16	54.4	46.3	62.2	45.4	56.9	49.2	57.8	43.0	53.1	42.6	62.0	42.4	59.0	41.6
17	54.5	45.8	61.0	44.7	56.4	49.8	57.4	43.3	53.2	45.0	61.0	42.6	59.4	42.4
18	54.1	46.1	62.5	44.9	56.9	49.8	57.7	43.6	52.8	44.5	60.1	43.0	58.8	41.7
19	53.7	45.8	62.0	44.8	57.8	49.3	58.0	43.5	53.3	44.2	60.9	41.1	60.2	41.0
20	53.2	48.3	61.4	43.5	58.1	49.9	57.3	43.1	53.6	44.4	60.9	42.3	58.7	41.8
21	53.6	45.1	62.7	45.3	58.0	49.9	58.3	43.4	53.6	44.9	61.6	43.3	59.2	42.8
22	54.5	47.0	61.3	45.1	57.0	48.9	57.6	43.7	53.2	45.8	60.9	43.0	59.0	42.0
23	54.2	46.4	63.1	45.4	56.6	50.0	57.9	42.9	53.4	44.6	61.3	42.3	60.6	42.7
24	54.3	46.0	62.8	45.2	57.5	49.5	57.1	43.1	52.7	44.4	60.7	43.8	59.9	41.8
25	53.9	46.2	62.2	44.7	56.3	48.7	57.5	43.9	53.9	44.3	61.9	42.4	59.4	42.1

In future work, we plan to extend WFCA as a paralleling method.

IV. EXPERIMENTAL EVALUATION

A. DATA DESCRIPTION

In this section, we evaluate WFCA on eight representative real-world networks selected from distinct fields, including one transportation network (Aviation), one biological network (Protein), one information network (Blogs), two infrastructure networks (Powergird and Euroroad), one social network (Friendships), two collaboration networks (Ca-AstroPh and DBLP). In brief, Aviation is an air traffic control network of USA's FAA (Federal Aviation Administration) and this is a directed network. Protein is a directed network of interactions between proteins in Humans. Our Blogs is a directed network which contains front-page hyperlinks between blogs in the context of the 2004 US election. Our Powergird network contains information about the power grid of the Western States of America and this is undirected. Euroroad is the international E-road undirected network which located mostly in Europe. Our Friendships is an undirected network which contains friendships between users of the website hamsterster.com. Ca-AstroPh is a collaboration graph of authors of scientific papers from the arXiv's Astrophysics (astro-ph) section. And DBLP is an attributed network of computer science bibliography. To extensively study the performance of our algorithm, we compare WFCA with several representative influential node detection algorithms, namely PageRank (PR), HITS, eigenvector centrality (EC), K-shell, H-index, closeness centrality (CC), and betweenness centrality (BC). The statistics of the eight networks datasets are presented in Table 4, and they are all publicly available from KONECT (http://konect.uni-koblenz.de/) and the DBLP dataset (http://dblp.uni-trier.de/).

B. MEASUREMENT

In this paper, we employ the SIR model [42] to investigate the spreading influences of ranked nodes. There are three components to such a system: (I) Susceptible (S) denotes the susceptible individuals who are not yet infected; (II) Infected (I) represents the infected individuals, who may spread the disease to susceptible individuals; (III) Recovered (R) stands for recovered individuals, who can never be infected again. The SIR model begins with one or more seed nodes. Then, the seed nodes infect adjacent nodes with a probability of λ. Next, the infected nodes recover with a probability of μ. Finally, the infected and recovered nodes are used to calculate the spreading influences of seed nodes. Each loop of spreading is regarded as a time step t. $F(t)$ denotes the number of infected and recovered nodes at time t, which is an indicator of the node importance. Obviously, $F(t)$ will gradually converge as the time t evolves, trending towards to a certain level.

To evaluate the performances of different influential node identification algorithms, Kendall [43] τ is introduced to measure the correlation of the node spreading influence with the eight methods. Kendall's tau as a rank correlation coefficient is usually used to measure the correlation between two ranking list. We assume that two sequences associated with the same number of nodes n, $X = (x_1, x_2, \ldots, x_n)$ and $Y = (y_1, y_2, \ldots, y_n)$. Any pair of two-tuples (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 if both $x_i < x_j$ 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

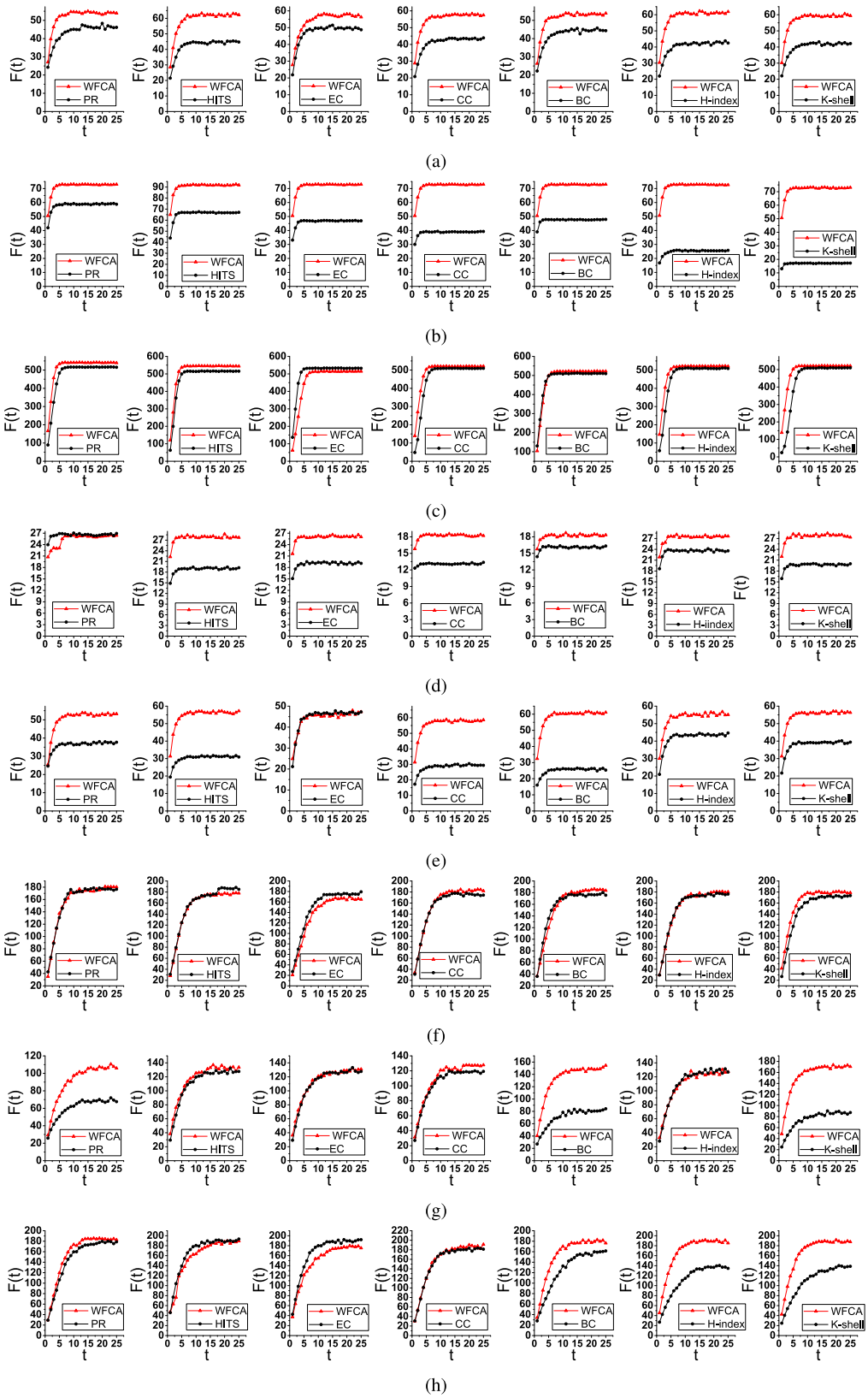


FIGURE 7. The propagation influence of the top-10 different nodes of ranking lists between WFCA and other algorithms, F(t) denotes the number of infected and recovered nodes at time t, and t varies from 1 to 25. Results are obtained by averaging over 1000 implementations and the spreading probability $\lambda = 0.1$. (a) Aviation. (b) Protein. (c) Blogs. (d) Powergrid. (e) Euroroad. (f) Friendships. (g) Ca-Astroph. (h) DBLP.

pair is neither concordant nor discordant. The Kendall's tau coefficient is defined as

$$\tau(X, Y) = \frac{n_c - n_d}{0.5n(n-1)} \tag{4}$$

In which n_c and n_d represent the number of concordant and discordant pairs respectively. One can note that, $\tau \in [-1, 1]$ is positively related to concordant of the ranking lists. The higher τ value is, the more accurate the ranking list the method could generate.

C. PERFORMANCE EVALUATION

In this experiment, to better distinguish the importance of nodes, we use relatively small values of λ in SIR model, namely $\lambda \in [0.01, 0.1]$. Because with a large λ, the spreading would cover almost all the network [26]. Fig. 5 shows Kendall's τ of the WFCA method where the ranking lists are generated by the PR, HITS, EC, K-shell, H-index, CC and BC. As shown in Fig. 5c, the Kendall's τ for the WFCA is between 0.8 and 0.9 for $\lambda \in [0.01, 0.1]$, indicating that the ranking lists generated by the WFCA and the real SIR spreading process are basically identical to each other. From Fig. 5, we can see that there are different performances for the different algorithms and the WFCA method performs well on different types of networks. For instance, for directed networks (Aviation, Protein and Blogs) and undirect networks (Powergrid, Euroroad and Ca-AstroPh) WFCA performs the best. For the Friendships network, EC and HITS have the best performance, and the WFCA also has a better effect than the others. In DBLP networks, the EC algorithm achieves the best performance, the WFCA method still provides a comparatively good performance and performs better than other algorithms. Note that the performances of K-shell, BC and CC are not good in Aviation, Protein, Blogs and Powergrid networks. Because, the K-shell algorithm is coarse-grained on the network division, and the nodes of each layer are seen as equally important, but in fact, they are not. Therefore, for the network with small number of layers, the effect is relatively poor. On the contrary, for multiple layers network, the effect is better. For the BC method, there are many nodes whose values are the same, because, these nodes have the same number of shortest paths through them. And the cc algorithm also has the same problem. The performance of H-index method is also not good in directed networks (Aviation, Protein and Blogs). As you can see, WFCA performs well not only in undirected networks, but also in directed networks.

To further compares the propagation performance of the algorithms, we investigate the spreading influences of ranked nodes in SIR model. Without loss of generality, we set the infection probability $\lambda = 0.1$, recovery probability $\mu = 1$ [44], [45], and time step $t = 500$. First, we compute the influence of each node using the various algorithms, and rank them in descending order. Table 5 presents the top-L ranked nodes. Owing to space limitations, we only display the top-10 nodes of two networks, including one directed network Aviation and one undirected network Euroroad. We can see that most nodes of top-10 of WFCA are presented in other algorithms, so, the WFCA method has better accuracy than other algorithms. For further evaluation, each ranked node is considered as the seed node (with only one seed node for each run). Finally, we calculate the number of infected nodes for each seed node by averaging over 1000 implementations.

Fig. 6 illustrates the average numbers of infected nodes ranked by the eight different algorithms. A good algorithm should result in a downward slope from left to right; that is, the number of infected nodes should decrease as L increases. This is because the more important a node is, the more infected nodes there are. Thus, this node ranks higher. In the Aviation, Protein, Blogs, Powergrid, Euroroad and aCa-AstroPh networks, WFCA performs the best among all eight algorithms. For the Friendships and DBLP networks, although EC achieves the best performance, WFCA still provides a comparatively good performance. Note that the WFCA method can not only rank the nodes by considering the topological structure of a network, but it can also rank nodes by considering the attributes of nodes (see Fig. 6h). From the tables and figures, we observe that WFCA delivers a superior spreading effect compared with the other algorithms.

Furthermore, we compare the influence of the top-10 nodes that are discriminatively selected by WFCA and other algorithms. All top-10 different nodes are used as seed nodes and the time step t is set ranging from 1 to 25. Table 6 further presents the propagation influence of top-10 different nodes in the Aviation network. We can observe that the number of accumulative infected nodes $F(t)$ increases as the time step t increases, and eventually obtains a steady value after several time points. Since there are 10 seed nodes, the propagation of most networks reaches a steady state at the time step $t = 15$ and we can clearly investigate the spread effect of WFCA and other algorithms. In addition, Fig. 7 illustrates the influences of the different top-10 nodes in eight networks. Note that the differences between two approaches can be distinguished effectively by investigating the effects of discriminative nodes of the two ranking lists. Fig. 7 shows that the WFCA method has a good spreading efficiency of top-10 different nodes. Specifically, WFCA has the best performance on the networks of Aviation, Protein, Powergrid, Euroroad and Ca-Astroph. In Friendships and DBLP networks, EC and HITS have the best propagation effect, and WFCA also has better spreading influence than other algorithms.

V. CONCLUSION

In this paper, we have considered the problem of detecting influential nodes based on weighted formal concept analysis. This method considers global information regarding a given network, and converts binary relationships between nodes in the network into a hierarchy. Then, nodes are aggregated according to their attributes, to rank node importance. To evaluate the efficiency of WFCA, we conducted experiments on eight real networks, and compared WFCA with several representative influential node detection algorithms.

These experiments further demonstrated the superiority of WFCA over state-of-the-art algorithms. In future work, we plan to extend WFCA as a paralleling method.

REFERENCES

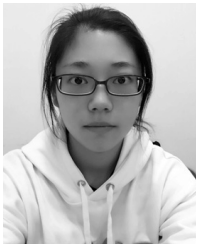
[1] J. Shao, Z. Han, Q. Yang, and T. Zhou, "Community detection based on distance dynamics," in *Proc. 21th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining KDD*, 2015, pp. 1075–1084.
[2] D. Chen, L. Lü, M.-S. Shang, Y.-C. Zhang, and T. Zhou, "Identifying influential nodes in complex networks," *Phys. A, Statist. Mech. Appl.*, vol. 391, pp. 1777–1787, Feb. 2012.
[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] J. Shao, C. Böhm, Q. Yang, and C. Plant, "Synchronization based outlier detection," in *Machine Learning and Knowledge Discovery in Databases*. Berlin, Germany: Springer, 2010, pp. 245–260.
[5] J. Shao, Q. Yang, H.-V. Dang, B. Schmidt, and S. Kramer, "Scalable clustering by iterative partitioning and point attractor representation," *ACM Trans. Knowl. Discov. Data*, vol. 11, no. 1, pp. 5:1–5:23, Jul. 2016.
[6] M. Perc, "The matthew effect in empirical data," *J Roy. Soc. Interface*, vol. 11, no. 98, p. 20140378, Jul. 2014.
[7] Y. Sun, Y. Ma, F. Zhang, Y. Ma, and W. Shen, "Key nodes discovery in large-scale logistics network based on MapReduce," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 2015, pp. 1309–1314.
[8] M. T. Irfan and L. E. Ortiz, "On influence, stable behavior, and the most influential individuals in networks: A game-theoretic approach," *Artif. Intell.*, vol. 215, pp. 79–119, Oct. 2014.
[9] F. D. Malliaros, M.-E. G. Rossi, and M. Vazirgiannis, "Locating influential nodes in complex networks," *Sci. Rep.*, vol. 6, p. 19307, Jan. 2016.
[10] M. R. D'Orsogna and M. Perc, "Statistical physics of crime: A review," *Phys. Life Rev.*, vol. 12, pp. 1–21, Mar. 2015.
[11] Z. Wang *et al.*, "Statistical physics of vaccination," *Phys. Rep.*, vol. 664, pp. 1–113, Dec. 2016.
[12] D. Helbing *et al.*, "Saving human lives: What complexity science and information systems can contribute," *J. Statist. Phys.*, vol. 158, no. 3, pp. 735–781, Jun. 2015.
[13] A. Szolnoki and M. Perc, "Collective influence in evolutionary social dilemmas," *EPL (Europhys. Lett.)*, vol. 113, no. 5, p. 58004, Mar. 2016.
[14] J. Hu, Y. Du, H. Mo, D. Wei, and Y. Deng, "A modified weighted TOPSIS to identify influential nodes in complex networks," *Phys. A, Statist. Mech. Appl.*, vol. 444, pp. 73–85, Feb. 2016.
[15] S. Aral and D. Walker, "Identifying influential and susceptible members of social networks," *Science*, vol. 337, no. 6092, pp. 337–341, Jun. 2012.
[16] N. Agarwal, H. Liu, L. Tang, and P. S. Yu, "Identifying the influential bloggers in a community," in *Proc. Int. Conf. Web Search Data Mining*, 2008, pp. 207–218.
[17] A. Özgür, T. Vu, G. Erkan, and D. R. Radev, "Identifying gene-disease associations using centrality on a literature mined gene-interaction network," *Bioinformatics*, vol. 24, no. 13, pp. i277–i285, Jun. 2008.
[18] P. Bonacich, "Factoring and weighting approaches to status scores and clique identification," *J. Math. Sociol.*, vol. 2, no. 1, pp. 113–120, Jan. 1972.
[19] G. Sabidussi, "The centrality index of a graph," *Psychometrika*, vol. 31, no. 4, pp. 581–603, Dec. 1966.
[20] L. C. Freeman, "Centrality in social networks conceptual clarification," *Social Netw.*, vol. 1, no. 3, pp. 215–239, Jan. 1978.
[21] 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.
[22] L. Lü, Y.-C. Zhang, C. H. Yeung, and T. Zhou, "Leaders in social networks, the delicious case," *PLoS ONE*, vol. 6, no. 6, p. e21202, Jun. 2011.
[23] 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.
[24] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *J. ACM*, vol. 46, no. 5, pp. 604–632, 1999.
[25] R. Wille, "Restructuring lattice theory: An approach based on hierarchies of concepts," in *Ordered Sets*. Amsterdam, The Netherlands: Springer, 1982, pp. 445–470.
[26] M. Kitsak *et al.*, "Identification of influential spreaders in complex networks," *Nature Phys.*, vol. 6, pp. 888–893, Aug. 2010.
[27] 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, p. 056103, May 2005.
[28] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, Oct. 1999.
[29] D.-B. Chen, H. Gao, and L. Lü, and T. Zhou, "Identifying influential nodes in large-scale directed networks: The role of clustering," *PLoS ONE*, vol. 8, no. 10, p. e77455, Oct. 2013.
[30] M. Wermelinger, Y. Yu, and M. Strohmaier, "Using formal concept analysis to construct and visualise hierarchies of socio-technical relations," in *Proc. 31st Int. Conf. Softw. Eng.-Companion (ICSE-Companion)*, May 2009, pp. 327–330.
[31] V. Dufour-Lussier, J. Lieber, E. Nauer, and Y. Toussaint, "Text adaptation using formal concept analysis," in *Case-Based Reasoning. Research and Development*. Berlin, Germany: Springer, 2010, pp. 96–110.
[32] J. Poelmans, P. Elzinga, S. Viaene, and G. Dedene, *Formal Concept Analysis in Knowledge Discovery: A Survey* (Lecture Notes in Computer Science). Berlin, Germany: Springer, 2010, pp. 139–153.
[33] A. Formica, "Ontology-based concept similarity in formal concept analysis," *Inf. Sci.*, vol. 176, no. 18, pp. 2624–2641, Sep. 2006.
[34] U. Priss, "Linguistic applications of formal concept analysis," in *Formal Concept Analysis*. Berlin, Germany: Springer, 2005, pp. 149–160.
[35] V. Codocedo and A. Napoli, "Formal concept analysis and information retrieval—A survey," in *Formal Concept Analysis*. Berlin, Germany: Springer, 2015, pp. 61–77.
[36] K. Sumangali and C. A. Kumar, "Determination of interesting rules in FCA using information gain," in *Proc. 1st Int. Conf. Netw. Soft Comput. (ICNSC)*, Aug. 2014, pp. 304–308.
[37] D. G. Kourie, S. Obiedkov, B. W. Watson, and D. van der Merwe, "An incremental algorithm to construct a lattice of set intersections," *Sci. Comput. Programm.*, vol. 74, no. 3, pp. 128–142, Jan. 2009.
[38] L. Zou, Z. Zhang, and J. Long, "A fast incremental algorithm for constructing concept lattices," *Expert Syst. Appl.*, vol. 42, no. 9, pp. 4474–4481, Jun. 2015.
[39] J. Outrata and V. Vychodil, "Fast algorithm for computing fixpoints of Galois connections induced by object-attribute relational data," *Inf. Sci.*, vol. 185, no. 1, pp. 114–127, Feb. 2012.
[40] S. Andrews, "In-close2, a high performance formal concept miner," in *Proc. Int. Conf. Conceptual Struct.*, 2011, pp. 50–62.
[41] M. A. Babin and S. O. Kuznetsov, "On links between concept lattices and related complexity problems," in *Formal Concept Analysis*. Berlin, Germany: Springer, 2010, pp. 138–144.
[42] S. Tuljapurkar, "Infectious diseases of humans: Dynamics and control," *Science*, vol. 254, no. 5031, pp. 591–593, Oct. 1991.
[43] M. G. Kendall, "A new measure of rank correlation," *Biometrika*, vol. 30, nos. 1–2, pp. 81–93, Jun. 1938.
[44] J.-G. Liu, J.-H. Lin, Q. Guo, and T. Zhou, "Locating influential nodes via dynamics-sensitive centrality," *Sci. Rep.*, vol. 6, nos. 1–2, p. 21380, Feb. 2016.
[45] L. Lü, T. Zhou, Q.-M. Zhang, and H. E. Stanley, "The H-index of a network node and its relation to degree and coreness," *Nature Commun.*, vol. 7, nos. 1–2, Jan. 2016, Art. no. 10168.

ZEJUN SUN received the B.Sc. degree in computer science from Henan Polytechnic University in 2003 and the M.Sc. degree in computer science from Xidian University, China, in 2008. 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.

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. He is currently a Professor with the School of Information Science and Engineering, Central South University. 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. She is currently an Associate Professor with the School of Information Science and Engineering, Central South University. Her research interests include transparent computing and big data processing.

YIXIANG HU received the B.Sc. degree in computer science from Hengyang Normal University, China, in 2004, and the M.Sc. degree in computer science from the University of South China, China, in 2007. She is currently pursuing the Ph.D. degree with Central South University, China. Her research interests include network and information security issues.



YIHAN WANG received the B.Sc. degree in Internet of Things engineering from Central South University, China, in 2015, where she is currently pursuing the M.Sc. degree. Her research interests include software engineering and big data.

JUNMING SHAO received the Ph.D. (*Summa Cum Laude*) degree (Hons.) from the University of Munich, Germany, in 2011. He became the Alexander von Humboldt Fellow in 2012. He not only authored papers on top-level data mining conferences, such as KDD, ICDM, and SDM (two of those papers have won the Best Paper Award), but also authored data mining-related interdisciplinary work in leading journals, including the *Brain*, the *Neurobiology of Aging*, and the *Water Research*. His research interests include data mining and neuroimaging.