Knowledge-Based Systems xxx (xxxx) xxx

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Finding influential nodes in social networks based on neighborhood correlation coefficient☆

Ahmad Zareie [a], Amir Sheikhahmadi [b,*], Mahdi Jalili [c], Mohammad Sajjad Khaksar Fasaei [b]

[a] School of Computer Science, University of Manchester, UK
[b] Department of Computer Engineering, Sanandaj Branch, Islamic Azad University, Sanandaj, Iran
[c] School of Engineering, RMIT University, Melbourne, Australia

ARTICLE INFO

Article history:
Received 15 September 2018
Received in revised form 1 November 2019
Accepted 25 January 2020
Available online xxxx

Keywords:
Social networks
Influential nodes
Influence range
Information propagation
Susceptible–Infected–Recovered model

ABSTRACT

Finding the most influential nodes in social networks has significant applications. A number of methods have been recently proposed to estimate influentiality of nodes based on their structural location in the network. It has been shown that the number of neighbors shared by a node and its neighbors accounts for determining its influence. In this paper, an improved cluster rank approach is presented that takes into account common hierarchy of nodes and their neighborhood set. A number of experiments are conducted on synthetic and real networks to reveal effectiveness of the proposed ranking approach. We also consider ground-truth influence ranking based on Susceptible–Infected–Recovered model, on which performance of the proposed ranking algorithm is verified. The experiments show that the proposed method outperforms state-of-the-art algorithms.

1. Introduction

Rapid propagation of news and advertisements over social networks has created new opportunities for social media platforms to replace conventional means of advertisement such as through radio, television, and banners [1,2]. Currently, social networks are maybe the most important advertising platforms, on which many advertising messages are exchanged every day [3,4]. It is almost impossible to directly reach out all users, and a clever solution is to target limited number of influential users and communicate with them with the hope that they can maximize the desired influence on others [5,6]. If the target users are selected carefully, they can spread the messages more widely by propagating them among their friendship networks.

Users do not have the same importance, and some of them have higher influence given their personal and social characteristics and being placed in strategic locations. Influential users play more effective roles that others in the success of information spread. Specifying the influence range of users and ranking them based on their influence range has attracted much attention in recent years [7–10]. In many cases, the only available information to discover influential nodes, is the structure of the connection network. A number of techniques have been proposed in the literature to use network structure to find influential users [7,11–17].

In order to discover influential nodes, one can use various centrality measures such as degree [18], betweenness [11], closeness [19], and k-shell [12], where the nodes with high centralities are considered to be more influential that those with lower centralities. Given that social graphs are often large-scale, degree centrality is a time efficient method, but it suffers from low accuracy. Although considering average degree of neighbors can improve the accuracy of degree centrality [20,21], disregarding interaction among neighbors might have negative impact on the accuracy [22–24]. Cluster rank [22] tries to tackle this and aims to estimate nodes' influentiality based on degree of their neighbors and their interaction. The intuition behind this method is that users sharing more friends with their neighbors tend to propagate the information over shorter ranges. In cluster rank method, only the number of friends shared by each user and their neighbors is taken into account. In this work, we further consider properties of the commonalities of the users with their neighbors. To estimate influence score of a node, the proposed method uses similarity (or correlation) of connection structure of the node and its neighbors. The proposed method is applied on a number of synthetic and real networks and compared with a number of state-of-the-art influence ranking algorithms. The major contribution of the proposed algorithm is to consider detailed correlation structure between the neighborhood and effectively use it to discover the most influential nodes. We apply Pearson correlation for this purpose.

☆ No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.knosys.2020.105580.

* Corresponding author.
E-mail address: asheikhahmadi@iausdj.ac.ir (A. Sheikhahmadi).

https://doi.org/10.1016/j.knosys.2020.105580
0950-7051/

The rest of the paper is as follows. Section 2 provides a review of previous works. The proposed method is presented in Section 3. Section 4 includes experimental settings, results and discussions. Finally, the paper is concluded in Section 5.

2. Related works

In this section, we briefly review the existing works to discover influential nodes in complex networks. A number of centrality measures have been proposed to estimate influence of nodes in effective information propagation. Degree centrality [18] is the simplest metric to determine influential nodes. However, it frequently happens that nodes with smaller degrees have much higher influence than those with higher degrees. Indeed, a low-degree node might be located in a strategic location of the network such that it facilitates information spread more than some nodes with higher degrees [12]. Given the degrees of the neighbors of each node and based on h-index measure, a centrality metric was proposed in [25] that is more effective than degree to determine influential nodes. [26] further considered h-index of the neighborhood set in combination with the h-indices of the nodes and obtained a ranking method with better accuracy. Another extension of h-index centrality was proposed in [16]. Askari-Sichani and Jalili showed that nodes with low degrees that are connected to high-degree nodes can be good candidates to be influential nodes [27]. In the method based on entropy centrality [15], the influence range of a node is determined based on degree distribution of its first and second-order neighbors.

Kitsak et al. [12] introduced a measure based on closeness of a node to the graph core. They used k-shell decomposition approach, where nodes located in shells closer to the core are identified as influential nodes. In order to obtain the shells, nodes of degree 1 are removed from the graph in the first step. The removed nodes are considered as k-shell $= 1$. In the second step of the algorithm, nodes of degree 2 are removed from the graph, to obtain nodes belonging to k-shell $= 2$. The procedure continues until higher order shells are extracted. Nodes located at higher order shells are closer to the graph core. Upon removal of a neighbor of a node in the k-shell decomposition algorithm, the number of its remaining neighbors may better indicate its influence range. Mixed Degree Decomposition (MMD) centrality [28] used this idea and introduced a ranking algorithm with better accuracy than the original coreness proposed in [12]. Bae and Kim [7] proposed a method that related influentiality to nodes whose neighbors are closer to the graph core. Wang et al. [13] argued that disregarding the iterations in which nodes are removed in the k-shell decomposition algorithm, may reduce the accuracy. They presented an improved version of k-shell algorithm to overcome this problem. Hierarchal k-shell (HKS) [14] was proposed as a hierarchical approach to improve precision of identifying influential nodes. Li et al. [29] proposed a method that divides the neighbors of each node into four different classes based on the order of their removal. The authors in [30] showed that significance of the links between a node and its neighbors has also important role in the influentiality of the node. They proposed a ranking algorithm based on degree, k-shell and link significance. [17] proposed a method by first calculating influence sphere neighbors using k-shell, and then determining the centrality of nodes based on the sphere diversity. Direct and indirect influence of nodes were both taken into account in [31] to propose a centrality measure. A randomized spanning tree approach is proposed in [32] to determine influential nodes based on local and global connection patterns of the network. According to the distance between nodes, a method is proposed in [33] to rank influential nodes based on fuzzy local dimension. In [34], the effect of the node removal on average shortest paths of network is assessed to propose a centrality measure.

In some research studies, several centrality measures were taken into account to determine influentiality. In [35], it was estimated using evidential theory based on nodes' topological information such as degree, betweenness, and correlation. Another method was introduced in [36] to identify nodes' influentiality based on their degree, betweenness, and closeness. An entropy weighting method was proposed in [37] to determine the influentiality based on nodal centrality measures. Chen et al. [22] showed that increased inter-connectedness between neighborhood set of a node often reduces the likelihood of widespread propagation of messages initiated from that node. They proposed cluster rank measure that is based on local clustering concept. Clustered local degree [38] is another centrality measure to determine influential nodes that is based on local clustering of a node and the degrees of its neighbors. Our proposed method is based on structural similarity between a node and its neighbors, and experiments show that it outperforms state-of-the-art methods.

3. The proposed method

Let us denote the social network as graph $G = (V, E)$, where users are $V = \{v_1, v_2, \ldots, v_{|V|}\}$ and the relationships between them are $E = \{e_1, e_2, \ldots, e_{|E|}\}$. In the remainder of the paper, N_i is used to represent the set of neighbors of node v_i. The number of neighbors of node v_i determines its degree, and is shown by d_i.

Cluster rank method [22] regards relationships between the neighbors of a node as a factor affecting its influence range. It has been shown that strong interactions among neighbors of a node often have negative impacts on its influentiality [23,24]. In other words, if the neighbors are tightly inter-connected, it might cause messages to be spread in small cycles of the graph, which might reduce the likelihood of reaching out large portion of the network. The extent of relationships between the neighbors of a node can be expressed with the notion of local clustering coefficient. The local clustering coefficient of node v_i (LC_i) is calculated as Eq. (1):

$$LC_i = \sum_{v_j \in N_i} \frac{|N_i \cap N_j|}{d_i\,(d_i - 1)} \tag{1}$$

Indeed, the local clustering coefficient examines the number of neighbors shared by a node and each of its neighbors. We aim at finding effective measures of correlations between a node and its neighbors by examining what the node has in common with its neighbors. In the proposed approach, referred to as Extended Cluster Coefficient Ranking Measure (ECRM), we consider common hierarchy between a node and its neighbors rather than the number of their common neighbors. To this end, the network is first divided into different parts. Nodes that are removed in the same iterations (IT) of the k-shell algorithm can be assumed to be in the same hierarchy. Consider the graph in Fig. 1 for instance. This graph is composed of 3 shells that are shown by different colors. Inspired by the k-shell decomposition, the following process is applied to determine the iteration (IT) in which the nodes are removed. First, IT counter is set at 1. The nodes with the lowest degree are removed from the graph; these nodes are considered in the first hierarchy, and $IT = 1$ is assigned to them. In the next stage, IT counter is increased by one, and again the nodes with the lowest degree are removed from the graph. Removed nodes in this stage are considered in the second hierarchy, and $IT = 2$ is assigned to them. This process is iterated until there remains no node in the graph, and at each iteration IT counter is increased by one and assigned to those nodes removed in that stage. As can be seen in Fig. 1, node 8 has a neighbor, 16, in hierarchy 3 ($IT = 3$). Node 9 has also a neighbor, 10, in this hierarchy. Thus, this hierarchy is considered as a commonality between these two nodes.

Please cite this article as: A. Zareie, A. Sheikhahmadi, M. Jalili et al., Finding influential nodes in social networks based on neighborhood correlation coefficient, Knowledge-Based Systems (2020) 105580, https://doi.org/10.1016/j.knosys.2020.105580.

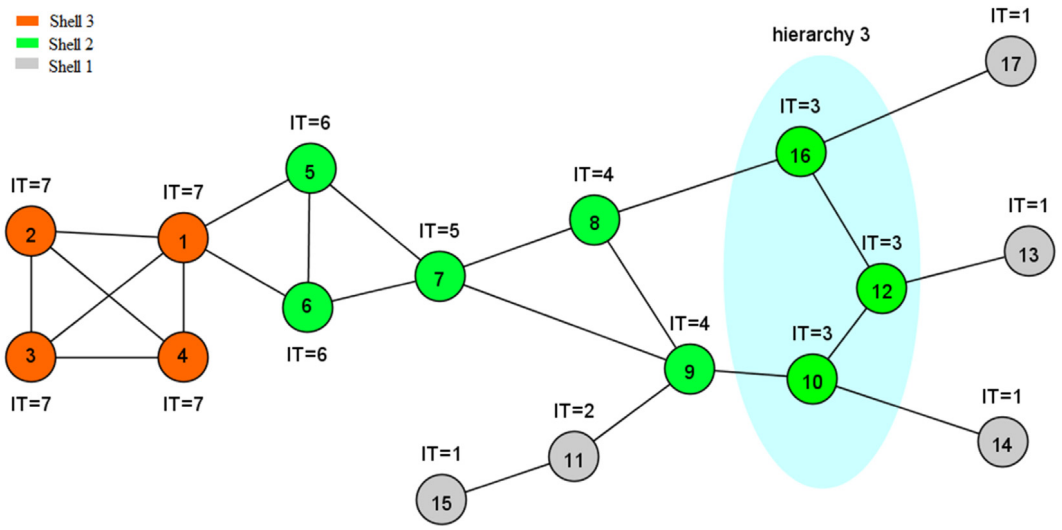

Fig. 1. A sample graph with 3 shells. The graph has 3 shells each shown by different colors. It is decomposed into 7 hierarchies. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

After decomposing the network into different hierarchies, a shell vector is calculated for each node v_i based on the hierarchy of its neighbors.

Definition 1 (*Shell Vector*)**.** The shell vector for node v_i is defined as

$$SV_i = \left\{ \left|N_i^{(1)}\right|, \left|N_i^{(2)}\right|, \left|N_i^{(3)}\right|, \ldots, \left|N_i^{(f)}\right| \right\} \tag{2}$$

where $\left|N_i^{(k)}\right|$ indicates the number of neighbors of node v_i belonging to hierarchy k, and f is the maximum hierarchy in the network.

In the graph of Fig. 1, we have $f = 7$, as the maximum iteration is 7. For instance, node 8 has one neighbor in hierarchy 3, one neighbor in hierarchy 4, and one neighbor in hierarchy 5, thus the values of the 3rd, 4th , and 5th cells of shell vector SV_8 are 1, 1, and 1, respectively, while it is 0 for other cells. Node 9 has 1, 1, 1, and 1 neighbor in hierarchies of 2, 3, 4, and 5, respectively, thus the values of the 2nd, 3rd, 4th, and 5th cells of SV_9 are 1, 1, 1, and 1, respectively, and it is 0 for other cells. Thus, the shell vectors for nodes 8, 9 and 7 are:

$$\begin{aligned} SV_8 &= \{0, 0, 1, 1, 1, 0, 0\} \\ SV_9 &= \{0, 1, 1, 1, 1, 0, 0\} \\ SV_7 &= \{0, 0, 0, 2, 0, 2, 0\} \end{aligned}$$

ECRM is based on correlation between nodes, which is calculated using Pearson's correlation coefficient between the shell vectors. In order to measure commonality between neighbors of two nodes, we consider Pearson correlation between their shell vector, which is obtained as:

$$C_{i,j} = \frac{\sum_{k=1}^{f}(SV_i^{(k)} - \overline{SV}_i)(SV_j^{(k)} - \overline{SV}_j)}{\sqrt{\sum_{k=1}^{f}\left(SV_i^{(k)} - \overline{SV}_i\right)^2}\sqrt{\sum_{k=1}^{f}\left(SV_j^{(k)} - \overline{SV}_j\right)^2}} \tag{3}$$

where $SV_i^{(k)} = \left|N_i^{(k)}\right|$ represents the value of the kth cell in vector SV_i, and $\overline{SV}_i$ indicates the mean value in the shell vector. Given that the sum of the values in SV_i is equal to the degree of node v_i, we have $\overline{SV}_i = \frac{d_i}{f}$. The correlation coefficient between two vectors is always in the range $[-1, +1]$, where 0 indicates lack of correlation, negative values denote indirect correlation, and positive values suggest direct correlation. For example, $\overline{SV}_8 = \frac{3}{7}$ and $\overline{SV}_9 = \frac{4}{7}$, and the correlation between these nodes is calculated as

$$C_{8,9} = \frac{\sum_{k=1}^{f}(SV_8^{(k)} - \frac{3}{7})(SV_9^{(k)} - \frac{4}{7})}{\sqrt{\sum_{k=1}^{f}\left(SV_8^{(k)} - \frac{3}{7}\right)^2}\sqrt{\sum_{k=1}^{f}\left(SV_9^{(k)} - \frac{4}{7}\right)^2}} \cong 0.75$$

Definition 2 (*Shell Clustering Coefficient*)**.** The shell clustering coefficient of node v_i is calculated as

$$SCC_i = \sum_{v_j \in N_i}\left(\left(2 - C_{i,j}\right) + \left(2\frac{d_j}{\max(d)} + 1\right)\right) \tag{4}$$

The above equation has two parts. The correlation coefficient is included in the first part and the degree in the second. Higher correlation between node v_i and each of its neighbors has negative effects on spreading ability of v_i. Thus, $\left(2 - C_{i,j}\right)$ is used for calculation of the shell clustering coefficient of node v_i. In addition to the correlation between node v_i and its neighbor v_j, the degree of the latter is also effective on its shell clustering coefficient. Therefore, the term $\left(2\frac{d_j}{\max(d)} + 1\right)$ is considered in Eq. (4), where $\max(d)$ is the maximum degree in the graph.

Application of the neighborhood rule can increase accuracy of determining the most influential nodes [39]. In the proposed approach, the neighborhood rule is first applied for calculation of the Cluster Coefficient Ranking Measure (CRM) for each node given the shell clustering coefficients of the neighbors. Eq. (5) shows how this measure is calculated for node v_i. Along the same lines, ECRM is obtained by summation of the CRMs for the neighbors. Eq. (6) shows how ECRM is calculated for node v_i.

$$CRM_i = \sum_{v_j \in N_i} SCC_j \tag{5}$$

$$ECRM_i = \sum_{v_j \in N_i} CRM_j \tag{6}$$

Algorithm 1 shows the pseudo code of the proposed method that also includes the details of the calculation steps.

In the above algorithm, the k-shell algorithm is applied in line 4 to determine the *IT* values for all nodes. The shell vector is calculated for each node in lines 5–7 using Eq. (2). In lines 8–14,

Algorithm 1: Pseudo-code of ECRM method

```
01  Input: G = (V,E)   // G is undirected and unweight graph with V as the set of nodes
                       and E as the set of edges.
02  Output: A ranking list of nodes' influentiality.
03  Begin
04      Assign IT(v_i) for each v_i ∈V using the k-shell algorithm
05      for i = 1 to |V|
06          Calculate SV_i using Eq. (2)
07      End for
08      for i = 1 to |V|
09          Set SCC_i=0
10          for each v_j ∈ N_i
11              Calculate C_{i,j} using Eq. (3)
12              SCC_i = SCC_i + (2 − C_{i,j}) + (2 d_j / max(d) + 1)
13          End for
14      End for
15      for i = 1 to |V|
16          Set CRM_i=0;
17          for each v_j ∈ N_i
18              CRM_i = CRM_i + SCC_j
19          End for
20      End for
21      for i = 1 to |V|
22          Set ECRM_i=0;
23          for each v_j ∈ N_i
24              ECRM_i = ECRM_i + CRM_j
25          End for
26      End for
27      Sort the nodes in descending order based on ECRM scores to obtain the ranking
        list.
28  End
```

the correlation $C_{i,j}$ between node v_i and every $v_j \in N_i$ is calculated using Eq. (3), and then the shell clustering coefficient SCC_i is calculated for v_i. This process is repeated for all nodes. Then, the Cluster Coefficient Ranking Measure CRM_i is calculated for each $v_i \in V$ according to SCC index of its neighbors. In lines 21–26, the Extended Cluster Coefficient Ranking Measure $ECRM_i$ is calculated for each $v_i \in V$. Finally, nodes are sorted based on ECRM index to obtain the ranking. Time complexity of the proposed method includes:

- The complexity of the k-shell decomposition algorithm that is $O(|E|)$.
- The complexity of lines 5–7 that is $O(|V|)$.
- It is $O(|V|\langle d\rangle f)$ for lines 8–14, with $\langle d\rangle$ and f being the average degree of nodes and the maximum IT, respectively. Often, we have $|V| \gg \langle d\rangle$ and $|V| \gg f$ in large graphs, and thus the complexity is reduced to O $(|V|)$ for large graphs.
- The complexity of each section in lines 15–20 and 21–26 is $O(|V|)$.

Thus, the time complexity of ECRM is $O(|E| + |V|)$.

4. Experimentation results

To assess the proposed approach, its performance is compared to a number of state-of-the-art methods including k-shell (Ks) [12], cluster rank (CR) [22], mixed degree decomposition (MDD) [28], extended neighborhood coreness (Cnc+) [7], k-shell iteration factor (KS-IF) [13], H-index [25], Mixed Core, Semi-local Degree and Weighted Entropy (CDE) [8], Entropy based ranking measure (ERM) [15], classified neighbors (CN) [29], clustered local degree (CLD) [38], local H-index (LH) [26], and link significance (LS) [30] methods. These method are applied on 9 real-world datasets including Dolphins [40], Copperfield [41], NetScience [41], C-elegans [42], EuroRoad [43], Chicago [44], Hamsterster [45], PowerGrid [46], and PGP [47]. Moreover, two synthetic datasets are generated using Lancichinetti–Fortunato–Radicchi (LFR) model [48]. LFR is a benchmark for generation of networks with community structure. It has a number of tuning parameters including the number of nodes ($|V|$), the average degree of nodes ($\langle d\rangle$), the mixing parameter of the community structure (μ), and the power-law of the degree distribution (γ). For generation of LFR-200 dataset, the parameter values are set to $\gamma = 2$, $\langle d\rangle = 5$, $|V| = 200$, and $\mu = 0.2$. These parameters are set to $\gamma = 2$, $\langle d\rangle = 10$, $|V| = 1000$, and $\mu = 0.2$ for LFR-1000 dataset. Details of the datasets used for the experimentation are represented in Table 1. The table include information on the number of nodes, the number of edges, the maximum degree, average degree, and assortativity. Assortativity of a graph is the correlation between node degrees; in assortative networks, nodes with high (low) degrees tend to connect to those with high (low) degrees, whereas in disassortative networks, node with high (low) degrees tend to connect to those with low (high) degrees. Networks without any specific pattern between degrees of connected nodes will have assortativity of close to zero.

Please cite this article as: A. Zareie, A. Sheikhahmadi, M. Jalili et al., Finding influential nodes in social networks based on neighborhood correlation coefficient, Knowledge-Based Systems (2020) 105580, https://doi.org/10.1016/j.knosys.2020.105580.

Table 1
Properties of the datasets used in the experiments.

Network	$\|V\|$ (# nodes)	$\|E\|$ (# edges)	Max degree	Average degree	Assortativity
Dolphins	62	159	12	5.129	−0.04359
Copperfield	112	425	49	7.589	−0.12935
LFR-200	200	1,052	16	10.520	0.21883
Netsciense	379	914	34	4.823	−0.08170
C-elegans	453	4,596	639	20.291	−0.22582
LFR-1000	1,000	10,610	98	21.22	−0.07430
EuroRoad	1,174	1,417	10	2.414	0.12668
Chicago	1,467	1,298	12	1.770	−0.50492
Hamsterster	2,426	16,631	273	13.711	0.04740
PowerGrid	4,941	6,594	19	2.669	0.00346
PGP	10,680	24,316	205	4.554	0.23821

4.1. Discrimination capability

In the first experiment, the discrimination capability of the ranking lists produced by different methods is investigated. The monotonicity function [7] is used for this purpose, which is calculated as

$$M(R) = \left[1 - \frac{\sum_{r \in R} |V|_r * (|V|_r - 1)}{|V| * (|V| - 1)}\right]^2 \quad (7)$$

where $|V|_r$ is the number of nodes allocated to rank r of ranking list R, and $|V|$ is the total number of nodes. Function M generates a value in range [0, 1], which will be higher if ranking list R has better discrimination capability.

Table 2 compares the monotonicity values of the ranking lists produced by different algorithms including the proposed ECRM. The results show that ERM and ECRM have better discrimination capability than others, while Ks has the worst discrimination capability.

4.2. Ranking accuracy

In this section, the accuracy of different algorithms in estimating influence of nodes is investigated. To this end, the spreading process is first modeled using a spreading model, and the ground-truth ranking of nodal influence is extracted. We then correlate this ground-truth influence ranking with the ranking measure produced by each algorithm. The higher is the correlation of a ranking algorithm, the more accurate is the algorithm. A number of models have been proposed in the literature to simulate spreading process in social networks. These model are divided into three general classes: cascading models [49], threshold models, and epidemic models [50]. Susceptible–Infected–Recovered (SIR) [51] is one of the most popular epidemic spreading models, which has been frequently used for this purpose in the literature [7,13–15,29]. Because of its simplicity, wide acceptance, and ability to properly simulate epidemic process in contagious spreading processes [52], it is used here to produce the ground-truth rankings for nodal influence, which is denoted by σ. However, all nodes and edges are considered homogeneous in SIR that can be mentioned as its cons.

Table 3
Epidemic threshold, β_{th}, and infection rate, β, of the datasets.

Network	β_{th}	β
Dolphins	0.147	0.15
Copperfield	0.073	0.1
LFR-200	0.18	0.2
NetScience	0.125	0.15
C-elegans	0.006	0.01
LFR-1000	0.053	0.06
EuroRoad	0.333	0.35
Chicago	0.185	0.2
Hamsterster	0.024	0.03
PowerGrid	0.258	0.3
PGP	0.053	0.1

In SIR model, each node can be in three states: susceptible, infected or removed. Let us put the susceptible nodes in set S, infected nodes in set I, and removed node in set Re. In order to obtain the influence range of node v_i, we first consider set I to be composed on only this node, while other nodes are in set S. In each timestamp, with probability of β a node from I infects its neighbors that are in state S, which are then moved to set I if infected. The node itself then moves to set Re. The process continues until steady-state is obtained. Finally, the number of nodes that are in set Re represents the influence range of node v_i. The process is repeated for all nodes to obtain their influence range. The numerical simulations are performed 1000 times and the averages over these realizations are used for analysis. According to [7], β should be set slightly larger than the epidemic threshold $\beta_{th} = \frac{\langle d \rangle}{\langle d^2 \rangle}$, where $\langle d \rangle$ and $\langle d^2 \rangle$ are the average degrees of the first and second-order neighbors of the graph nodes, respectively. β_{th} and β considered for each datasets are shown in Table 3.

Kendall's Tau correlation coefficient is used to examine the correlation between ranking list R offered by the algorithms measures and the ground-truth ranking list σ. The correlation is calculated as

$$\tau(\sigma, R) = \frac{n_c - n_d}{0.5\,(n)\,(n-1)} \quad (8)$$

where n_c and n_d represent the numbers of concordant and discordant pairs on the two ranking lists, respectively, and n is the size of the ranking vector. Let $(\sigma_1, R_1), (\sigma_2, R_2), \ldots, (\sigma_n, R_n)$ be a set of rank pairs on ranking lists σ and R. Two pairs (x_i, y_i) and (x_j, y_j) are considered concordant if ($x_i > x_j$ and $y_i > y_j$) or ($x_i < x_j$ and $y_i < y_j$), and are considered discordant if ($x_i > x_j$ and $y_i < y_j$) or ($x_i < x_j$ and $y_i > y_j$).

Table 4 shows the correlation between the ground-truth influence rank of nodes and the one produced by the ranking algorithms. As it is seen, ECRM is the top-performer with the best

Table 2
Monotonicity values obtained for the ranking lists produced by different methods.

Network	Ks	CR	MDD	Cnc+	KS-IF	H-Index	CDE	ERM	CN	CLD	LH	LS	ECRM
Dolphins	0.3769	0.9842	0.9041	0.9873	**0.9979**	0.6841	0.9623	**0.9979**	0.9314	0.9758	0.9592	0.9969	**0.9979**
Copperfield	0.5990	0.9974	0.9181	0.9968	0.9977	0.8110	0.9853	**0.9997**	0.9382	0.9971	0.9830	**0.9997**	**0.9997**
LFR-200	0.0034	0.9811	0.7065	0.9503	0.9864	0.2623	0.9706	**0.9999**	0.7286	0.9802	0.8790	0.9994	**0.9999**
NetScience	0.6421	0.9806	0.8215	0.9893	0.9946	0.6825	0.9767	**0.9953**	0.9243	0.9792	0.9513	0.9951	0.9952
C-elegans	0.8413	0.9974	0.9277	0.9984	0.998	0.7311	0.9978	**0.9989**	0.9713	0.9974	0.9962	**0.9989**	**0.9989**
LFR-1000	0.6560	0.9995	0.8687	0.9994	0.9998	0.7275	0.9994	**0.9999**	0.9561	0.9994	0.9925	**0.9999**	**0.9999**
EuroRoad	0.2126	0.8658	0.6498	0.9175	0.9618	0.2534	0.8682	0.9978	0.7040	0.8442	0.7165	0.7502	**0.9979**
Chicago	0.0000	0.4143	0.0536	0.4115	0.6041	0.0034	0.0550	0.8555	0.0535	0.4115	0.2262	0.0530	**0.8942**
Hamsterster	0.8714	0.9849	0.9264	0.9855	0.9855	0.8835	0.9637	0.9848	0.9702	0.9837	0.9768	0.9827	**0.9858**
PowerGrid	0.2460	0.9182	0.6928	0.9420	0.9806	0.3930	0.8370	**0.9999**	0.7708	0.9000	0.8262	0.8726	**0.9999**
PGP	0.4806	0.9695	0.6678	0.9851	0.9906	0.5172	0.7081	**0.9997**	0.8066	0.9641	0.9191	0.6970	**0.9997**

Please cite this article as: A. Zareie, A. Sheikhahmadi, M. Jalili et al., Finding influential nodes in social networks based on neighborhood correlation coefficient, Knowledge-Based Systems (2020) 105580, https://doi.org/10.1016/j.knosys.2020.105580.

Table 4
Correlation between the ranking lists offered by the algorithms including the proposed method ECRM and the ground-truth.

Network	Ks	CR	MDD	Cnc+	KS-IF	H-Index	CDE	ERM	CN	CLD	LH	LS	ECRM
Dolphins	0.5791	0.8334	0.8038	0.8847	0.8445	0.7774	0.8075	0.9196	0.8117	0.7858	0.8736	0.8377	**0.9249**
Copperfield	0.7260	0.8843	0.8399	0.9004	0.8652	0.8269	0.8563	0.9225	0.8468	0.8491	0.9073	0.8546	**0.9273**
LFR-200	0.0470	0.7499	0.6830	0.8451	0.8320	0.4369	0.6975	0.8818	0.6349	0.7573	0.7424	0.5800	**0.8920**
NetScience	0.5018	0.6907	0.5886	0.8263	0.8171	0.5519	0.6159	0.8768	0.6239	0.7906	0.7017	0.7870	**0.9006**
C-elegans	0.6946	0.8356	0.6886	0.8265	0.7216	0.5655	0.7204	0.8419	0.7027	0.7885	**0.8739**	0.8705	0.8338
LFR-1000	0.5929	0.7416	0.5479	0.7792	0.7647	0.5836	0.6078	0.7816	0.5999	0.7568	0.7308	0.7292	**0.7845**
EuroRoad	0.4082	0.7355	0.6015	0.8003	0.8024	0.4289	0.6768	**0.8646**	0.6251	0.7560	0.6614	0.6135	0.8582
Chicago	0.0000	0.5458	0.1669	0.5433	0.5669	0.0550	0.1681	0.5791	0.1663	0.5433	0.2833	0.1657	**0.6074**
Hamsterster	0.6836	0.8078	0.7007	0.8266	0.8245	0.6926	0.7085	0.8339	0.7189	0.8071	0.8071	0.8214	**0.8378**
PowerGrid	0.3359	0.6052	0.5071	0.7147	0.7246	0.4295	0.5447	0.7740	0.5344	0.6823	0.6085	0.5047	**0.7899**
PGP	0.4073	0.5956	0.4361	0.7144	0.6821	0.4157	0.4490	0.7158	0.4907	0.6303	0.6600	0.6798	**0.7316**

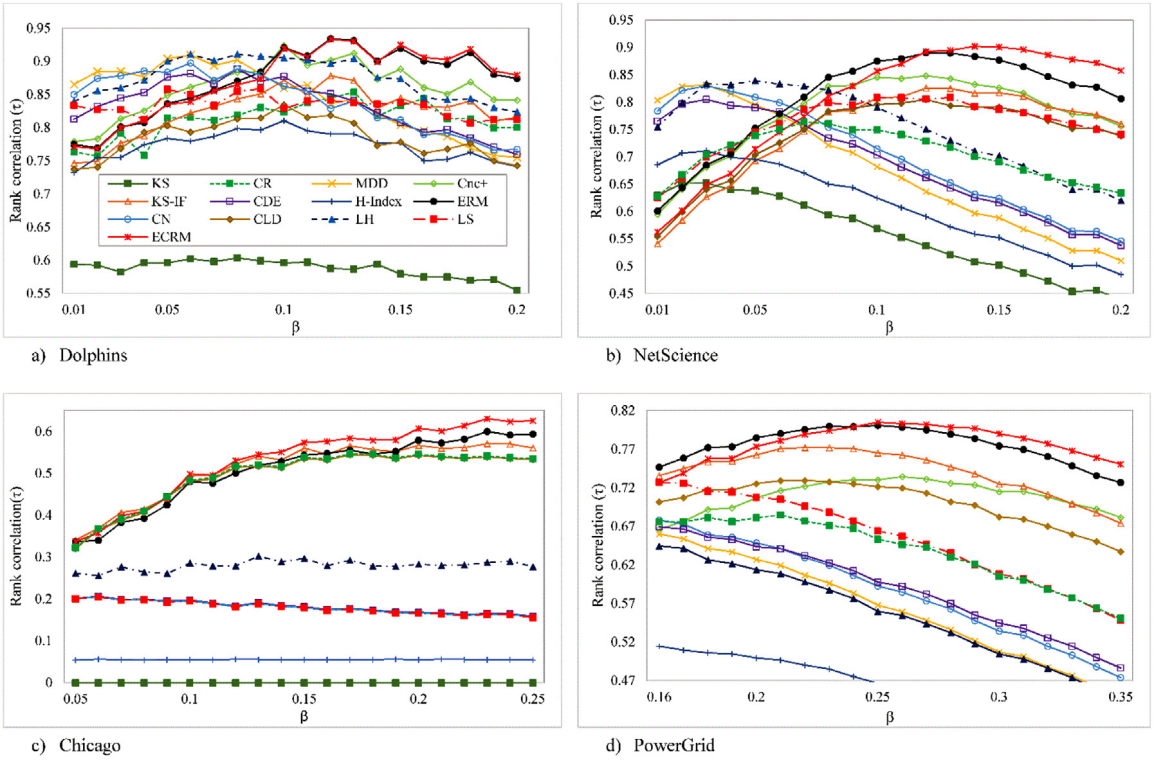

Fig. 2. Effect of changes in the value of β on the accuracy of different methods over four datasets including Dolphins, NetScience, Chicago, and PowerGrid. The accuracy is measured by Kendall correlation between the ground-truth nodal ranking and those obtained by the algorithms.

performance in 9 out of 11 datasets. ERM is the top-performer in EuroRoad dataset, while LH has the best performance in C-elegans dataset. Ks shows the worst performance with the lowest correlation values. These results reveal superiority of the proposed influence estimation method over these state-of-the-art algorithms.

We next study the effect of the infection rate of SIR model (parameter β) on the accuracy of the algorithms. Fig. 2 shows Kendall correlation of the ground-truth ranking and those produced by the ranking algorithms, as a function of the infection rate β. We perform the experiments on four networks including Dolphins, NetScience, Chicago, and PowerGrid. It is clear that the correlations depend on the value of the infection rate for all algorithms. However, the proposed method exhibits higher accuracy than others as the value of β increases, and in particular when it exceeds β_{th}. ERM exhibits close results to those of ECRM. It is worth to mention that as is shown in Table 2, all nodes are assigned in one shell in Chicago dataset, and thus most of the methods applying k-shell approach to determine nodes' influentially, has accordingly lower accuracy in this dataset.

Often, one would like to find only the top-ranked nodes instead of obtaining a ranking list that include all nodes. Therefore in another experiment, the accuracy of the proposed algorithm is compared with others when only a portion of top-ranked nodes are considered. To this end, the similarity between the ranking list R offered by the algorithms and the real ranking list σ obtained based on SIR model, is examined using Jaccard similarity coefficient [53]. We consider top-T nodes of the ranking list and vary T. Jaccard similarity coefficient J is obtained as

$$J(T) = \frac{|\sigma(T) \cap R(T)|}{|\sigma(T) \cup R(T)|} \tag{9}$$

where $\sigma(T)$ and $R(T)$ are vectors containing the top-T ranked nodes of σ and R, respectively.

Fig. 3 shows Jaccard similarity coefficient as a function of T on four datasets including LFR-200, NetScience, Chicago, and EuroRoad. As it is seen, ECRM is again the top-performer in all datasets and across almost all values of T. ERM results in close performance as ECRM, often much better than other algorithms. Influential nodes often have high degrees and are located in dense parts of the graphs. On the other hand, it is likely that commonalities of the friends between the neighbors of these nodes is higher due to being locating in dense parts of the graphs. The proposed

Please cite this article as: A. Zareie, A. Sheikhahmadi, M. Jalili et al., Finding influential nodes in social networks based on neighborhood correlation coefficient, Knowledge-Based Systems (2020) 105580, https://doi.org/10.1016/j.knosys.2020.105580.

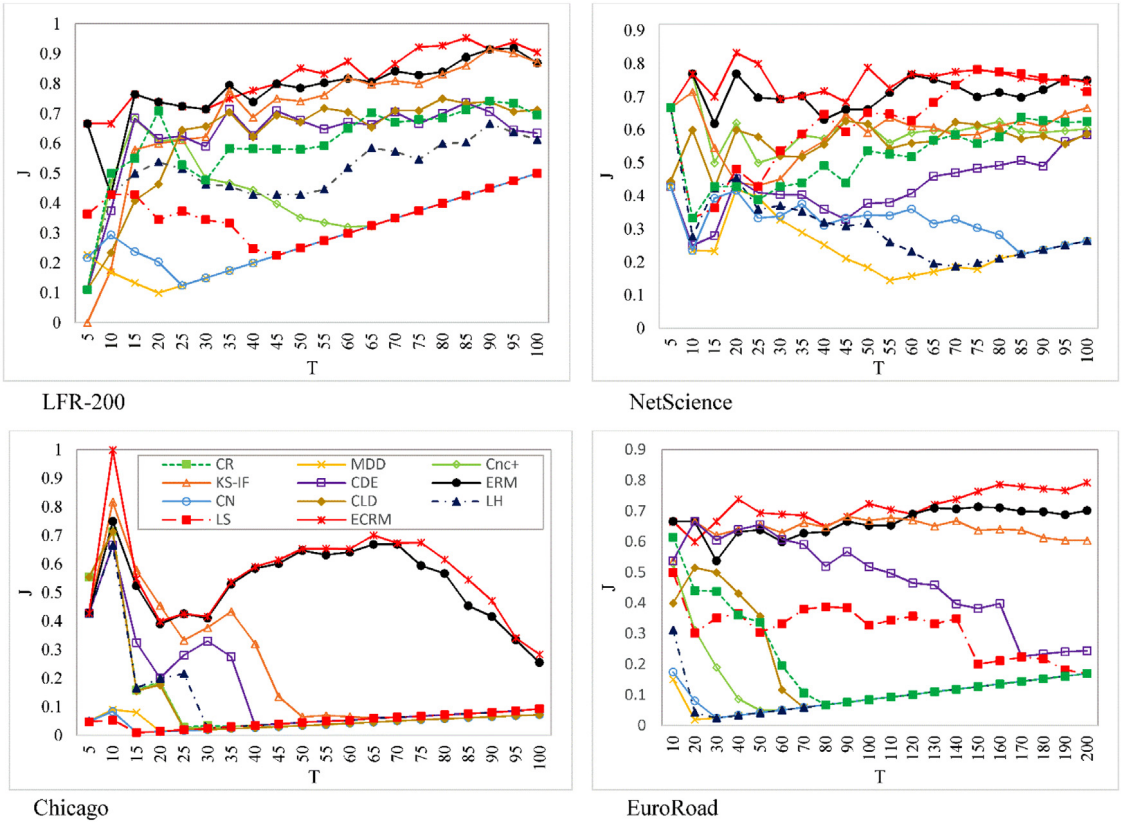

Fig. 3. Jaccard similarity coefficient J between the ground-truth ranking and those obtained by the algorithms, when only top-T ranked nodes are considered from the list.

method considers these commonalities and try to distinguish the high-degree nodes according to the commonalities.

We next study how varying parameters of the LFR model affects the performance of algorithms. We fix the parameters of the model as $|V| = 1000$, $\mu = 0.2$, and $\gamma = 2$, and the average degree $\langle d \rangle$ varied from 5 to 15. The proposed method has the best performance for almost all the values of these parameters, followed by ERM (Fig. 4). It can be seen from Fig. 4(a) that accuracy of the algorithms first increases by increasing the average degree, but if the average degree further increases from a certain threshold (which is different for different algorithms), the accuracy declines. In the second experiment, by fixing $|V| = 1000$, $\langle d \rangle = 7$, and $\mu = 0.2$, γ is varied from 2 to 3. Generally, the accuracy declines by increasing γ (Fig. 4(b)). The main reason for this profile is that by increasing γ, heterogeneity of degree distribution increases, resulting in fewer high-degree nodes.

Impact of varying the mixing parameter of the community structure μ is investigated in the third experiment by setting $|V|$, $\langle d \rangle$, and γ as 1000, 7, and 2, respectively, and varying the value of μ from 0.2 to 0.7. The algorithms have different profiles against μ; while ECRM and ERM show declined accuracy as μ increases, accuracy of others improves (Fig. 4(c)). However, ECRM is still the top-performer followed by ERM.

In summary, the Experiments show that the proposed method has higher discrimination capability than the other algorithms. Although ERM provides close results to ECRM, the latter takes into account more accurate structural information, which is the main reason for its better performance. ECRM considers commonality between the nodes and their neighbors as a criterion to estimate their influentiality. Indeed, as commonality between nodes is reduced (or minimized on optimal cases), it is likely that one can reach to further parts of the network by the same effort, and thus obtaining better influentiality.

5. Conclusion

Social networks analysis and mining have recently gained ever-increasing importance with many potential applications in diverse industries. Influence maximization is one of the topics that has attracted much attention in this field. An important challenge in influence maximization is to find the most influential nodes based on their structural location in the network. In this manuscript, we proposed a new method to estimate influence of nodes in information networks. The proposed method is based on local clustering coefficient and uses similarity of connections between neighboring nodes. The method is based on k-shell decomposition approach where influentiality of a node depends on how the node has shared connections with its neighbors. Two nodes having neighbors within the same parts of the networks are considered as correlated nodes in the proposed method. Correlated nodes often have limited spreading capabilities, as they together span smaller parts of the network that uncorrelated ones. Therefore, a node having low correlation with its neighbors could be better candidate to be an influential node due to its ability in spreading messages to different parts of the network by the help of its neighbors. We considered a number of real and synthetic networks and compared the performance of the proposed method with a number of state-of-the-art algorithms. Our numerical simulation results showed that the proposed algorithm has better performance than the others in various settings.

Acknowledgment

Mahdi Jalili is supported by Australian Research Council through project No. DP170102303.

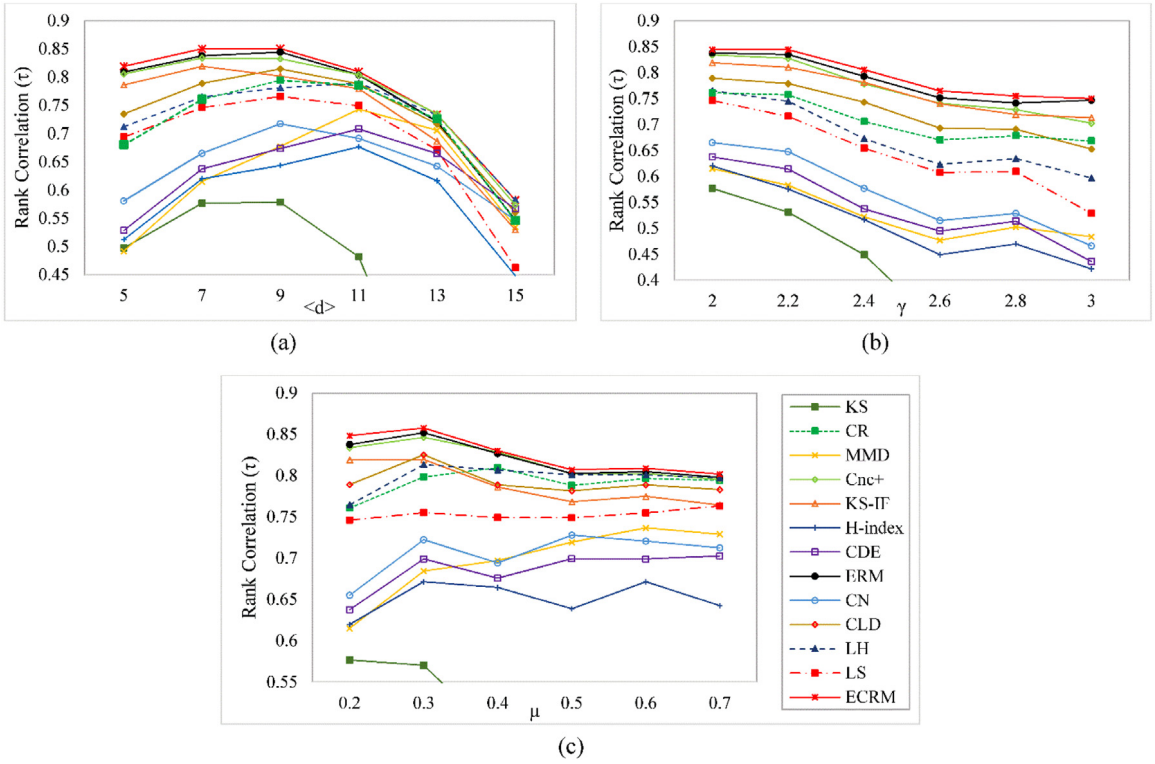

Fig. 4. The effect of parameters of the LFR model on the accuracy of the algorithms.

References

[1] W. Chen, C. Wang, Y. Wang, Scalable influence maximization for prevalent viral marketing in large-scale social networks, in: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2010, pp. 1029–1038.
[2] A. Sheikhahmadi, M.A. Nematbakhsh, A. Shokrollahi, Improving detection of influential nodes in complex networks, Physica A 436 (2015) 833–845.
[3] R.M. Bond, et al., A 61-million-person experiment in social influence and political mobilization, Nature 489 (7415) (2012) 295.
[4] A. Sheikhahmadi, M.A. Nematbakhsh, A. Zareie, Identification of influential users by neighbors in online social networks, Physica A 486 (2017) 517–534.
[5] M.Y. Cheung, C. Luo, C.L. Sia, H. Chen, Credibility of electronic word-of-mouth: Informational and normative determinants of on-line consumer recommendations, Int. J. Electron. Commer. 13 (4) (2009) 9–38.
[6] A. Zareie, A. Sheikhahmadi, K. Khamforoosh, Influence maximization in social networks based on TOPSIS, Expert Syst. Appl. 108 (2018) 96–107.
[7] J. Bae, S. Kim, Identifying and ranking influential spreaders in complex networks by neighborhood coreness, Physica A 395 (2014) 549–559.
[8] A. Sheikhahmadi, M.A. Nematbakhsh, Identification of multi-spreader users in social networks for viral marketing, J. Inf. Sci. 43 (3) (2017) 412–423.
[9] Z. Wang, C. Du, J. Fan, Y. Xing, Ranking influential nodes in social networks based on node position and neighborhood, Neurocomputing 260 (2017) 466–477.
[10] A. Zareie, A. Sheikhahmadi, M. Jalili, Identification of influential users in social networks based on users' interest, Inform. Sci. 493 (2019) 217–231.
[11] L.C. Freeman, A set of measures of centrality based on betweenness, Sociometry (1977) 35–41.
[12] M. Kitsak, et al., Identification of influential spreaders in complex networks, Nat. Phys. 6 (11) (2010) 888.
[13] Z. Wang, Y. Zhao, J. Xi, C. Du, Fast ranking influential nodes in complex networks using a k-shell iteration factor, Physica A 461 (2016) 171–181.
[14] A. Zareie, A. Sheikhahmadi, A hierarchical approach for influential node ranking in complex social networks, Expert Syst. Appl. 93 (2018) 200–211.
[15] A. Zareie, A. Sheikhahmadi, A. Fatemi, Influential nodes ranking in complex networks: An entropy-based approach, Chaos Solitons Fractals 104 (2017) 485–494.
[16] A. Zareie, A. Sheikhahmadi, EHC: Extended H-index centrality measure for identification of users' spreading influence in complex networks, Physica A 514 (2019) 141–155.
[17] A. Zareie, A. Sheikhahmadi, M. Jalili, Influential node ranking in social networks based on neighborhood diversity, Future Gener. Comput. Syst. 94 (2018) 120–129.
[18] L.C. Freeman, Centrality in social networks conceptual clarification, Soc. Netw. 1 (3) (1978) 215–239.
[19] G. Sabidussi, The centrality index of a graph, Psychometrika 31 (4) (1966) 581–603.
[20] D. Chen, L. Lü, M.-S. Shang, Y.-C. Zhang, T. Zhou, Identifying influential nodes in complex networks, Physica A 391 (4) (2012) 1777–1787.
[21] S. Gao, J. Ma, Z. Chen, G. Wang, C. Xing, Ranking the spreading ability of nodes in complex networks based on local structure, Physica A 403 (2014) 130–147.
[22] D.-B. Chen, H. Gao, L. Lü, T. Zhou, Identifying influential nodes in large-scale directed networks: the role of clustering, PLoS One 8 (10) (2013) e77455.
[23] T. Petermann, P. De Los Rios, Role of clustering and gridlike ordering in epidemic spreading, Phys. Rev. E 69 (6) (2004) 066116.
[24] T. Zhou, G. Yan, B.-H. Wang, Maximal planar networks with large clustering coefficient and power-law degree distribution, Phys. Rev. E 71 (4) (2005) 046141.
[25] L. Lü, T. Zhou, Q.-M. Zhang, H.E. Stanley, The H-index of a network node and its relation to degree and coreness, Nature Commun. 7 (2016) 10168.
[26] Q. Liu, et al., Leveraging local h-index to identify and rank influential spreaders in networks, Physica A 512 (2018) 379–391.
[27] O. AskariSichani, M. Jalili, Influence maximization of informed agents in social networks, Appl. Math. Comput. 254 (2015) 229–239.
[28] A. Zeng, C.-J. Zhang, Ranking spreaders by decomposing complex networks, Phys. Lett. A 377 (14) (2013) 1031–1035.
[29] C. Li, L. Wang, S. Sun, C. Xia, Identification of influential spreaders based on classified neighbors in real-world complex networks, Appl. Math. Comput. 320 (2018) 512–523.
[30] Y.-P. Wan, J. Wang, D.-G. Zhang, H.-Y. Dong, Q.-H. Ren, Ranking the spreading capability of nodes in complex networks based on link significance, Physica A 503 (2018) 929–937.
[31] S. Yu, L. Gao, L. Xu, Z.-Y. Gao, Identifying influential spreaders based on indirect spreading in neighborhood, Physica A 523 (2019) 418–425.
[32] Z. Dai, P. Li, Y. Chen, K. Zhang, J. Zhang, Influential node ranking via randomized spanning trees, Physica A 526 (2019) 120625.
[33] T. Wen, W. Jiang, Identifying influential nodes based on fuzzy local dimension in complex networks, Chaos Solitons Fractals 119 (2019) 332–342.
[34] Z. Lv, N. Zhao, F. Xiong, N. Chen, A novel measure of identifying influential nodes in complex networks, Physica A 523 (2019) 488–497.
[35] H. Mo, Y. Deng, Identifying node importance based on evidence theory in complex networks, Physica A (2019) 121538.

Please cite this article as: A. Zareie, A. Sheikhahmadi, M. Jalili et al., Finding influential nodes in social networks based on neighborhood correlation coefficient, Knowledge-Based Systems (2020) 105580, https://doi.org/10.1016/j.knosys.2020.105580.

[36] Y. Yang, L. Yu, X. Wang, Z. Zhou, Y. Chen, T. Kou, A novel method to evaluate node importance in complex networks, Physica A 526 (2019) 121118.
[37] Y. Yang, L. Yu, Z. Zhou, Y. Chen, T. Kou, Node importance ranking in complex networks based on multicriteria decision making, Math. Probl. Eng. 2019 (2019).
[38] M. Li, R. Zhang, R. Hu, F. Yang, Y. Yao, Y. Yuan, Identifying and ranking influential spreaders in complex networks by combining a local-degree sum and the clustering coefficient, Internat. J. Modern Phys. B 32 (06) (2018) 1850118.
[39] Y. Liu, M. Tang, T. Zhou, Y. Do, Identify influential spreaders in complex networks, the role of neighborhood, Physica A 452 (2016) 289–298.
[40] D. Lusseau, K. Schneider, O.J. Boisseau, P. Haase, E. Slooten, S.M. Dawson, The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations, Behav. Ecol. Sociobiol. 54 (4) (2003) 396–405.
[41] M.E. Newman, Finding community structure in networks using the eigenvectors of matrices, Phys. Rev. E 74 (3) (2006) 036104.
[42] J. Duch, A. Arenas, Community detection in complex networks using extremal optimization, Phys. Rev. E 72 (2) (2005) 027104.
[43] L. Šubelj, M. Bajec, Robust network community detection using balanced propagation, Eur. Phys. J. B 81 (3) (2011) 353–362.
[44] R. Eash, K. Chon, Y. Lee, D. Boyce, Equilibrium traffic assignment on an aggregated highway network for sketch planning, Transp. Res. 13 (1979) 243–257.
[45] J. Kunegis, Konect: the koblenz network collection, in: Proceedings of the 22nd International Conference on World Wide Web, ACM, 2013, pp. 1343–1350.
[46] D.J. Watts, S.H. Strogatz, Collective dynamics of 'small-world'networks, Nature 393 (6684) (1998) 440.
[47] M. Boguñá, R. Pastor-Satorras, A. Díaz-Guilera, A. Arenas, Models of social networks based on social distance attachment, Phys. Rev. E 70 (5) (2004) 056122.
[48] A. Lancichinetti, S. Fortunato, Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities, Phys. Rev. E 80 (1) (2009) 016118.
[49] M. Jalili, M. Perc, Information cascades in complex networks, J. Complex Netw. 5 (5) (2017) 665–693.
[50] A. Buscarino, L. Fortuna, M. Frasca, V. Latora, Disease spreading in populations of moving agents, Europhys. Lett. 82 (3) (2008) 38002.
[51] R. Pastor-Satorras, A. Vespignani, Epidemic dynamics and endemic states in complex networks, Phys. Rev. E 63 (6) (2001) 066117.
[52] Y.-H. Fu, C.-Y. Huang, C.-T. Sun, Using global diversity and local topology features to identify influential network spreaders, Physica A 433 (2015) 344–355.
[53] L. Hébert-Dufresne, A. Allard, J.-G. Young, L.J. Dubé, Global efficiency of local immunization on complex networks, Sci. Rep. 3 (2013) 2171.

Please cite this article as: A. Zareie, A. Sheikhahmadi, M. Jalili et al., Finding influential nodes in social networks based on neighborhood correlation coefficient, Knowledge-Based Systems (2020) 105580, https://doi.org/10.1016/j.knosys.2020.105580.