Entropy and Structural-Hole Based Node Ranking Methods

Chinenye Ezeh, Tao Ren, Yan-Jie Xu, Shi-Xiang Sun, Zhe Li

College of Information Science and Engineering, Northeastern University, China
noblenenye@gmail.com, chinarentao@163.com, 1285906739@qq.com, sunshixiangp@163.com, gislzneu@163.com

Abstract

Several research works had been carried out to discover suitable algorithms to quantify node centralities. Among the many existing centrality metrics, only few consider centrality at the sub-graph level or deal with structural hole capabilities of pivot nodes. Research has proven the importance of sub-graph information in distinguishing influential nodes. In this work, two centrality metrics are proposed to distinguish and rank nodes in complex networks. The first metric called Sub-graph Degree Information centrality is based on entropy quantification of a node's sub-graph degree distribution to determine its influence. The second metric called Sub-graph Degree and Structural Hole centrality considers a node's sub-graph degree distribution and its structural hole property. The two metrics are designed to efficiently support weighted and unweighted networks. Performance evaluations were done on five real world datasets and one artificial network. The proposed metrics were equally compared against some classic centrality metrics. The results show that the proposed metrics can accurately distinguish and rank nodes distinctly on complex networks. They can equally discover highly influential and spreader nodes capable of causing epidemic spread and maximum network damage.

Keywords: Sub-graph degree, Entropy, Node ranking, Structural hole, Influential node

1 Introduction

Over the years, several research works had been carried out to discover appropriate metrics to quantify node centralities in order to find the most influential nodes in complex networks [1-3]. Out of the several existing centrality metrics, just a few of them considered centralities at the sub-graph level as they are mostly designed with local or global network information. Additionally, these metrics cannot efficiently measure influence-based proximity [3] such as structural holes property of a node which it exerts between two of its unconnected neighbours.

In the literature, Closeness and Betweeness centralities are some of the most popular centrality metrics. But then, Estrada and Rodriguez-Velazquez [1] opined that these centrality methods are rather non-suitable measures of sub-graph centrality at the sub-graph level of a network. They equally pointed out that sub-graphs are very significant in real networks. To further explore the usefulness of a node's sub-graph information, Ref. [2] proposed an entropy based Sub-graph Degree Centrality that determines a node's total influence by considering its direct influence on its 1-hop neighbours and its indirect influence on its 2–hop neighbours. It has been revealed that apart from the topological connections a node has with its 1–hop neighbours and its indirect connection with its 2–hop neighbours, it could still play a very powerful role in bridging the communication among all of its 1-hop neighbours who have no direct link to each other.

In line with this thought, a hub node with so many connections which at the same time enjoys a brokerage position over its neighbours, may be strategically positioned not just as a hub but also as a bridge to its direct neighbours that have no direct connections to each other. This idea had been extensively explored by Ronald Burt in his seminal papers on "Structural Holes" [4]. A node is termed a structural hole, if it controls the communication between two or more of its unconnected neighbours. Such a node can benefit significantly by acting as a bridge to many of its direct neighbours who have no connection with each other [5, 6]. People benefit from acting as bridges between groups of people who do not otherwise interact [7]. In essence, structural-hole-rich networks provide informational benefits. In other words, a node that has many indirect ties is privy to more resources than a node with a limited reach in the network [5].

Bearing in mind that more information can be revealed about a node at its sub-graph level with respect to the points raised in the preceding paragraph, we first propose the Sub-graph Degree Information centrality (SDI). The degree distribution of a node's sub-graph which involves its 1–hop neighbours, is computed and thereafter its influence is quantified by Dehmer's entropy model. We also propose the Sub-graph Degree and Structural Hole (SDSH) metric which computes the degree distribution of a node at the

*Corresponding Author: Tao Ren; E-mail: chinarentao@163.com
DOI: 10.53106/160792642021092205007

sub-graph level involving its 1–hop neighbours as well as its structural hole capability among them. The two proposed algorithms are designed with the flexibility to support weighted and unweighted networks. In other words, the proposed metrics automatically switch between the options of considering or ignoring weights on networks as the situation demands. The proposed metrics were tested on five real-world datasets and one artificial network. They were equally compared against some select centrality metrics and, in some instances, their performance show that they can discover highly influential nodes that are epidemic spreaders which are capable of initiating rapid disease spread and can as well cause maximum network destruction.

In essence, the contributions of our work are summarized below:

- Two novel centrality metrics based on entropy and structural-hole are proposed.
- Sub-graph degree distribution of nodes is explored and their influence quantified by entropy model.
- Structural hole is exploited in combination with sub-graph degree distribution to detect highly influential spreader nodes.
- Both metrics are strategically designed to support weighted and unweighted networks.

The rest of this work is organized as follows: in section 2, literature review of some related works is presented. In section 3, the models of the proposed metrics and their algorithms are described. In section 4, we conduct experiments and performance evaluations of the proposed centrality metrics. Also, the experimental results are discussed and analyzed. Finally, recommendations for improvement and future research are offered and the paper concluded in section 5.

2 Related Works

The importance of sub-graph information of nodes can never be overemphasised. Estrada and Rodriguez-Velazquez observed that sub-graphs are very significant in real networks. Hence, they proposed the Sub–graph centrality metric (SC) in which each node partakes in the sub-graph of a network [1]. It is noted that SC discriminates nodes much better than some classic centrality methods [8]. This notwithstanding, the computational complexity of this algorithm is quite high.

Ref. [9] proposed a new node influence metric known as H-index and established its relation to Degree and Coreness centralities. The H–index centrality has had tremendous success with a number of variants that are designed for unweighted networks mostly. But we know that real world networks have weights which could add more meaningfulness in the detection of influential nodes on networks.

Ref. [10] proposed a semi-local centrality metric to find influential nodes. They took advantage of Degree centrality's low characteristics and combined it with the high computational complexities of Betweeness and Closeness centralities to achieve the Semi Local centrality. The performance of this metric in detecting influential nodes is quite reasonable yet, it does not support weighted networks nor can it be applied on star networks [11].

Ref. [2] proposed an Entropy based centrality model by disintegrating a graph into sub-graphs. This enabled them to compute the entropy of the sub-graph degree distribution of a node inspired by Shannon's information theory on Entropy as formulated by Mathias Dehmer. Using entropy theory, they quantified the local influence of a node on its neigbhours and an indirect influence on its 2-hop neigbhours. Nonetheless, this model is specifically designed for undirected and unweighted networks. Yang and An [6] proposed Degree and Structural Hole Count algorithm to detect critical nodes. They leveraged the advantage provided by structural holes to design the model such that a node that has high degree with large number of neighbours who are not connected to each other, would be highly influential. The algorithm is designed for only unweighted networks.

We modify the models proposed by Refs. [2] and [6] to come up with two centrality algorithms. First, we propose the Sub-graph Degree Information centrality. This metric computes the sub-graph degree information of a node and thereafter, its influence is quantified by computing the entropy of the sub-graph degree information using Dehmer's entropy model. Second, the sub-graph degree and structural hole centrality is proposed. For every node, its sub-graph degree and those of its neighbours are computed in combination with its structural hole property. The two metrics are designed with the flexibility to support weighted and unweighted networks which gives them an edge over existing methods. Performance evaluations were done on five real-world datasets and one artificial network against some classic centrality methods. Results show that the proposed methods are very efficient in detecting influential nodes capable of causing maximum network destruction or rapid epidemic spread.

3 Methodology

A typical weighted and undirected graph G is given as a set of vertices, edges and weights $G(V, E, W)$ where, $V = v_i, i = 1, 2, 3, ..., N$ which represent the set of nodes and E represents the set of edges connecting them, whereas W is the weight set of E. An edge e_{ij} is given by v_{ij} where $v_i \in V$ with a weight $w_i \in W$. An edge $e_{ij} = 1$ if v_i and v_j are connected else $e_{ij} = 0$ in an unweighted network. In a given weighted network,

w_{ij} is a weighted adjacency matrix that shows the weight of connections between nodes v_i and v_j. A self-connection $v_{ii} = 0$ and $v_{ij} = v_{ji}$. The sub-graph of node v_i is made up of first order neighborhood set N_i^1 of node v_i inclusive of itself.

3.1 Sub-graph Degree Information Centrality

In order to ascertain the influence of nodes on a network, one needs to quantify the structural information of such network. To achieve this, we assume that in a given sub-graph, the probability of node v_i to receive resources [12] is given by:

$$|\frac{dv_i'}{n'}| \tag{1}$$

where dv_i' is the degree of node v_i in the sub-graph and $|n'|$ is the cardinality of the number of nodes in that sub-graph. Since we are dealing with a symmetric network, we assume that a focal node can send and receive resources to its immediate neighbours. The strength of node connections is also captured alongside its topological connections. To achieve this, we consider the average weights incident on the focal node [13]. The model is given as:

$$(\frac{s_i'}{dv_i'})^\alpha \tag{2}$$

$S_i' = \Sigma w_{ij}$ is the sum of weights incident on the focal node and $\alpha = [0,1]$ is a tuning parameter. To compute the degree distribution of a node using its sub-graph information content, we propose the Sub-graph Degree (*SD*) model and Sub-graph Degree Information (*SDI*) centrality in equations 3 and 4 respectively:

$$SD = \frac{dv_i'}{|n'|} \times (\frac{S_i'}{dv_i'})^\alpha = \frac{dv_i'^{(1-\alpha)} \times S_i'^{(\alpha)}}{|n'|} \tag{3}$$

Therefore,

$$SD_i = \sum_{j=1}^{|n'|} \frac{dv_i'^{(1-\alpha)} \times S_i'^{(\alpha)}}{|n'|} \tag{4}$$

where j is node v_i's neighbors and $|n'|$ is the total number of nodes in node v_i's sub-graph. In other words, when

$$\alpha = \begin{cases} 0, SD = \dfrac{dv_i'}{|n'|} \\ 1, SD = \dfrac{S_i'}{|n'|} \end{cases}$$

α is a tuning parameter used to control the level of importance given to topological or strength of connections between nodes. It is equally possible to determine the values of α automatically from the network information using entropy weighting method [14]. In this case, the control parameter α is replaced by α_1 and α_2. Therefore, α_1 controls the topological connections of nodes, while α_2 controls the strength of their connections.

3.2 Highlights of Information Entropy

Shannon's information entropy of an assumed stochastic variable A is given by:

$$H(A) = -\sum_{i=1}^{n} P_i \log_2 P_i \tag{5}$$

Matthias Dehmer [15] obtained the probability distribution of a graph from the information functional of such graph. He described and represented the measured structural information of a graph as the resultant graph entropy. i.e., $P = P_1, P_2, P_3, ..., P_n$. Ref. [2] then used this probability distribution to define the relationship between any given node and its neigbhours at the subgraph level as:

$$P_i = \frac{\lambda_i}{\sum_{j=1}^{n} \lambda_j}, \ j = 1, 2, 3, ..., n \tag{6}$$

where λ represents the i_{th} non-negative integer. Therefore, entropy $H(A)$ becomes:

$$H(A) = -\sum_{i=1}^{n} P_i \log_b P_i = (\log_b \sum_{j=1}^{n} \lambda_i) - \sum_{j=1}^{n} \frac{\lambda_i}{\sum_{j=1}^{n} \lambda_i} \log_b \lambda_i \tag{7}$$

3.3 Influence Quantification of Nodes

For any given graph, the influence of each node can effectively be quantified with the following equation:

$$I_i = \log_{10}(\sum_{j=1}^{n'} SDI_i) - \sum_{j=1}^{n'} \frac{SD_i}{\sum_{j=1}^{n'} SDI_i} \log_{10} SD_i \tag{8}$$

Note that in all computations made in this work, the logarithm base $b = 10$ is used. The pseudocode to compute a node's sub-graph degree information and influence is presented in Algorithm 1.

Algorithm 1. Sub-graph degree information and influence computation

Input: $G(V, E, W)$

Output: SDI_i, I_i

1. if G.is_weighted == 0 then
2. $\quad\alpha_1 = 1$ and $\alpha_2 = 0$
3. else
4. $\quad$*compute* α_1 and α_2
5. end if

for $i = 1 : len|G|$ do
    $SubG_i = G'_i$ #Extract the sub-graph
    #details of node v_i
    for $ii = 1 : |SubG_i|$ do
        determine degree(*ii*); weights(*ii*)
        compute SDI_i #use equation 3
    end for
    compute SDI_i #use equation 4
    compute I_i #use equation 8
end for

3.4 Sub-graph Degree and Structural-hole Centrality

In their work, Yang and An [6] proposed the Degree and Structural Hole Count (DSHC) algorithm. The model is given as:

$$DSHC_i = \sum_{j \in \Gamma_i} \left(\left(\frac{1}{dv_i} + \frac{1}{dv_j} \right) \times \frac{1}{1 + \Delta_{ij}} \right)^2 \quad \textbf{(9)}$$

where dv_i is the degree of node v_i, dv_j represents the degree of its neighbours and Δ_{ij} is the number of structural holes that exist in node v_i which originates from its neighbours.

In view of the assumption in section 3.1, in any given sub-graph, a node can send or receive resources to any of its direct neighbours and at the same time function as the bridge connecting all of its neighbours who are not connected to each other. What this means is that unconnected neighbours of node v_i can still pass resources to each other through node v_i. This arrangement gives node v_i much influence over others. We adapt equation 3 into equation 9 to obtain the Sub-graph Degree and Structural Hole (SDSH) model:

$$SDSH_i = \sum_{j \in \Gamma_i} ((SD_i + SD_j) \times (1 + \eta_{ij})) \quad \textbf{(10)}$$

where SD_i is the sub-graph degree of a focal node v_i, SD_j represents the sub-graph degree of node v_i's neighbour and Γ_i is the set of all node v_i's neighbours. η_{ij} is the number of structural holes that exist in node v_i which originates from its neighbours. The higher the value of SDSH, the more important node v_i.

The pseudocode to compute SDSH is presented in Algorithm 2.

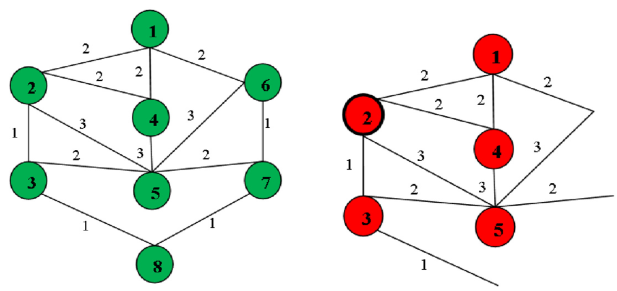
Algorithm 2. Sub-graph degree and Structural-hole algorithm

Input: $G(V, E, W)$
Output: *SDSH*

if *G*.is_weighted == 0 then
    $\alpha_1 = 1$ and $\alpha_2 = 0$
else
    compute α_1 and α_2
end if
for $i = 1 : len|G|$ do
    $SubG_i = G'_i$ #Extract the sub-graph
    #details of node v_i
    for $ii = 1 : |SubG_i|$ do
        $\eta_{ij} = 0$
        for $jj = 1 : reversed|SubG_i|$ do
            find_path(*ii*, *jj*)
            if *jj* not in G.neighbours(*ii*) then
                $\eta_{ij} += 1$
            end if
        end for
        compute SD_i, SD_j #use equation 3
    end for
    compute $SDSH_i$ #use equation 9
end for

An example of how to compute the sub-graph degree using node 2's sub-graph shown in Figure 1(b) is given in Table 1.

Computational example of *SDI* centrality



(a) Toy graph (b) Sub-graph of node 2

Figure 1. Example toy network. Picture adapted from [2]

$$SDI_2 = \sum_{i=1}^{5} SD_2 + SD_1 + SD_3 + SD_4 + SD_5 = 4.8583$$

Node 2's influence can be determined with equation 8.

Table 1. Computation of sub-graph degree

Node ID	$\dfrac{dv_i'^{(\alpha_1)} \times S_i'^{(\alpha_2)}}{\lvert n' \rvert}$	SD_i
2	$\dfrac{4^{0.5916} \times 8^{0.4084}}{5}$	1.0618
1	$\dfrac{6^{0.5916} \times 6^{0.4084}}{5}$	0.7963
3	$\dfrac{3^{0.5916} \times 4^{0.4084}}{5}$	0.6748
4	$\dfrac{3^{0.5916} \times 7^{0.4084}}{5}$	0.8481
5	$\dfrac{5^{0.5916} \times 13^{0.4084}}{5}$	1.4473

$$I_2 = 4.8583 - \sum_{i=1}^{5}[(\frac{1.0618}{4.8583} \times \log_{10} 1.0618) + (\frac{0.7963}{4.8583} \times \log_{10} 0.7963) + (\frac{0.6748}{4.8583} \times \log_{10} 0.6748) + (\frac{0.8481}{4.8583} \times \log_{10} 0.8481) + (\frac{1.4773}{4.8583} \times \log_{10} 1.4773)]$$

$$I_2 = 0.6817$$

In Figure 2, the concept of structural hole is demonstrated. The number of structural holes that exist between node A and node Ego is equal to 2 indicated by the dotted lines. Notice that node B is not counted since node A has a direct connection to it. In like manner, the number of structural holes that exist between node C and node Ego is 3.

Computational example of SDSH centrality

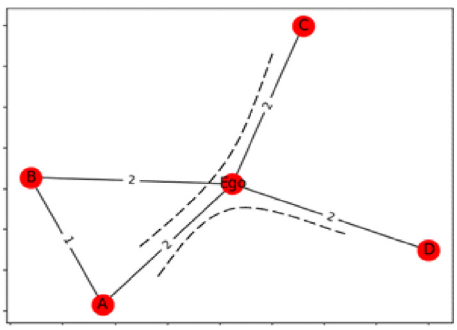


Figure 2. An Ego network Picture adapted from [6]

To compute the sub-graph degree and structural hole centrality of node 2, we go back to the sub-graph degree details of node 2 in Table 1.

$$SDSH_2 = \sum_{i=1}^{5}((SD_2 + SD_1) \times (1+2)) + ((SD_2 + SD_3) \times (1+2)) + ((SD_2 + SD_4) \times (1+1)) + ((SD_2 + SD_5) \times (1+1))$$

$$SDSH_2 = \sum_{i=1}^{5} 5.5743 + 5.2097 + 3.8197 + 5.0782 = 19.6819$$

4 Results and Discussion

All experiments were implemented with Python 3.7.4. The codes available on GitHub were run on a computer with Windows 10 Operating System (64 bits), Intel (R) Core^{TM} *i*3-2310M CPU @ 2.10GHz and 4GB RAM.

4.1 Tests on Network Datasets

Five real-world datasets and one artificial (Barabasi-Albert (BA-net)) network were used for the experimental evaluations. Self-looped edges as well as isolated nodes were deleted from the five real-world network datasets before use. Also, directed networks were converted to undirected networks. The BA-net was built with 10,000 nodes with the connection to two new nodes for every new connection. Details of the datasets are presented in Table 2. n is number of nodes, m is number of edges, $< k >$ is average degree, C is clustering coefficient, $< d >$ is average path length, R is assortavity and β_{th} is the propagation threshold of each network.

4.2 SIR Model

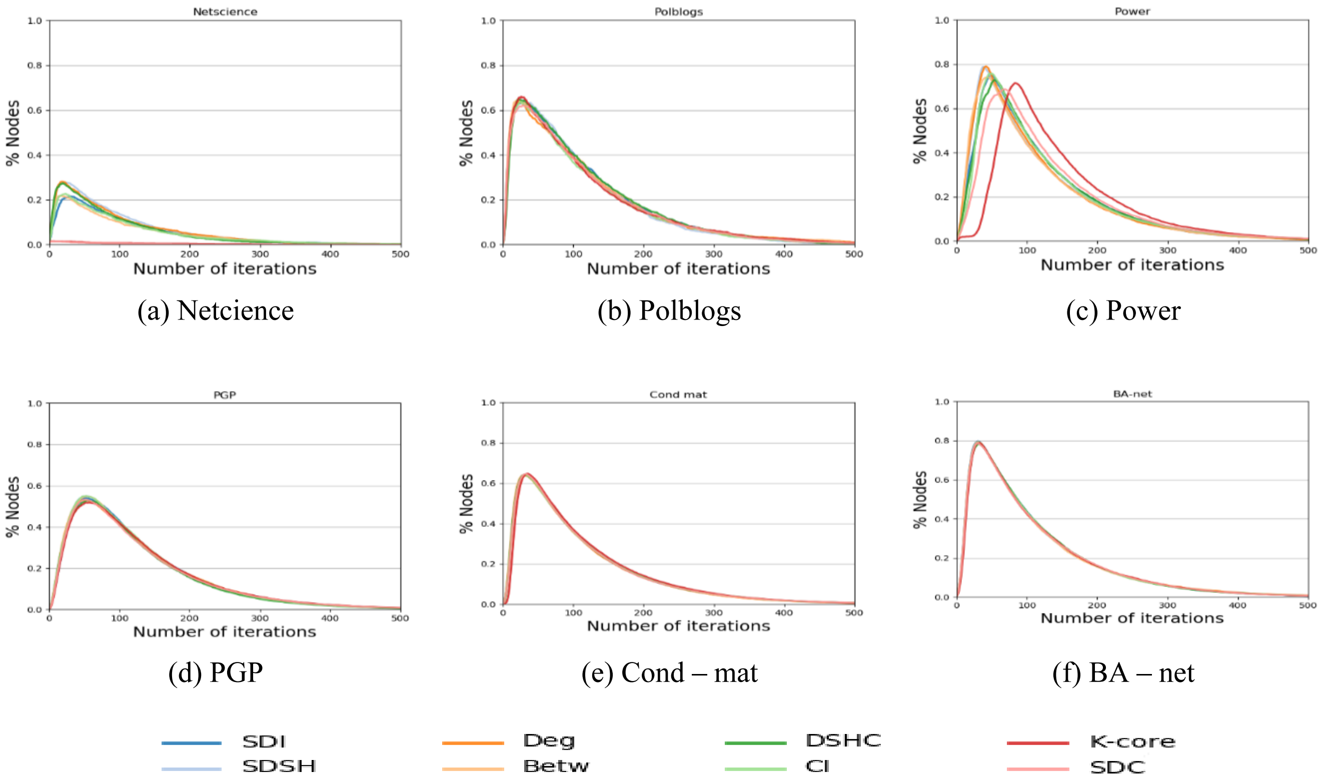
The popular SIR epidemic model is adopted to quantify the disease propagation power of some influential nodes. This model has three compartments, Susceptible (S), Infected (I) and Recovered (R) compartments [21]. Before the epidemic propagation starts, all nodes are assumed to be uninfected and are in the Susceptible state. Then, to set off the epidemic propagation, one or a group of highly influential nodes are chosen as seed nodes to start the spread. These nodes and all subsequent infected nodes move to the Infected state. The infected nodes can infect susceptible nodes with the probability β. Also, all the infected nodes can recover to the Recovered state with the probability λ. In this work, the propagation threshold β_{th} as shown for each network in Table 2 was obtained with the model given in equation (11).

$$\beta_{th} = \frac{< k >}{< k^2 > - < k >} \tag{11}$$

To actualize the SIR epidemic spread experiment, NDlib Python package [22] was used to implement SIR diffusion trend. The spreading probability β was set to $\beta = \beta_{th}$. The value of the recovery rate was set at $\lambda = 0.01$ to allow the infection to spread on the networks. A group of 10 highly influential seed nodes were chosen as the initial infected nodes. Results from the diffusion spread comparison evaluation is shown in Figure 3.

Table 2. Properties and descriptions of network datasets

Network	n	m	$<k>$	C	$<d>$	R	β_{th}	Description
Netscience	1461	2742	3.7536	0.6937	1.138	0.4616	0.1684	Co-authorship network of scientists [16].
Polblogs	1224	16715	27.3121	0.3197	1.8687	-0.2212	0.0125	Network of hyperlinks among weblogs on US politics [17].
Power	4941	6594	2.6691	0.0801	18.9892	0.0035	0.3483	Western States Power Grid of the US [18].
PGP	10680	24316	4.5536	0.2659	7.4855	0.2382	0.0559	Interaction network of users of the Pretty Good Privacy algorithm [19].
Cond-mat	16264	47594	5.8527	0.638	1.1116	0.1846	0.0838	Co-authorship network of scientists posting preprints [20].
BA-net	10000	19996	3.9992	0.007	4.9484	-0.0393	0.0783	Artificial network



(a) Netcience (b) Polblogs (c) Power

(d) PGP (e) Cond – mat (f) BA – net

Figure 3. SIR diffusion trend comparison on different networks

For each network, the diffusion process was run for 500 iterations. From Netscience network shown in Figure 3(a), SDSH, DHSC and Deg have very rapid spread, infecting about 30% of the entire network. But, SDSH was sustained just a little longer more than the other algorithms. From Polblogs network shown in Figure 3(b), all the methods have rapid spread with K-core peaking at almost 63% over other methods. From Power network in Figure 3(c), SDSH, Deg and Betw have very rapid infection spread with SDSH and Deg peaking higher than Betw by infecting roughly 78% of the entire network.

Continuing, all the methods show uniform rapid spread on PGP network as shown in Figure 3(d) with SDSH, CI and Deg peaking above other methods at almost 56%. From the Cond-mat network shown in Figure 3(e), all the methods exhibit uniform performance. From BA-net shown in Figure 3(f), all methods exhibit uniform spread with SDI peaking at 80% slightly above other methods.

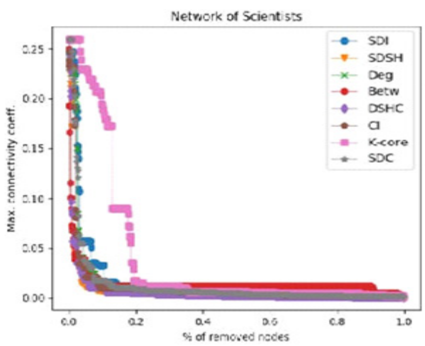
Overall, SDSH shows a very remarkable spreading performance in terms of rapid infection spread and the total number of population infected. This validates the fact that SDSH can detect highly influential or spreader nodes better than other methods.

4.3 Maximum Connectivity Coefficient

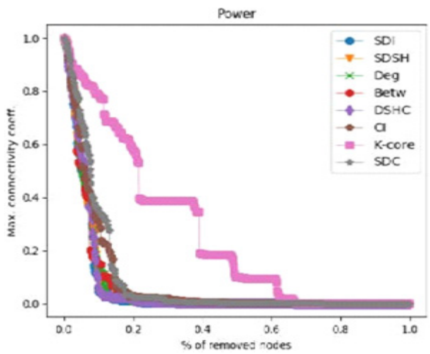
Maximum connectivity G is one of the metrics used in evaluating the performance of a centrality method. It is the maximum damage that may be caused to a network by the removal of highly influential nodes. The degree of change caused to a network's connectivity by the removal of a node is commensurate to the importance of that node. G is defined as:

$$G = \frac{R}{N} \tag{12}$$

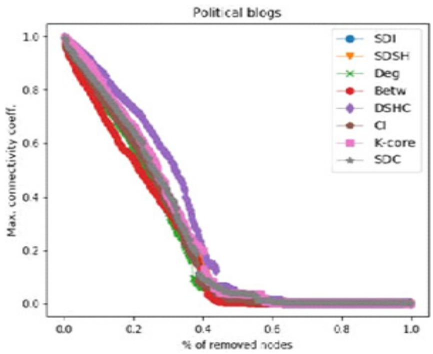
where R denotes number of nodes in the maximum connected component of a network after the removal of a node. N is the total number of nodes on the network. The lesser the value of G or the faster it decreases, the better the attack stratagem adopted [6]. The centrality methods were used as the attack strategies to destroy the networks by gradually removing the most important nodes consecutively. The results are presented in Figure 4.



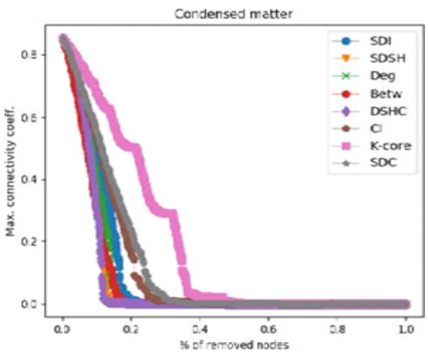
(a) Network of scientists



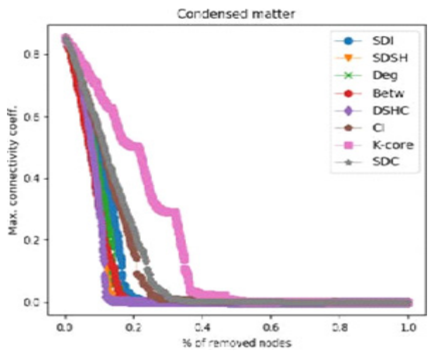
(b) Power grid network



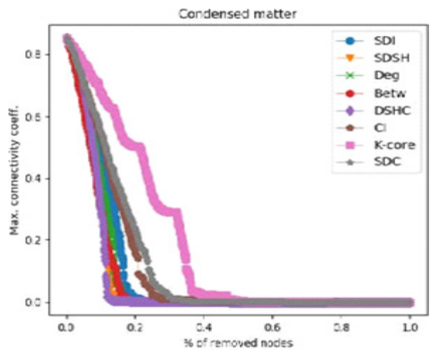
(c) Network of political blogs



(d) Condensed matter network



(e) Pretty good privacy network



(f) Barabarsi-Albert artificial network

Figure 4. Maximum connectivity coefficient results of various networks

In network of scientists shown in Figure 4(a), SDSH seem to have caused the most damage to the network when about 5% of the most important nodes were removed from the network. Following closely is DSHC and CI. In power grid network shown in Figure 4(b), SDI, SDSH and DSHC have the best performance. The three methods almost have a tie in destroying the network when about 13% of the most important nodes are removed. In network of political blogs shown in Figure 4(c), Deg and Betw surprisingly have the best performance when about 45% of the most important nodes are removed. DSHC has the worst performance. In condensed matter network shown in Figure 4(d), DSHC have the best performance followed closely by SDSH. They seem to collapse the network when about 17% of the most important nodes are removed. In pretty good privacy network shown in Figure 4(e), SDSH and DSHC have a tie as they put up the best performance in collapsing the network when about 13% of the most important nodes are removed. They are closely followed by Deg. In Barabarsi-Albert artificial network shown in Figure 4(f), Deg and DSHC have a tie as they have the best performance in collapsing the network when about 15% of the most important networks are removed. SDI and SDSH collapse the network completely when about 20% and 30% of the most important nodes are removed from the network.

The ability of the proposed metrics to discover very influential nodes capable of causing maximum network damage on real-world networks are highly impressive especially for SDSH. Overall, in terms of wider area of applicability, the proposed centrality methods can be effectively applied on both weighted and unweighted networks to achieve impactful results. The compared metrics are limited in this sense as they can only be applied to unweighted networks.

4.4 Computational Efficiency

From Table 3, Degree centrality has the lowest computational complexity of $O(n)$ followed by SDI, SDC and K-core with $O(n \log n)$, $O(n \log n)$ and $O(m)$ respectively. SDI's time complexity is reasonably fast and can complete computations in a reasonable time limit but SDSH's time complexity is somewhat high but it is better than Betweeness centrality which has the highest time complexity.

Table 3. Computational complexity of the algorithms

Algorithm	Complexity
SDI	$O(n \log n)$
SDSH	$O(n(n \log n)^2)$
Deg	$O(n)$
Betw	$O(nm)$
DSHC	$O(n(n \log n)^2)$
CI	$O(n^2 \log n)$
K-core	$O(m)$
SDC	$O(n \log n)$

5 Conclusions and Recommendations

There has been an increased need to design suitable algorithms to distinguish nodes to facilitate the discovery of top spreaders. In this work, two centrality algorithms namely Sub-graph Degree Information centrality (SDI) and Sub-graph Degree and Structural Hole (SDSH) centrality were proposed. Experiments and comparison analyses were conducted on five real-world datasets and one artificial network. SDSH showed impressive performance in network destruction on four of the real-world dataset as it declined faster than the other methods. The proposed methods can be deployed in areas such as epidemic or rumour control, viral product campaigns or product recommendation and so on.

Finally, we hope to extend the algorithms implementation to directed networks and we will apply them to several other application scenarios.

Funding

This work is partially supported by Fundamental Research Funds for the Central Universities N2017009, N2017008, N181706001, N182608003, N161702001, N2018008, N181703005), National Natural Science Foundation of China (61902057), the Doctoral Start-up Funds of Liaoning Province (2019-BS-084).

Acknowledgements

C.E. is grateful to Ali Ahmad Hasan who made available his PC for the experiments through his friend Akubue Uchenna.

References

[1] E. Estrada, J. A. Rodŕıguez-Vel´azquez, Subgraph centrality and clustering in complex hyper-networks, *Physica A: Statistical Mechanics and its Applications*, Vol. 364, pp. 581-594, May, 2006.

[2] T. Qiao, W. Shan, C. Zhou, How to identify the most powerful node in complex networks? a novel entropy centrality approach, *Entropy*, Vol. 19, No. 11, Article No. 614, November, 2017.

[3] W. Wang, W. Street, Modeling influence diffusion to uncover influence centrality and community structure in social networks, *Social Network Analysis and Mining*, Vol. 5, No. 1, Article No. 15, May, 2015.

[4] S. B. Andrews, Reviewed Work: Structural holes: The social structure of competition. by R. S. Burt, *Administrative Science Quarterly*, Vol. 40, No. 2, pp. 355-358, June, 1995.

[5] G. Ahuja, Collaboration networks, structural holes, and innovation: A longitudinal study, *Administrative Science Quarterly*, Vol. 45, No. 3, pp. 425-455, September, 2000.

[6] H. Yang, S. An, Critical nodes identification in complex networks, *Symmetry*, Vol. 12, No. 1, Article No. 123, January, 2020.

[7] R. Burt, Structural holes and good ideas, *American Journal of Sociology*, Vol. 110, No. 2, pp. 349-399, September, 2004.

[8] D. Stevanovic, Comment on "subgraph centrality in complex networks", *Physical review E, covering statistical, nonlinear, biological, and soft matter physics*, Vol. 88, No. 2, Article No. 026801, August, 2013.

[9] L. Lu, T. Zhou, Q.-M. Zhang, H. E. Stanley, The h-index of a network node and its relation to degree and coreness, *Nature Communications*, Vol. 7, Article No. 10168, January, 2016.

[10] D. Chen, L. Lu, M. Shang, Y.-C. Zhang, T. Zhou, Identifying influential nodes in complex networks, *Physica A: statistical Mechanics and Its Applications*, Vol. 391, No. 4, pp. 1777-1787, February, 2012.

[11] W. Kang, G. Tang, Y. Sun, S. Wang, Identifying influential nodes in complex network based on weighted semi-local centrality, *2nd IEEE International Conference on Computer and Communications*, Chengdu, Sichuan, China, 2016, pp. 2467-2471.

[12] Y. Liu, J. Liu, Z. Zhang, L. Zhu, A. Li, Rem: From structural entropy to community structure deception, *33rd Conference on Neural Information Processing Systems*, Vancouver, Canada, 2019, pp. 12918-12928.

[13] T. Opsahl, F. Agneessens, J. Skvoretz, Node centrality in weighted networks: Generalizing degree and shortest paths, *Social Networks*, Vol. 32, No. 3, pp. 245-251, July, 2010.

[14] X. Li, K. Wang, L. Liu, J. Xin, H. Yang, C. Gao, Application of the entropy weight and TOPSIS method in safety evaluation of coal mines, *Procedia Engineering*, Vol. 26, pp. 2085-2091, 2011.

[15] M. Dehmer, Information processing in complex networks: Graph entropy and information functionals, *Applied Mathematics and Computation*, Vol. 201, No. 1-2, pp. 82-94, July, 2008.

[16] M. Newman, Finding community structure in networks using the eigenvectors of matrices, *Physical review E, covering statistical, nonlinear, biological, and soft matter physics*, Vol. 74, No. 3, Article No. 036104, September, 2006.

[17] L. A. Adamic, N. S. Glance, The political blogosphere and the 2004 U.S. election: divided they blog, *3rd international workshop on Link discovery*, Chicago, Illinois, USA, 2005, pp.

36-43.
[18] D. J. Watts, S. H. Strogatz, Collective dynamics of 'small-world' networks, *Nature*, Vol. 393, No. 6684, pp. 440-442, June, 1998.
[19] M. Boguna, R. Pastor-Satorras, A. Diaz-Guilera, A. Arenas, Models of social networks based on social distance attachment, *Physical review E, covering statistical, nonlinear, biological, and soft matter physics*, Vol. 70, No. 5, Article No. 056122, November, 2004.
[20] M. E. J. Newman, The structure of scientific collaboration networks, *Proceedings of the National Academy of Sciences of the United States of America*, Vol. 98, No. 2, pp. 404-409, January, 2001.
[21] K. Berahmand, A. Bouyer, N. Samadi, A new centrality measure based on the negative and positive effects of clustering coefficient for identifying influential spreaders in complex networks, *Chaos, Solitons & Fractals*, Vol. 110, pp. 41-54, May, 2018.
[22] G. Rossetti, L. Milli, S. Rinzivillo, A. Sirbu, D. Pedreschi, F. Giannotti, NDlib: a Python Library to Model and Analyze Diffusion Processes Over Complex Networks, *Journal of Data Science and Analytics*, Vol. 5, No. 1, pp. 61-79, February, 2018.

Biographies

Chinenye Ezeh received B.Eng and M.Eng degrees in Computer Engineering from ESUT and UNIYO in 2008 and 2015 respectively. He is a faculty member Computer Engineering Dept., Michael Okpara University of Agriculture, Umudike. He is a PhD student at Northeastern University, Shenyang, China. His research interests include complex networks.

Tao Ren received the B.S. degree in Automatic Control, M.S. and Ph.D. degrees in control Theory and Control Engineering in 2003, 2005 and 2007 respectively from Northeastern University, Shenyang, China. He is currently a professor in Northeastern University. His current research interests include complex network and intelligent optimization algorithm.

Yan-Jie Xu received the B.S. degree in Software Engineering from Nanchang University, Nanchang, China, in 2015. She is studying for a Ph.D. degree in Software Engineering from Northeastern University, Shenyang, China. Her current research interests include complex network and machine learning.

Shi-Xiang Sun received the B.S. and the M.S. degrees in School of Mathematics and Statistics Science from Ludong University, Yantai, China, in 2014 and 2018 respectively. She is currently a doctoral student in Northeastern University. Her current research interests include intelligent optimization algorithm, complex network and stochastic control.

Zhe Li received the B. S. degree in Information Security, the M. S. degree in Communication and Information System and Doctorate degree in 2011, 2015 and 2020 respectively from Northeastern University, Shenyang, China. His current research interests include complex network and machine learning.