Identifying Influential Nodes in Complex Networks Based on Local and Global Methods

Li Mijia*, Wei Hongquan1, Li Yingle1, Liu Shuxin1

1PLA Army Strategic Support Force Information Engineering University

*Li Mijia’s e-mail: 2014141462126@stu.scu.edu.cn

Abstract. Identifying Influential Nodes in complex networks is of great significance in both theory and reality. K-shell decomposition method is a local method which is suitable for increasing scale of complex networks but limited in accuracy because many nodes are partitioned with the same K-shell value. To overcome the coarse result of K-shell, an improved K-shell which considers the number of nodes’ iteration layers and degrees is proposed. Unlike local methods, global methods such as Betweenness Centralities (BC) are accurate but time-consuming. We employed an algorithm framework which combines advantages of both local and global methods where core network is extracted by improved K-shell and then BC is used to quantitatively analyze nodes in the core network. We compare the proposed method with other existing methods on Susceptible-Infective-Removal (SIR) mode. Experiments on three real networks show that the proposed method is more efficient and accurate.

1. Introduction

Social networks are collections of relationships between individuals and groups, including interpersonal networks, Twitter networks, paper co-authoring networks, etc. Identifying key nodes in social networks is of great research value, especially when there are millions or even tens of millions of nodes in the network.

Degree centrality supposes that node with more neighbors is more important, which is simple but inaccurate especially in large-scale networks. Kitsak M [1] proposed K-shell that node's position in the network needs to be considered when measuring its importance, which is suitable for large-scale networks because of low complexity. But the granularity of the division results is large and most nodes are in the same shells. Wangmin et al. [2] proposed an improved algorithm based on K-shell and node information entropy, which is fast but neglects nodes’ global information. Both Degree Centrality and K-shell are local methods which are fast but less accurate.

Global methods such as Closeness Centrality (CC) and Betweenness Centrality (BC) are accurate but time-consuming. Linton et al.[3] proposed CC. The closer the nodes are, the closer they are to the network center and the more important they are. Freeman et al.[4] proposed BC, the more the shortest paths go through a node, the more important the node is.

We improve K-shell by considering the number of nodes’ iteration layers and degrees. Then we propose a fast algorithm framework [5] by using a microstructure to replace the original network. First, the improved K-shell decomposition method is used to partition the networks into several shells. Nodes in higher-level shells are extracted as a new core network. Then we calculate nodes’ BC value in the core network. Combining nodes’ K-shell with BC value, the network is quantitatively analyzed to find out key nodes. Experiments show that the proposed method can significantly reduce the time complexity of BC, while being more accurate and efficient.
2. Methods based on improved K-shell and Betweeness Centrality

2.1. K-shell algorithm

K-shell algorithm is illustrated below in Figure 1, which is an unweighted, directionless network consisting of 15 nodes and 19 edges.

The first step is removing the 1-degree nodes and their connected edges, then the number of these nodes’ iteration layer is marked as 1. In the remaining subgraph, degree of some nodes turns to 1. Remove these nodes and their edges, mark their iteration layer as 2. Observe whether there is a node with degree 1 in the remaining subgraph. If yes, remove the node and its connected edges, the number of their iteration is also incremented by 1. Repeat this step until there are no nodes with degree 1 in the network.

Repeat until all nodes are deleted.

Finally, the network is partitioned into three layers by K-shell decomposition. As Figure 2 shows, the darker the node is, the higher the degree value is.

![Figure 1. Undirectional network](image1)

![Figure 2. K-shell decomposition](image2)

| Number of iteration | node          | Ks |
|---------------------|---------------|----|
| 1                   | 1,2,3,7,9,15  | 1  |
| 2                   | 4,14          | 1  |
| 3                   | 5,6,13        | 2  |
| 4                   | 8,10,11,12    | 3  |

2.2. Improved K-shell algorithm

In Figure 2, node 1 and node 4 are in the same layer by K-shell decomposition with the same K-shell value, but obviously 4 is more important than 1. In the partition process, their iteration layers are different, degrees are also different. To get a higher distinguishability, we combine node’s Ks value with its iteration layer and degree value, which is defined as KD.

\[
KD (i) = K \times D + n
\]

(1)

In Equation (3), \( K \) is node’s Ks value, \( D \) is node’s degree, \( n \) is node’s iteration layers when node is deleted.

In Figure 1, iteration layer of node 4 is 2, \( D \) is 4, \( K \) is 1, \( KD \) is 6. Iteration layer of node 1 is 1, \( D \) is 1, \( K \) is 1, \( KD \) is 2. As we can see, the improved K-shell method has a higher distinguishability than traditional K-shell method.
2.3. Betweenness Centrality
Betweenness Centrality refers to the proportion of the number of the shortest paths passing through the node in the total number of the shortest paths in the network. BC value of node $i$ is defined as:

$$BC(i) = \sum_{m,n} {\delta^i_{mn} \over \delta_{mn}}$$

where $\delta_{mn}$ is the number of the shortest paths between node $m$ and node $n$; $\delta^i_{mn}$ is the number of the shortest paths between node $m$ and node $n$ that go through node $i$.

2.4. Social network modelling
We carried out experiments according to three stages which combines the advantages of both K-shell and BC. The main steps to build a key nodes mining model are as follows:

- Analyze network data and make adjacency matrix of directed network.
- Use the improved K-shell decomposition method to quickly score all nodes in the network.
- According to nodes’ KD value, strip about 80% of the unimportant nodes and related links to reduce the scale of the network.
- BC algorithm is used to score the nodes in the core network extracted in the previous stage. Normalized both KD value and BC value of nodes in the core network.
- Multiply KD value and BC value to get the results and sort them. Then find out the key nodes.

The combination of the two algorithms not only reduces the time complexity of BC, but also helps to improve the accuracy, which is called KDB algorithm.

3. Evaluation criteria and datasets

3.1. Evaluation criteria
The SIR model [6] is now commonly used in the fields of disease transmission, information dissemination, and rumor transmission. It consists of three types of states: Susceptible state $S$ where the node is not infected but is likely to be affected by information or disease; Infected state $I$ where the node is infected and has the ability to infect its neighbor nodes; Recover state $R$ where once the node recovers it will never be infected again.

In the initial stage of transmission, certain infected nodes spread information or disease to other nodes whose initial state are denoted as $S$. Nodes in state $I$ can infect neighbors who are in state $S$ with probability $\beta$ and be transformed into state $R$ with probability $\gamma$. This process will continue in a loop until there are no nodes in state $I$ in the entire network. The dynamic equations of the model are defined in Equation (4):

$$\begin{align*}
\frac{ds(t)}{dt} &= -\beta S(t)I(t) \\
\frac{dI(t)}{dt} &= \beta S(t)I(t) - \gamma I(t) \\
\frac{dR(t)}{dt} &= \gamma I(t)
\end{align*}$$

In the SIR model, the number of all nodes is denoted as $N=S(t)+I(t)+R(t)$ where $S(t)$, $I(t)$, $R(t)$ are the number of nodes in three states in the network at time $t$. The influence of a node in the network is represented by the number of nodes in state $R$ when spreading starts from this node.

3.2. Datasets
The Koblenz Data Repository is a tool published online for those engaged in big data processing. In this paper, we selected three directed and unweighted networks as the experimental networks. Details are shown in Table 2:
Table 2. Basic statistical characteristics of the three datasets

| Networks  | Number of nodes | Number of edges | Sparsity    |
|-----------|-----------------|-----------------|-------------|
| Physicians| 241             | 1098            | 1.90×10⁻²   |
| Blogs     | 1224            | 19025           | 1.27×10⁻²   |
| Ciation   | 12591           | 49743           | 3.14×10⁻⁴   |

4. Experiment and analysis

4.1. Experimental analysis

To evaluate the effectiveness of different methods, SIR model is used to examine node’s influence which are evaluated as the number of nodes covered when spreading from this node. DC, PageRank [7], LeaderRank [8] and KDB are compared in this paper. The effectiveness of each algorithm is represented by the sum of the spreading ability of the nodes that rank in the top-10 list.

As is shown in Figure 3, the cumulative number of the nodes in state R is plotted as a function of time. When the network is small, DC outperforms KDB and PageRank, but the accuracy of DC decreases dramatically as the network size becomes larger because DC considers node's local information without considering its position and higher-order neighbors, resulting that some marginal nodes can increase the degree value by establishing contiguous links with a large number of ordinary nodes.

In contrast, KDB performs poor when the network size is small. This is because K-shell works well when the difference between the two nodes’ KS values is relatively large (e.g., more than 10). Whereas, in small-scale networks, the KS values of nodes do not differ much from each other, resulting in some important nodes being removed when the core network is extracted using K-shell decomposition. As is shown in Figure 3, the accuracy of KDB algorithm increases as the network size gets larger. In the Citation network, the KDB algorithm is even more accurate than LeaderRank.

![Figure 3. KDB algorithm on three networks compared to other three methods](image-url)
The accuracy rate of the top-10 nodes mined by the four algorithms is showed in Table 3 compared to nodes mined by SIR model. The accuracy of DC drops sharply as the network scale increases. In contrast, PageRank and LeaderRank algorithms are more stable regardless of the network scale. The new algorithm KDB proposed in this paper has a higher accuracy rate as the network size increases. However, since the experiment uses SIR model as the criterion to evaluate the influence of nodes and each node needs to be calculated at least 300 times, it takes a long time to reach stable. Given the time cost, the maximum number of nodes in social networks selected in this paper is only 12,000. Predictably, KDB will work better on a larger scale network.

Table 3. The accuracy rate of the top-10 nodes mined by the four algorithms compared to that of the SIR model

| Datasets   | O(DC,SIR) | O(PageRank,SIR) | O(LeaderRank,SIR) | O(KDB,SIR) |
|------------|-----------|-----------------|-------------------|------------|
| Physicians | 0.8       | 0.5             | 0.5               | 0.3        |
| Blogs      | 0.5       | 0.5             | 0.6               | 0.6        |
| Citation   | 0.1       | 0.6             | 0.7               | 0.8        |

4.2. Computational complexity analysis

Table 6 shows the time complexity of the different methods in this paper. It can be seen from the calculation process of KDB that the time complexity of KDB depends on the calculation of K-shell method. KDB greatly improves accuracy with the same time complexity as K-shell. Meanwhile, the time complexity of KDB is much lower than that of LeaderRank and PageRank. In Table 4, n and m are the number of nodes and edges in the network, I is the number of iteration.

Table 4. Time complexity of ranking methods mentioned in this paper

| Method  | DC    | KDB   | K-shell | PageRank | LeaderRank | BC |
|---------|-------|-------|---------|----------|------------|----|
| Category| Local | Hybrid| Local   | Global   | Global     | Global |
| Complexity| O(n) | O(n)  | O(n)    | O(mI)    | O(mI)      | O(mn) |

5. Conclusion

In this paper, we present a new approach to detect influential nodes by comprehensively considering nodes’ local and global information. It proves that KDB can not only get a higher distinguishability, but also has a low time complexity, which is suitable for large-scale networks.

References

[1] Kitsak M, Gallos LK, Havlin S, et al. Identification of influential spreaders in complex networks. Nature Physics. 2010;6(11):888-93.
[2] Min Wang, Wanchun Li, Yuning Guo, Xiaoyan Peng, Yingxiang Li. Identifying influential spreaders in complex networks based on improved k-shell method[J]. Physica A: Statistical Mechanics and its Applications, 2020, 554.
[3] Isaiah G. Adebayo, Yanxia Sun. A novel approach of closeness centrality measure for voltage stability analysis in an electric power grid[J]. International Journal of Emerging Electric Power Systems, 2020, 21(3).
[4] Costa Andrea, Petrenko Anne A, Guizien Katell, Doglioli Andrea M. On the calculation of betweenness centrality in marine connectivity studies using transfer probabilities.[J]. PloS one, 2017, 12(12).
[5] Yingjie Xia, Xiaolong Ren, Zhengchao Peng, Jianlin Zhang, Li She. Effectively identifying the influential spreaders in large-scale social networks[J]. Multimedia Tools and Applications, 2016, 75(15).
[6] Leonid Shaikh. Improving Stability Conditions for Equilibria of SIR Epidemic Model with Delay under Stochastic Perturbations[J]. Mathematics, 2020, 8(8).
[7] Francisco Pedroche, Regino Criado, Julio Flores, Esther García, Miguel Romance. On PageRank versatility for multiplex networks: properties and some useful bounds[J]. Mathematical Methods in the Applied Sciences, 2020, 43(14).

[8] Li L, Zhang Y C, Yeung C H, et al. Leaders in social networks, the delicious case. PLoS One, 2011, 6: e21202.