A Graph-Based Shortest Path Community Expansion Method

Wang Wenzhang and Zheng Xiaoyan(✉)

Tianjin University of Technology and Education, Tianjin 300222, China
zhengxy@tute.edu.cn

Abstract. With the advent of the era of big data, research on community discovery has become more and more popular. For the division of nodes in the network, the mainstream method is to calculate the fitness function of nodes and communities. This paper proposes a graph-based shortest path community expansion method (A Graph-Based Shortest Path Community Expansion Method, hereinafter referred to as SPCE algorithm). The algorithm mainly includes four steps: selecting seed nodes, expanding seed communities, finding overlapping nodes, and optimizing communities. In the process of community expansion, the SPCE algorithm does not use the current mainstream fitness function method for community expansion. Instead, it uses the characteristics of dense connections within communities and sparse connections between communities to expand using the shortest path method of graphs. After experiments in real networks and artificial networks, the SPCE algorithm can more accurately discover the community structure in the network.

Keywords: community discovery · shortest paths · community expansion · overlapping communities

1 Introduction

People are connecting more frequently and having tighter relationships as society develops and science and technology grow, and this intricate web of connections has given rise to a sophisticated social network. Protein interaction networks, email networks, gene association networks, metabolic networks, transportation networks, and many other networks are examples of networks that are comparable. Because of its complex structure, network development, diversity of connections and nodes, and multi-complexity fusion, this sort of network is referred to as

Supported by Tianjin Science and Technology Planning Project (Grant No. 64822KPXMRC00170), Science and Technology Think Tank Young Talent Program, China (Grant No. 64920220615ZZ07110153).

Y. Cao and X. Shao (Eds.): MONAMI 2022, LNICST 474, pp. 293–305, 2023.
https://doi.org/10.1007/978-3-031-32443-7_21

a complex network. Complex network research has long been popular in a wide range of disciplines. Complex networks frequently have community structure, and the network as a whole is made up of several communities. While connections between communities are scarce, those inside a community are intimately intertwined. To fully comprehend the structure and operation of a network, the appropriate social structure must be identified.

The biological field, metabolic network analysis, gene regulatory network analysis, master gene identification, etc., are currently the main applications of community discovery. In the new crown epidemic from the previous year, we can stop the epidemic's spread by strengthening protection, identifying medium- and high-risk areas, and disrupting the virus's transmission network. Community discovery is used in e-commerce to examine groups of individuals in order to carry out more precise ad placement, create a more trustworthy recommendation system, and actualize tailored interest suggestions. Additionally, community discovery may be used to investigate criminal activity, successfully attack criminal networks, and preserve social order.

2 Related Research

Among community discovery methods, it can be divided into global optimization and local optimization [1], where local optimization does not require the information of the entire network. Therefore, when the network size is larger, local optimization is more chosen. According to different strategies, local optimization can be mainly divided into four types: local expansion, faction filtering, label propagation, and local edge clustering [2].

Li et al. [3] proposed the CLFMw algorithm. Based on faction filtering, the LFM algorithm [4] is improved and combined with the CPM algorithm [5]. Start by finding the largest faction in the network for community expansion. After the max faction expansion is complete, lower factions re-expand until the min 2-faction. However, the small faction community may be expanded and annexed by the large faction community, resulting in an inaccurate division. Xie et al. [6]] proposed the CSLPA algorithm. First, the same BronKerbosch algorithm as the GCE algorithm [7] is used to find the K factions in the network. Then merge similar factions, treating each faction as a node. Start to iterate, the iterative process is the same as the SLPA algorithm [8]. The algorithm ends when the number of iterations is reached or the result no longer changes.

Although the faction approach can solve the problem that the community finds unstable results, it still has shortcomings. Not suitable for less dense networks. The results of the label transmission method are not stable and are strongly random. In contrast to the local expansion process, the seed nodes are selected by ranking, and communities of any size and density can be divided, and the community division results are stable. Therefore, the selection of seed nodes is very important. The selection of seed nodes can be divided into global ranking and local ranking.

Guo et al. [9] proposed the oclu-detect method, which calculates the node weight by the average connection strength and association density between the

node and its adjacent nodes. The node with the largest weight is selected as the central node for community expansion. The second step is to calculate the membership degree of the node to the community and complete the dynamic division of the community according to the changing trend of the newly added data. Yang et al. [10] proposed a new method, which is mainly divided into three steps. In the first step, the connection strength between two adjacent nodes is calculated to convert the unweighted network into a weighted network. The second step is to generate the maximum spanning tree through the authorized network, find the core nodes in the network, and expand it. The third step is to optimize the community and merge overlapping communities.

For the method of selecting seeds by local ranking, Chen et al. [11] proposed the LMD algorithm, which uses the node with the largest local node degree as the central node to realize local community discovery. To become a central node, the degree of the node must be greater than or equal to that of other surrounding nodes, so only one node can become the central node in the local scope. Then expand the community to complete the community division. Wang et al. [12] proposed that the LCD-NJ algorithm needs to give an initial node and ensure that the initial node can be transferred to any core node in the local community within k steps. Through the PageRank algorithm, the nodes within the k steps of the initial node are ranked, and the core nodes are selected. Afterward, it expands outward through core nodes to complete social discovery.

This paper proposes a graph-based shortest path community expansion method (SPCE) and selects seed nodes through a combination of global and local methods. The node with the highest influence ranking is selected as the seed node, and its neighbor nodes are removed from the ranking list. After that, the seed node is expanded into a seed community, and the sum of the shortest path distances from the free node to the n nodes in the seed community is calculated. Add it to the seed community with the smallest distance to complete the community expansion. Then judge the overlapping nodes for the edge nodes. Finally, similar communities are merged to complete the division.

3 SPCE Algorithm

SPCE algorithm proposes new methods in three aspects: seed node selection, community expansion, and overlapping node detection. Specifically: (1) the SPCE algorithm adopts a combination of global and local methods in selecting seed nodes, avoiding the use of only global The problem is that the diversity of seed nodes cannot be guaranteed due to ranking. (2) A new shortest distance-based community expansion method is proposed by taking advantage of the characteristics of close connections within communities and sparse connections between communities. (3) In view of the large amount of calculation caused by the need to calculate the fitness between all nodes and all communities to find overlapping nodes using the fitness function, a new method for finding overlapping nodes is proposed, which only needs to calculate the edge nodes of the community, which can effectively reduce the amount of computation.

3.1 Algorithm Description

The algorithm is mainly divided into four steps: (1) seed node selection and seed community formation (2) seed community expansion (3) overlapping node discovery (4) community optimization. All node influence values are calculated in the first phase and are arranged in descending order of size. The node with the highest influence value in the list is chosen, and in order for it to serve as a seed node, its node degree must be higher than that of the majority of its neighbors. The seed node is then combined with its neighboring nodes to form a seed community. The free node is added to the closest seed community after calculating the shortest path between it and every other node in the seed community. Finding the community edge nodes' neighbors is the third phase. Calculate the similarity between the next node and the current community, and if it exceeds the threshold, join the community if it does not already belong. The merging of communities with a lot of similarities is the fourth step.

3.2 Seed Node Selection and Seed Community Formation

The selection of seed nodes is very important for the subsequent community expansion process, so the selection of seed nodes should be as close to the community center as possible. In a community network, the connections within a community tend to be tighter, which means there are more edges within the community. On the contrary, there are fewer edges between the community and the community, so the seed node of the community must have the characteristics of a high node degree. The influence value of a node is calculated by combining the node degree with the closeness of the node adjacent to the node. The formula for calculating the node influence value I(v) is as follows:

$$I(v) = k_v \times \sum_{u \in N(v)} (k_u \times J_{uv}) \tag{1}$$

The higher I(v), the higher the influence of node v in graph G, where kv is the degree of node v, and Juv is the Jaccard coefficient of nodes u, v, defined as follows:

$$Jaccard(u, v) = \frac{|N(u) \cap N(v)|}{|N(u) \cup N(v)|} \tag{2}$$

The larger the Jaccard coefficient, the higher the intimacy between the two nodes. The more similar the two nodes are. N(v) represents the set of adjacent nodes of node v, which is defined as follows:

$$N(v) = u : u \in V, (u, v) \in E \tag{3}$$

After calculating the node influence value, sort the nodes according to the node influence to get the ranking list (Inflist). Select the node n with the highest influence value and calculate that the influence of node n is greater than the

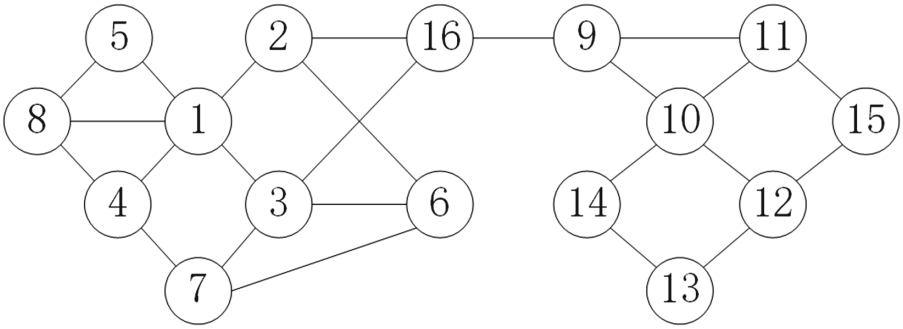

Fig. 1. Node distribution diagram

number L of all its adjacent nodes. When L is greater than the threshold, then n becomes the seed node. Remove node n and its neighbors from the ranking list (Inflist) to ensure that there is only one seed node in the local community. Continue to repeat this step until there are no nodes in the list. As shown in Fig. 1, the influence value of the nodes in the calculation graph is ranked, and the results are shown in Table 1. After selection, nodes 1 and 10 become seed nodes.

Table 1. Influence value table.

Node	Influence Value	Node	Influence Value
10	66.75	4	14.83
11	43.41	3	14.55
12	43.41	6	13.8
15	29.49	7	13
1	28.93	13	9.64
14	20.79	5	9.5
8	19.9	2	9.15
9	15.15	16	4.8

After obtaining the seed list (Seedlist), expand the seed node to obtain the seed community (Seedcommunity). According to formula (2), the neighbor node (Neigbornode) most similar to the seed node is calculated and extended to obtain the seed community. See formula (2) for calculating the similarity between the seed node and the adjacent node. As shown in Fig. 1, seed nodes 1 and 10 are expanded, and the two most similar nodes are selected to expand to become the seed community. Seed Community 1 (1, 5, 8) and Seed Community 2 (10, 11, 12) are available.

3.3 Seed Community Expansion

In a community network, the connections within a community tend to be tighter, which means there are more edges within the community. On the contrary, there are fewer edges between communities. Therefore, the distance from the internal node of the community to the community core must be smaller than the distance to other community cores. Use Dijkstra's algorithm to calculate the shortest distance sp1, sp2, sp3, ... between the free nodes in the graph and the nodes n1, n2, n3, ... in the seed community respectively. Calculate the sum of its shortest distances (Sumsp), get a list of distances (Splist) from free nodes to different seed communities, and select the seed community with the smallest sum of distances to join. Until all free nodes are divided, the community expansion is completed. Continuing to take Fig. 1 as an example, the results are shown in Table 2. Nodes 2, 3, 4, 6, and 7 are closer to seed community 1 and expand into community 1. Nodes 9, 13, 14, 15, and 16 are closer to seed community 2 and expand into community 2. Figure 1 is finally divided into two communities.

Table 2. Shortest distance table.

free node	Shortest distance from community 1	Shortest distance from community 2
2	5	10
3	5	16
4	4	16
6	8	13
7	7	16
9	11	4
13	17	5
14	17	4
15	17	3
16	8	7

3.4 Overlapping Node Discovery

In this step, overlapping nodes are primarily filtered. Find the nodes whose neighbor node (Neighbornode) and edge node (Sidenode) are members of separate communities. The similarity S between the community and its surrounding nodes can be calculated using formula (4). The node is included in the existing community if S exceeds the threshold. To finish the overlapping community discovery, iterate through each community. The following equation can be used to determine how similar a community and a node are:

$$S(C, v) = \frac{|N(C) \cap N(v)|}{|N(C) \cup N(v)|} \tag{4}$$

The larger S(C, v) is, the more likely the node v belongs to the community C. where N(C) represents the set of adjacent nodes of community C, which is defined as follows:

$$N(c) = \bigcup_{v \in c} N(v) \tag{5}$$

3.5 Community Optimization

Utilizing formula (6), the similarity S between the communities is determined as the final stage in community optimization. The two communities are joined if the resemblance is more than. Find out the outcome of the community division.

$$S(c_i, c_j) = \frac{|c_i \cap c_j|}{min(|c_i, c_j|)} \tag{6}$$

The larger S(Ci, Cj) is, the more similar the structures of the two communities Ci and Cj are. where Ci represents the nodes in the community and the adjacent nodes of the community.

4 Experiment

The experiment compares real data sets with fictional data sets of various scales in order to evaluate the performance of the algorithm. The testing environment consists of a laptop with an i7-10750H processor, a 6-core CPU, 16 GB of memory, and Windows 10 64-bit operating system. The algorithm code is implemented using Python 3.9.

4.1 Experimental Dataset

Real Dataset
The karate network Karate [13], the Dolphins network Dolphins [14], the American political book network Polbooks [15], the American college football network Football [16], and the Facebook network Facebook are selected. The details of the experimental dataset are shown in Table 3.

Table 3. Real network dataset.

network	number of nodes	number of sides
Karate	34	78
Dolphins	62	159
Polbooks	105	441
Football	115	616
Facebook	4039	88234

Artificial Dataset
Using the artificial simulation network generated by the LFR-benchmark benchmark program [17], a total of 5 groups of different artificial simulation networks are generated. The specific network parameter settings are shown in Table 4.

Table 4. The LFR-benchmark benchmark network dataset.

network	number of nodes	mu	om	on
A	1000	0.1,0.3,0.5	0.1	3
B	5000	0.1–0.5	0.1	3
C	5000	0.3	0.1	2–6
D	5000	0.3	0.3	2–6
E	1000–20000	0.3	0.1	3

The rest of the parameters were set with the same settings: k = 20, Kmax = 50, Cmin = 20, and Cmax = 100.

where mu represents the network's complexity, and the higher its value, the more complicated the network is. The percentage of overlapping nodes in the network is represented by on. In om, overlapping nodes are members of n communities simultaneously.

4.2 Evaluation Standard

Overlapping Modularity EQ: Overlapping modularity [18] is an improvement from modularity and is often used as an evaluation criterion for judging the quality of overlapping community structures. The greater the modularity, the clearer the structure of the community. Therefore, the closer the value of EQ is to 1, the better the quality of the community is divided by the algorithm. Modular EQ is defined as follows:

$$EQ = \frac{1}{2m}\sum_{k=1}^{c}\sum_{i,j\in C_k}\frac{1}{O_i O_i}[A_{ij} - \frac{k_i k_j}{2m}] \tag{7}$$

Among them, m represents the total number of edges in the network, c represents the number of divided communities, ki represents the degree of a node, Oi represents the number of communities to which a node belongs, and Aij represents whether there is an edge between nodes I and j. If there is an edge, yes is indicated by 1, not by 0.

Normalized Mutual Information NMI: Normalized Mutual Information [19] uses entropy to measure the difference between a standard network and an algorithmically partitioned network. Therefore, it is suitable as an evaluation criterion for artificially generated network division. The closer the value of NMI is to 1, the better the community effect of the algorithm. Standardized mutual information NMI is defined as follows:

$$NMI = \frac{-2\sum_{i=1}^{C_A}\sum_{j=1}^{C_B} N_{ij} log(\frac{N_{ij}\times N}{N_i \times N_j})}{\sum_{i=1}^{C_A} N_i log(\frac{N_i}{N}) + \sum_{j=1}^{C_B} N_j log(\frac{N_j}{N})} \tag{8}$$

Among them, CA represents the standard community partition result, CB represents the community partition result obtained by the algorithm, the row of matrix N corresponds to the standard community partition result, and the column of matrix N corresponds to the community partition result obtained by the algorithm, and the sum of the i-th row is denoted as Ni, the sum of the j-th column is denoted as Nj.

4.3 Experimental Comparison

Real Network Comparison

By comparing with the other two algorithms, the experimental results are shown in Fig. 2. The SPCE algorithm only lags behind the dolphin network and the polbooks network and has good performance in other networks. The CPM algorithm cannot complete the division of the facebook network, so the EQ value is 0. The experimental results show that the SPCE algorithm can better discover the community structure no matter in the low-node network or the high-node network.

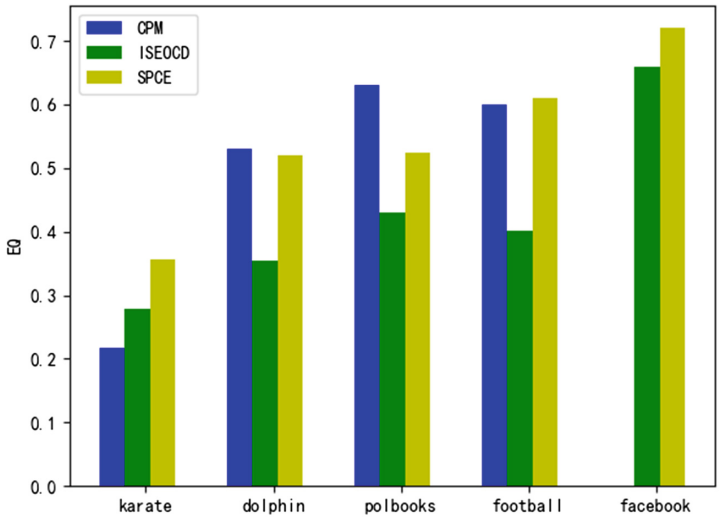

Fig. 2. Real network comparison

Artificial Network Comparison

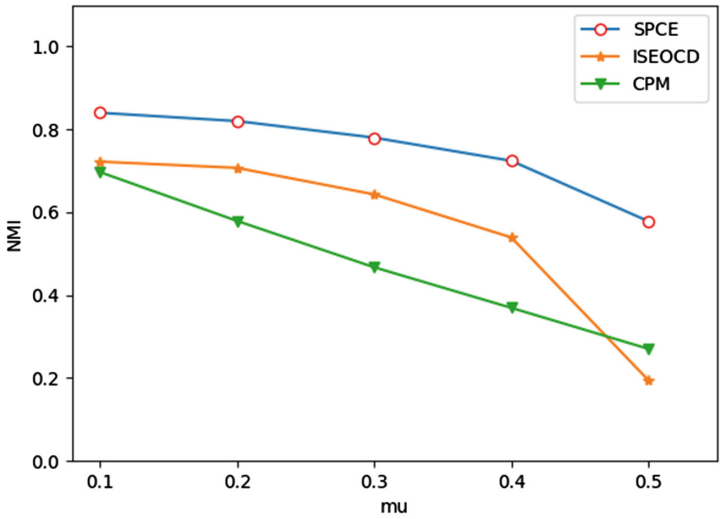

Fig. 3. Comparison of different mu values of 5000 nodes

Figure 3 shows the comparison under different mu values when the number of nodes is 5000, using artificially generated network group B to conduct experiments. Through the comparison of the three graphs, the SPCE algorithm performs well in the case of different numbers of nodes. As the complexity increases, the SPCE algorithm does not show a significant decrease in the NMI value, and the trends are similar in the three graphs, indicating that the SPCE algorithm performs stably under different complexities.

Figure 4 shows the comparison of different om values when on is 0.1 in artificially generated network group C, and Fig. 5 is when on is 0.3 in artificially generated network group D. By analyzing the two sets of data, when on is 0.1, the accuracy of SPCE algorithm decreases with the increase of om, but the performance of SPCE algorithm is always better than the other two methods in terms of community division. With on being 0.3, the accuracy of the SPCE algorithm keeps decreasing slowly while outperforming the other two methods as om grows. It shows that the SPCE algorithm still has good performance in the case of high complexity.

When the number of nodes in the synthetically created network group E varies, a comparison is shown in Fig. 6. As can be shown, the accuracy of the SPCE algorithm does not significantly decrease as the number of nodes rises. It demonstrates that even with a large number of nodes, the SPCE algorithm still operates effectively.

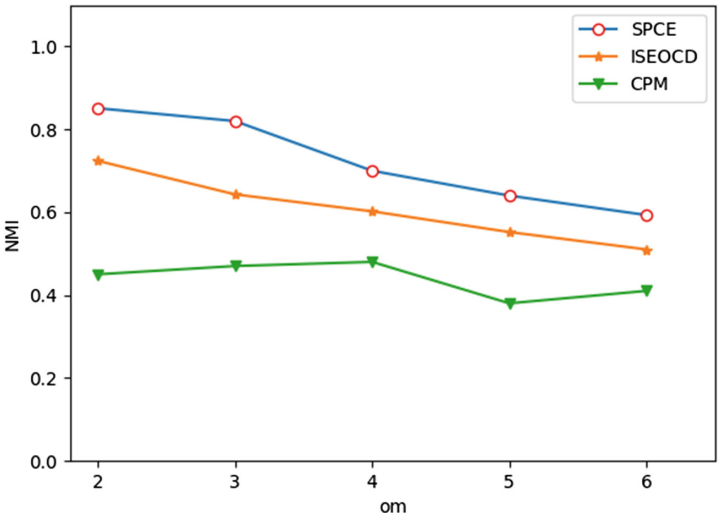

Fig. 4. Comparison of different om values at on = 0.1

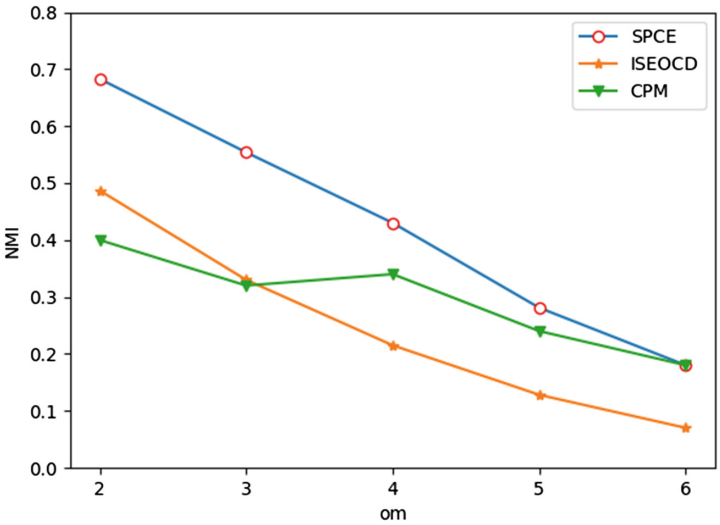

Fig. 5. Comparison of different om values at on = 0.3

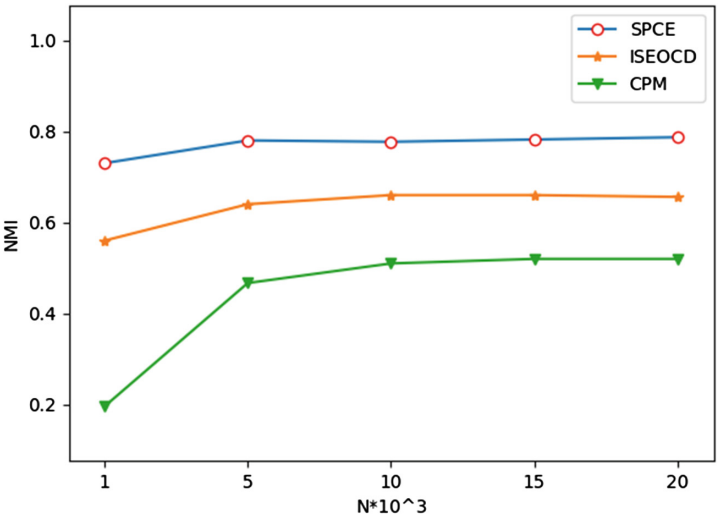

Fig. 6. Comparison of different numbers of nodes

5 Summarize

This paper proposes a graph-based shortest path community expansion method (SPCE). This method does not use the currently widely used fitness function to determine the attribution of nodes and seed communities. Instead, it utilizes the concept of tight connections within communities and sparse connections between communities in the network. It is inferred that the distance from the community node to the community center must be smaller than the distance to other community centers, so the nodes can be divided by the shortest path method of the graph. Experiments show that this method is effective, and has high accuracy in complex networks, strong stability in networks with many nodes, and can better divide the community structure.

References

1. Yancui, S., Yuan, W., Qing, Z., Xiankun, Z.: Research status of community discovery based on local expansion. J. Commun. **40**(01), 149–162 (2019)
2. Jianhua, L., Xiaofeng, W., Peng, W.: Research status of community discovery method based on local optimization. J. Chinese Acad. Sci. **30**(02): 238–247+180 (2015)
3. Jie, L., Xingwei, W., Jing, G., Chao, Y.: Local expansion group construction method for mobile communication networks. J. Northeast. Univ. (Nat. Sci. Ed.), **38**(12):1691–1695+1711 (2017)
4. Lancichinetti, A., Fortunato, S., Kertész, J.: Detecting the overlapping and hierarchical community structure in complex networks. New J. Phys. **11**(3), 033015–033015 (2009)

5. Lancichinetti, A., Fortunato, S.: Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. Phys. Rev. E **80**(1), 016118 (2009)
6. Xie, H., Yan, Y.: Detecting an overlapping community structure by using clique-to-clique similarity based label propagation. J. Korean Phys. Soci. **75**(6), 436–442 (2019)
7. Lee, C., Reid, F., Mcdaid, A., et al.: Detecting highly overlapping community structure by greedy clique expansion (2010)
8. Xie, J., Szymanski, B.K., Liu, X.: SLPA: uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process. In: 2011 IEEE 11th International Conference on Data Mining Workshops, pp. 344–349 (2011)
9. Lin, G., Wanli, Z., Tao, P.: Discovery of overlapping communities in social networks based on membership and analysis of dynamic cluster evolution. J. Electron. **44**(03), 587–594 (2016)
10. Yang, J.-X., Zhang, X.-D.: Finding overlapping communities using seed set. Physica A **467**, 96–106 (2017)
11. Chen, Q., Ting-Ting, W., Fang, M.: Detecting local community structures in complex networks based on local degree central nodes. Physica A **392**(3), 529–537 (2013)
12. Tao, W., Yang, L., Yaoyi, X.: A local community discovery algorithm based on core node jumping. J. Shanghai Jiaotong Univ. (Chin. Ed.) **49**(12), 1809–1816 (2015)
13. Zachary, W.W.: An information flow model for conflict and fission in small groups. J. Anthropol. Res. **33**(4), 452–473 (1977)
14. Lusseau, D., Schneider, K., Boisseau, O.J., Haase, P., Slooten, E., Dawson, S.M.: The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations. Behav. Ecol. Sociobiol. **54**(4), 396–405 (2003)
15. Newman, M.E.J., Girvan, M.: Finding and evaluating community structure in networks. Phys. Rev. E Stat. Nonlinear Soft Matter Phys. **69**(2), 026113 2004
16. Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. Proc. Nat. Acad. Sci. United States Am. **99**(12), 7821–7826 (2002)
17. Lancichinetti, A., Fortunato, S., Radicchi, F.: Benchmark graphs for testing community detection algorithms. Phys. Rev. E, Stat. Nonlinear, Soft Matter Phys. **78**(4), 046110 2008
18. Shen, H., et al.: Detect overlapping and hierarchical community structure in networks. Physica A: Stat. Mech. Appl. **388**(8), 1706–1712 (2009)
19. Danon, L., et al.: Comparing community structure identification. J. Stat. Mech. Theo. Exp. **2005**(09), P09008 (2005)