Research Article

Study on Discovery Method of Cooperative Research Team Based on Improved Louvain Algorithm

Dianting Liu,1,2 Kangzheng Huang,1 Chenguang Zhang,1 Danling Wu,1 and Shan Wu1

1College of Mechanical and Control Engineering, Guilin University of Technology, Guilin 541004, China
2College of Information Science and Engineering, Guilin University of Technology, Guilin 541004, China

Correspondence should be addressed to Kangzheng Huang; hkzstudy@163.com

Received 10 June 2021; Accepted 22 October 2021; Published 1 November 2021

Abstract

According to the needs of scientific research project research and development, the research of cooperative team excavation methods was carried out. Aiming at the current difficulties in accurately and reliably defining and identifying cooperative research teams from co-author network, an improved Louvain algorithm that integrates core node recognition was proposed: Louvain-LSCR algorithm. Based on the analysis of the Louvain algorithm, considering the local topology of the node in the network and the communication range of the node, a new algorithm LSCR for core node identification was constructed. The LSCR algorithm and Louvain algorithm were merged to obtain a new and improved algorithm, Louvain-LSCR. In this algorithm, the leaf nodes in phase 1 of Louvain algorithm were first pruned to reduce calculations; then, seed nodes were selected according to the LSCR algorithm in phase 2. The experimental results on related datasets show that LSCR algorithm has certain advantages in identifying core nodes. The modularity of Louvian-LSCR algorithm is better than other algorithms, and the community structure is more reasonable. It was verified that the algorithm can mine potential cooperative research teams in co-author network.

1. Introduction

Modern scientific research projects are becoming more and more complex and larger in scale, and their research and development often require multiple researchers to jointly complete [1]. Scientific research cooperation can effectively supplement the scientific research capabilities of scholars, and it is necessary and effective for multiple people to carry out project research and development work in cooperation. The first key is to find multiple suitable researchers to form a team. A cooperative research team is a special group formed by researchers with different skills or different knowledge to achieve a common goal. The cooperative research team found that the problem has attracted widespread attention from the academic community [2–4].

The methods discovered by the cooperative research team are mainly divided into two types: traditional cooperative research team discovery based on multiple explicit data and cooperative research team discovery based on network analysis. In recent years, the method based on network analysis has gradually replaced the traditional method; it first establishes an overall network through large-scale data and then uses the technology of community detection to discover small teams from the constructed network. In the real world, it is very important to discover communities in various fields [5, 6]. For example, in the IoT system, it is important to discover and analyze key parts that are easily damaged. Mining communities or nodes play an important role in the transmission of information in virtual communities [7, 8], and research and analysis are conducted on their behaviors. In addition, community detection can be applied to the knowledge graph [9, 10] to mine abnormal data [11, 12], which can lay the foundation for building a knowledge graph or big data mining providing services.

The cooperative research team is a group of researchers who work closely together, which can be understood as a community in the network reflected in the network structure. The members of the community cooperate closely, while the connections between members of different communities are sparse. In a complex network, nodes belonging
node recognition algorithm LSCR into the Louvain algorithm. The seed node is selected by calculating the LSCR value of the node to reduce the number of iterations and merge small communities.

(3) A comparative experiment was conducted on the public datasets to verify the effectiveness and advantages of the LSCR and Louvain-LSCR algorithms proposed in this paper, and the Louvain-LSCR algorithm was used to conduct a case test of potential cooperative research team mining.

The rest of this paper organized as follows. In Section 2, we introduced the basis of the core node recognition algorithm and the Louvain algorithm and its flaws. In Section 3, the improved core node recognition LSCR algorithm and the new Louvain-LSCR algorithm are introduced. In Section 4, the evaluation coefficient of core node recognition algorithm and community detection is introduced. In Section 5, the experimental results of this paper’s algorithms are tested and compared. Finally, we draw conclusions and discussions in Section 6.

2. Basic Theory and Algorithm

2.1. Core Node Recognition Algorithm

2.1.1. Degree Centrality. The degree centrality of the node \( v_i \) in the network is the number of node directly connected to it. The calculation formula is as follows:

\[
D(v_i) = \sum_{j} A_{ij}.
\]

2.1.2. Betweenness Centrality. The betweenness centrality of node \( v_j \) in the network is the ratio of the number of shortest paths of any two nodes through node \( v_j \) to all the shortest paths:

\[
BC(v_j) = \sum_{i \neq j \neq k} \frac{\sigma(i, j | k)}{\sigma(j, k)}.
\]

Among them, \( \sigma(i, j | k) \) represents the number of shortest paths from node \( v_i \) to node \( v_k \) through node \( v_j \) and \( \sigma(j, k) \) represents the number of shortest paths from node \( v_j \) to node \( v_k \).

2.1.3. Closeness Centrality. The closeness centrality of node \( v_j \) in the network is the reciprocal of the sum of the shortest
distances from the node to other nodes, and its calculation formula is as follows:

$$CC(v_i) = \frac{1}{\sum_{j \neq i} d_{ij}}$$

(3)

where \(d_{ij}\) is the shortest distance from \(v_i\) to \(v_j\).

2.1.4. K-Index Centrality. Giovanni et al. proposed K-index [30]. First, compare the value of H-index [31, 32]; when the H-index is equal, the citations (the sum of citations of h core papers) are used to further distinguish the size of influence. The K-index calculation formula is as follows:

$$K = h + \left[ 1 - \frac{h^2}{N_{c,h}} \right].$$

(4)

Among them, \(N_{c,h}\) represents the sum of the citations of a certain author at \(h\), which corresponds to the sum of the neighbor node degrees of the node \(v_i\) in network \(G\) at \(h\).

2.2. Louvain Algorithm

2.2.1. Concept of Louvian Algorithm. Louvain algorithm is an aggregation community detection algorithm based on modularity. By continuously merging nodes, when the modularity no longer increases, the community structure obtained at this time is the final division result [18].

Louvain algorithm is divided into two stages, and constant iterations are required in both stages. In the first stage, the nodes in the network are allocated to different communities. At this time, the number of communities is equal to the number of nodes; then it is calculated by the modularity gain \(\Delta Q\) when the node is moved to the community to which its neighbor node belongs. For node \(v_i\), suppose that node \(v_i\) has \(j\) neighbor nodes, and these \(j\) neighbor nodes belong to \(k\) different communities. At this time, the change of modularity gain \(\Delta Q\), \((j = 1, 2, \ldots, k)\) before and after moving node \(v_i\) to the \(k\) communities to which its neighbor node belongs, in turn, is recorded. Set the maximum value of \(\Delta Q\), to max \(\Delta Q\), when max \(\Delta Q > 0\), move node \(v_i\) to the community relative to max \(\Delta Q\). When max \(\Delta Q < 0\), keep the community to which node \(v_i\) belongs unchanged, until all the nodes belonging to the community are not changing. In the second stage, each community obtained in the first stage is merged into a super point, and the weight of the edge between the super points is reset to the sum of the weight of the edge between the two communities in the first stage. Then divide this new network by the method of phase one, until the modularity \(Q\) value does not increase. The formula for calculating modularity gain is as follows:

$$\Delta Q = \left[ \frac{C_{in} + K_{i,in}}{2m} - \left( \frac{C_{tot} + K_i}{2m} \right)^2 \right] - \left[ \frac{C_{in}}{2m} - \left( \frac{C_{tot}}{2m} \right)^2 - \left( \frac{K_i}{2m} \right)^2 \right].$$

(5)

In the formula, \(m\) represents the total number of edges in the network; \(K_i\) is the sum of the weights of all the edges connected to node \(v_i\); \(C_{in}\) is the sum of the weights of the internal edges in the community \(C\); \(K_{i,in}\) is the sum of the weights of the edges connected to nodes in the community \(C\) and node \(v_i\); \(C_{tot}\) is the sum of the weights of all the edges connected to the nodes in the community \(C\); and \(\Delta Q\) is divided into two parts. The first part represents the modularity of community \(C\) after node \(v_i\) is divided into community \(C\), and the latter part is the sum of the modularity of community \(C\) and the modularity of node \(v_i\) when node \(v_i\) is an independent community.

2.2.2. Defects of Louvain Algorithm. In real networks, there are often many leaf nodes, especially for some networks with the obvious hierarchical structure, these leaf nodes can be processed through pruning technology to improve the efficiency of the algorithm. The leaf node is only connected to one other node. For the leaf node, it must be divided into the community where the connected node is located, which can avoid a lot of related calculations, thereby improving the efficiency of the algorithm. Generally speaking, there are leaf nodes during phase 1 division. In phase 2, after the community divided in phase 1 is compressed into a super point, there are no more leaf nodes [33].

In the Louvain algorithm, stage 2 will iterate repeatedly until \(Q\) no longer changes. The problems in phase 2 of the Louvain algorithm are as follows: (1) the operation efficiency of the algorithm decreases due to too many iterations and (2) there are too many small communities.

3. Improved Algorithm

3.1. LSCR Algorithm. The importance of a node in the network depends not only on the degree of the node itself, but also on the degree of dependence of the neighboring nodes on the node within two hops of the node. Inspired by Ruan et al. [34], from the LHN similarity [35] to measure the overlap degree of neighbor nodes in terms of topological structure, the formula is as follows:

$$\text{sim}(i, j) = \begin{cases} \frac{|N(i) \cap N(j)|}{k_ik_j} & \text{if } i \text{ and } j \text{ have no edge,} \\ 1 & \text{if } b \text{ and } c \text{ have edge,} \end{cases}$$

$$\text{LSC}(i) = \sum_{x, y \in N(i)} (1 - \text{sim}(x, y)),$$

(6)

where \(N(i)\) represents the set of neighbor nodes of node \(i\) and \(|N(i) \cap N(j)|\) represents the number of common neighbor nodes of node \(i\) and node \(j\). The larger the Local Similarity Centrality (LSC) value of a node, the lower the overlap between the node and the neighbor node in terms of topological structure.

In people’s daily life, due to differences in ability, personality, etc., there will be inconsistencies in opinions. The relationship between closeness and estrangement is distinguished based on the value of opinion, and not everyone can
communicate with each other, and people have a certain range of trust when communicating. Inspired by these factors, the ramp function is used to correct the above LSC value, and its calculation formula is as follows:

\[
R(x) = \begin{cases} 
  x, & \text{if } x > 0 \\
  0, & \text{otherwise,}
\end{cases}
\]

\[
\text{LSCR}(i) = \sum_{j \in N(i)} R(\text{LSC}(i) - \text{LSC}(j)).
\]

(7)

### 3.2. Louvain-LSCR Algorithm

The improved community detection algorithm Louvain-LSCR has two stages. In both stages, the weights between nodes are taken into consideration. Phase 1 is different from phase 1 of the Louvain algorithm in that it distinguishes which nodes are leaf nodes at the beginning and records the nodes connected to them. If the node is a leaf node during calculation, then the subsequent operations of the node are skipped. Finally, find the community of the node connected to the leaf node and move it in. After the algorithm passes through stage 1, the main community has formed. However, the quality of the division is not good enough, and there are too many communities, so the communities need to be further divided through algorithm stage 2. In algorithm stage 2, first compress the result of stage 1 division, and rebuild a new network \(G'\), and then the LSCR value of each node according to Algorithm 1, and then the seed nodes are selected by formula (8), and its set is

\[
\text{Seed set} = \{i \mid \text{LSCR}(i) > \text{LSCR}_{\text{aver}}\},
\]

(8)

where \(\text{LSCR}_{\text{aver}}\) represents the average importance value of the node. Finally, each node in stage 2 is divided into different communities, and then for non-seed node \(v_i\), the modularity gain \(\Delta Q\) of the community to which its neighbor node belongs is calculated. If its neighbor node communities contain seed nodes and the max \(\Delta Q > 0\) of these communities, the node \(v_i\) will be preferentially moved to the community corresponding to max \(\Delta Q\). Otherwise, the calculation will move it to max \(\Delta Q\) without the seed node community; if this max \(\Delta Q > 0\), then node \(v_i\) will be moved to the community corresponding to max \(\Delta Q\). If the above conditions are not met, the community to which node \(v_i\) belongs remains unchanged.

In the Louvain algorithm, the loop of phase 1 and phase 2 occupies most of the running time of the algorithm. It takes \(O(c \mid V)\) time complexity to traverse a node in the division process of phase 1, \(c\) is the time cost required to move a node, phase 2 adopts the idea in phase 1, and its complexity is the same as phase 1. The algorithm in this paper is improved based on the Louvain algorithm. The addition of pruning in stage 1 reduces the complexity of the division, improves the iterative process in stage 2, and reduces the number of iterations. Therefore, the algorithm in this paper is superior to Louvain algorithm in terms of operating efficiency.

### 4. Evaluation Criteria

#### 4.1. Core Nodes Evaluation Criteria

This paper uses different core node recognition algorithms to identify the core nodes and deliberately attacks the network through the core nodes. The maximum connected subgraph coefficient and network efficiency indicators are used to quantify the impact of deliberate attacks on the network structure and function, in order to evaluate the effectiveness of the core node recognition algorithm.

##### 4.1.1. Maximum Connectivity Coefficient

Connected subgraph refers to a subgraph of the network. In this subgraph, there is at least one path between any two nodes. For a nonconnected graph, it can be decomposed into two or more subgraphs, and the one with the largest number of nodes becomes the largest connected subgraph. After the network is attacked, its topology will also change, and the network is decomposed into multiple subgraphs. The formula for calculating the maximum connectivity coefficient is as follows:

\[
G = \frac{N'}{N}
\]

(9)

Among them, \(N'\) represents the number of nodes in the maximum connected subgraph of the network after the network is attacked and \(N\) represents the number of initial network nodes.

##### 4.1.2. Network Efficiency

Network efficiency is to examine the removal of nodes in the network and all their corresponding edges under a certain attack strategy, so that some paths in the network are interrupted and the shortest path between some nodes becomes larger. In turn, the average path length of the entire network becomes larger, which
Input: the adjacency matrix of the $G(V, E)$
Output: the result of community detection on $G(V, E)$
Step 1: initialize the network and assign each node to different communities.
Step 2: the leaf nodes in the network are pruned, and then execute stage1.
Step 3: compress each community obtained in stage 1 into a super point, and construct a new network graph $G'$, and then obtain the LSCR value of the node according to Algorithm 1 in $G'$, and then select the seed node according to formula (8).
Step 4: for non-seed node $v_i$, divide it according to the principle described above.
Step 5: repeat Step 4 until all the non-seed nodes $v_i$ in $G'$ belong to the community no longer change.
Step 6: output the community detection result $C_i$ and the algorithm ends.

Algorithm 2: Louvain-LSCR.

affects the connectivity of the network. The network efficiency is expressed as

$$\alpha = \frac{1}{N(N-1)} \sum_{i \in V, j \in E} A_{ij}$$  \hspace{1cm} (10)

Under a certain attack strategy, calculate the network decline ratio before and after the network is attacked to measure the robustness of the network. The calculation formula for the decline in network efficiency is as follows:

$$\mu = 1 - \frac{\alpha}{\alpha_0}.$$ \hspace{1cm} (11)

Among them, $\alpha$ represents the network efficiency after the network is attacked and $\alpha_0$ represents the original network efficiency. The larger the value of $\mu$, the weaker the robustness of the network after being attacked.

4.2. Community Detection Evaluation Criteria. To measure the results of community network division, Newman and Girvan proposed the concept of modularity ($Q$) [36]. Generally speaking, when modularity $Q \geq 0.3$, it means that the community structure of the network divided by the algorithm is obvious. When the modularity $Q$ value is closer to 1, it means that the quality of the community obtained by the algorithm is higher, and it also means that the community structure of the network is more obvious. The modularity is defined as follows:

$$Q = \frac{1}{2m} \sum_{ij} (A_{ij} - \frac{k_i k_j}{2m}) \delta(C_i, C_j)$$

$$m = \frac{1}{2} \sum_{ij} A_{ij}$$ \hspace{1cm} (12)

In the above formula, $k_i$ represents the sum of the weights of the edges connecting node $v_i$ and all other nodes. In an undirected graph, when the value of $A_{ij}$ is 0, it means that there is no edge between node $v_i$ and node $v_j$; otherwise, it means the connection weight between node $v_i$ and node $v_j$. $C_i$ represents the community to which node $v_i$ belongs, and $\delta(C_i, C_j)$ is a binary function. If $C_i$ and $C_j$ are equal, the function value is 1; otherwise, the function value is 0.

5. Algorithm Verification and Examples

Found by Cooperative Research Teams

The experiment in this article needs to verify the performance of the improved algorithm. The experimental running environment is as follows: processor Intel® Core™i5-4258U, CPU @2.40 GHz, memory 8G, programming language is Python, and programming environment is Visio Studio Code 1.48.2.

5.1. Algorithms Verification. The datasets used in the experiment are the Zachary Karate network, the Dolphins social network, the Polbooks network, the Adjnoun network, the Celegansneural network, and the WS small world network ($N = 500, K = 5, P = 0.5$). The basic parameters are shown in Table 1, where $N$ represents the number of network nodes, $M$ represents the number of network edges, $\langle K \rangle$ represents the average degree, and $\langle L \rangle$ represents the average shortest path length.

5.1.1. Experiment and Analysis of Core Node Recognition Algorithm. Based on the above network, this article compares the LSCR algorithm with the other four algorithms that also use local information. The four algorithms are Degree [37], LLS [34], K-index [30], and LC [38]. According to the ranking results of the five algorithms, nodes are removed by static attacks, and the maximum connected subgraph coefficients and network efficiency changes when the network is deliberately attacked, so as to evaluate the accuracy of the core node recognition algorithm.

Figure 1 reflects the impact of using different core node recognition algorithms to attack the network on the largest connected subgraph of the network. In Figure 1(a), the LSCR algorithm is slightly lower than the LLS algorithm only in a certain part, and the rest are in the optimal situation among these algorithms. In Figure 1(b), the LSCR algorithm is in the optimal situation in the top 23% and the bottom 50% and is slightly lower than the LLS algorithm in the middle part. In Figure 1(c), in the top 25%, the LLS, Degree, LC, and LSCR algorithms are not much different. In the latter part, the LSCR algorithm is better than the other four algorithms. In Figures 1(d) and 1(c), the LSCR algorithm in the second half reaches the minimum value faster than other algorithms. In
Table 1: Topological characteristics of the datasets.

| Datasets     | N   | M   | \(<K>\) | \(<L>\) |
|--------------|-----|-----|---------|---------|
| Karate       | 34  | 78  | 4.59    | 2.41    |
| Dolphins     | 62  | 159 | 5.13    | 3.36    |
| Polbooks     | 105 | 441 | 8.40    | 3.08    |
| Adjnoun      | 112 | 425 | 7.59    | 2.54    |
| Celegansneural | 297 | 2148| 14.46   | 2.46    |
| WS           | 500 | 1000| 4.0     | 4.97    |

Figure 1: Continued.
Figure 1(f), the LSCR algorithm is significantly better than the other four algorithms.

Figure 2 reflects the effect of using different core node recognition algorithms to delete network nodes on network efficiency. In Figure 2(a), the LSCR algorithm only performs slightly worse in a small part and is in the optimal situation in the rest. In Figures 2(b) and 2(c), the LSCR algorithm is the first to achieve a 100% reduction in efficiency, and the remaining algorithms have not reached 100% after removing 80% of the nodes. In Figures 2(d), 2(e), and 2(f), the LSCR algorithm is almost in the best situation, and its network efficiency drops the fastest, while the \( K \) algorithm performs the worst.

In summary, the LSCR algorithm is more capable of identifying the core nodes in the network than other algorithms, and it is feasible to apply it to the Louvain algorithm to select core nodes.

5.1.2. Louvain-LSCR Algorithm Experiment and Analysis.

In order to further illustrate the difference in modularity and number of communities of the Louvain-LSCR algorithm compared with other algorithms, the following will compare the Louvain-LSCR algorithm with SLPA [39], SCD [40], and ILouvain [27] algorithms. The results are shown in Table 2. The algorithm in this paper is better than the ILouvain and SLPA algorithms, but some of the modularity is lower than the SCD algorithm. The reason may be that the SCD algorithm detects too many communities. For example, on the Karate network, SCD gets 16 communities, and the remaining algorithms are 3 to 4.

5.2. Cooperative Research Team Found Examples.

The above experimental results show that the Louvain-LSCR community detection algorithm has better modularity than other algorithms and a more reasonable community structure, indicating that the improved algorithm is more reasonable for community detection than other algorithms. It is feasible to use the improved algorithm for team mining on social networks.

5.2.1. Data Source. In this paper, the author’s co-occurrence matrix is used to construct the co-author network, and the author’s published papers are used as the data source to construct the author’s co-occurrence matrix. The experimental data in this article are all from the papers included in CNKI “China Academic Journals Online Publishing Database” and “China’s Important Conference Papers Full-Text Database,” and the retrieval period is until March 2020. First, find an expert in mechanical engineering, collect the data of his published papers, and then collect the rest of the paper data through the “snowball” [41] method, then these thesis data are programmed to clean the original data, and finally 1042 valid records are obtained after manual screening. Using these paper data as the source of data analysis, a total of 963 authors were involved, and there were 3102 relationships between them. First, convert these collaborative papers into co-occurrence matrices, considering the influence of the weight of the edge on the division result, so the co-occurrence matrix in this article is not a binary, but a non-binary matrix and the number of collaborations is the weight of the edge. And because the cooperation between the authors of the paper is mutual, the co-author network is undirected. Among them, the nodes of the co-author network represent the authors in the paper. If there is cooperation between two authors, then the nodes represented by the two authors are connected by lines.

5.2.2. Analysis of Experimental Results. The Louvain-LSCR algorithm is compared and analyzed through experiments, and two evaluation indicators of modularity \( Q \) and number of communities are used to analyze the pros and cons of the algorithm [42]. What can reflect the quality of community
Figure 2: The change in network efficiency reduction rate $\mu$ after using different core node recognition algorithms to attack the network. (a) Karate. (b) Dolphins. (c) Polbooks. (d) Adjnoun. (e) Celegansneural. (f) WS.
detection is the modularity $Q$ and the number of communities. The larger the $Q$, the better the effect of community detection.

In order to evaluate the quality of community detection, the Louvain-LSCR algorithm is compared with LPA \cite{43}, SCD \cite{40}, and ILouvain \cite{27} algorithms. The experimental results are shown in Table 3. Two evaluation indicators are used to evaluate the pros and cons of the algorithms: $Q$ and number of communities. The larger the $Q$, the more compact the community structure and the more reasonable community detection results.

It can be known from Table 3 that the modularity of the Louvain-LSCR algorithm is greater than that of the other algorithms and the number of communities is smaller than that of the other algorithms. Compared with the ILouvain algorithm, $Q$ is relatively improved by 10.9%. Compared with its SCD and LPA, it has a more obvious improvement, and the obtained community structure is more compact. Comparing the ILouvain algorithm with the Louvain-LSCR algorithm in this paper, the result is shown in Figure 3. The ILouvain algorithm requires 11 iterations, while the Louvain-LSCR only needs 7 times to complete the algorithm. Due to the introduction of edge weights, the influence of edge weights on the division results will be considered when dividing, and a more compact community will be obtained. In the end, the Louvain-LSCR algorithm got 15 communities, and the ILouvain algorithm got 23. Further comparing the communities with less than 1% members, the result is shown in Figure 4. There are 5 in ILouvain and only 2 in Louvain-LSCR, which shows that the Louvain-LSCR algorithm can merge small communities in time. In summary, the community quality obtained by the algorithm in this article is better than the community quality obtained by the above algorithms.

### 5.2.3. Analysis of Cooperative Research Team

The author's co-authoring relationship is formed into a co-author network, which has 963 nodes and 3102 edges. Count the number of authors, average degree, and internal team connection of the 15 teams divided according to the Louvain-LSCR algorithm, respectively, and calculate the average value. The results are shown in Table 4.

Analysis of the structure of the co-author network found that there are close connections between certain nodes in the network, which causes these nodes to be combined into a small group. In a complex network, this feature of the network is called a community \cite{44, 45}. The community characteristics in the co-author network reflect the characteristics of teamwork between authors. The main purpose of community analysis in complex network analysis is to discover how many communities there are in the network. The community detection method can measure the strong, direct, and closely related communities in the network. Therefore, use Networkx \cite{46} to draw the co-author network, and then use the improved Louvain algorithm to conduct community detection on the co-author network. It can be found that there are 15 communities in close cooperation; that is, there are 15 potential cooperative research teams in the network, numbered 1–15, as shown in Figure 4(b). From the results of team mining, it is found that some research teams have a large number of people, and some research teams have a small number of people, because in the network, the research team does not have a unified size and clear boundaries \cite{47}. The 15 research teams further explore the relationships between the teams; the relationship is shown in Figure 5. It shows that the distribution of teams is obvious; they do not exist in isolation but have cross-team communication and cooperation with each other. In the entire network, there are nodes between teams that act as bridges, connecting these teams into a large cooperation network.

### Table 2: Comparison of algorithms experiment results.

| Evaluation indicator | Datasets  | Louvain-LSCR | SCD | SLPA | ILouvain |
|----------------------|----------|--------------|-----|------|----------|
|                      | Karate   | 0.4188       | 0.4085 | 0.3547 | 0.3614  |
|                      | Dolphins | 0.5112       | 0.5174 | 0.4986 | 0.4952  |
|                      | Polbooks | 0.5198       | 0.4357 | 0.4811 | 0.4833  |
|                      | Adjnoun  | 0.2730       | 0.3339 | 0       | 0.2584  |
|                      | Celegansneural | 0.3757       | 0.2922 | 0       | 0.3449  |

|                      | Karate   | 4            | 16   | 3     | 4        |
|                      | Dolphins | 6            | 25   | 6     | 5        |
|                      | Polbooks | 5            | 28   | 4     | 5        |
|                      | Adjnoun  | 6            | 46   | 1     | 6        |
|                      | Celegansneural | 6       | 88   | 1     | 6        |

### Table 3: Comparison of Louvain-LSCR and other algorithms results.

| Evaluation indicator | Louvain-LSCR | SCD | SLPA | ILouvain |
|----------------------|--------------|-----|------|----------|
|                      | Q            | 0.7565       | 0.3534 | 0.6468 | 0.6820  |
|                      | Number of communities | 15       | 425   | 112    | 23      |
Figure 3: Comparison of the number of iterations.

Figure 4: Proportion of team members. (a) Proportion of team members to ILouvain algorithm. (b) Proportion of team members to Louvain-LSCR algorithm.

Table 4: Team information of the co-author network.

| Team number | Number of team members | Internal team connection | Average degree within the team |
|-------------|------------------------|--------------------------|-------------------------------|
| 1           | 133                    | 349                      | 5.248                         |
| 2           | 128                    | 329                      | 5.141                         |
| 3           | 123                    | 323                      | 5.252                         |
| 4           | 95                     | 213                      | 4.524                         |
| 5           | 90                     | 263                      | 5.814                         |
| 6           | 80                     | 151                      | 3.675                         |
| 7           | 71                     | 164                      | 4.320                         |
| 8           | 56                     | 125                      | 4.454                         |
| 9           | 54                     | 133                      | 4.926                         |
| 10          | 50                     | 88                       | 3.520                         |
| 11          | 35                     | 81                       | 4.629                         |
| 12          | 20                     | 43                       | 4.300                         |
| 13          | 12                     | 19                       | 3.137                         |
| 14          | 9                      | 11                       | 2.444                         |
| 15          | 7                      | 9                        | 2.571                         |
| Average     | 64.2                   | 153.4                    | 4.290                         |
6. Conclusions

In order to explore the potential cooperative research teams in the co-author network, this paper proposes corresponding improvement measures to improve the detection effect of the cooperative research team in view of the deficiencies of the Louvain algorithm. First, in order to identify the core nodes in the network, an LSCR algorithm based on local information is proposed. By combining Louvain and LSCR algorithms, a Louvain-LSCR algorithm is proposed to mine teams in the network. It uses the LSCR algorithm to select seed nodes, which avoids the problems of too many iterations and too many small communities in phase 2 of the Louvain algorithm. In the experiments of the co-authored network and 5 other real networks and 1 artificial network, the LSCR algorithm is better than the four algorithms based on local information (LLS, Degree, LC, and K algorithms), while the Louvain-LSCR algorithm is better than the SCD, SLPA, and ILouvian algorithms.

The core nodes in the network play an important role in the transmission of information, such as finding influential users in the virtual community and protecting the core nodes to improve the robustness of the network. In addition, community detection can be applied to the knowledge graph to mine abnormal data, which can lay a foundation for building a knowledge graph or big data mining providing services.

However, this study only briefly analyzed the co-authoring network, and did not consider the order of authors and the influence of corresponding authors. In addition, it did not consider the time attenuation of cooperation. Therefore, building related models and solving algorithms for the above attributes is our next research direction.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this study.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant no. 71961005), Natural Science Foundation of Guangxi Zhuang Autonomous Region (Grant no. 2020GXNSFAA297024), and Guangxi Key Laboratory of Manufacturing System and Advanced Manufacturing Technology (Grant no. 20-065-40S002).

References

[1] T. Jin, Q. Wu, X. Ou, and J. Yu, “Community detection and co-author recommendation in co-author networks,” International Journal of Machine Learning and Cybernetics, vol. 12, no. 2, pp. 597–609, 2020.
[2] S. T. Bell, S. G. Brown, A. Colaneri, and N. Outland, “Team composition and the ABCs of teamwork,” American Psychologist, vol. 73, no. 4, pp. 349–362, 2018.
[3] E. B. Delia and J. D. James, “The meaning of team in team identification,” Sport Management Review, vol. 21, no. 4, pp. 416–429, 2018.
[4] S. T. Bell, S. G. Brown, and J. A. Weiss, “A conceptual framework for leveraging team composition decisions to build human capital,” Human Resource Management Review, vol. 28, no. 4, pp. 450–463, 2018.
[5] T. Jin, H. Ding, H. X. Xia, and J. F. Bao, “Reliability index and Asian barrier option pricing formulas of the uncertain fractional first-hitting time model with Caputo type,” Chaos, Solitons & Fractals, vol. 142, Article ID 110409, 2021.
[6] W. Deng, J. J. Xu, X. Z. Gao, and H. M. Zhao, “An enhanced MSIQDE algorithm with novel multiple strategies for global optimization problems,” IEEE Transactions on Systems, Man, and Cybernetics: Systems, pp. 1–10, 2020.
[7] D. Shang and M. Ghriga, “Examining the impacts of key influencers on community development,” Computers, Materials & Continua, vol. 61, no. 1, pp. 1–10, 2019.
[8] B. Shazad, H. U. Khan, Z. Rehman, and M Farooq, “Finding temporal influential users in social media using association rule learning,” Intelligent Automation and Soft Computing, vol. 26, no. 1, pp. 87–98, 2020.
[9] R. Hua, Y. Bao, S. Chen, and Z. Zhuang, “Optimization of the dynamic measure of spillover effect based on knowledge...
12 Scientific Programming

[10] B. Niu and Y. Huang, “An improved method for web text affective cognition computing based on knowledge graph,” Computer Systems Science and Engineering, vol. 34, no. 4, pp. 215–223, 2019.

[11] J. Wang, Y. Yang, T. Wang, R. Sherratt, and J. Zhang, “Big data service architecture: a survey,” Journal of Internet Technology, vol. 21, no. 2, pp. 393–405, 2020.

[12] W. Deng, S. F. Shang, X. Cai, and H Zhao, “Quantum differential evolution with cooperative coevolution framework and mutation strategy for large scale optimization,” Knowledge-Based Systems, vol. 224, Article ID 107080, 2021.

[13] M. Guerrero, R. Banos, C. Gil, F. G. Montoya, and A. Alcayde, “Evolutionary algorithms for community detection in continental-scale high-voltage transmission grids,” Symmetry, vol. 11, no. 12, p. 1472, 2019.

[14] X. Cai, H. M. Zhao, S. F. Shang, and Y Zhou, “An improved quantum-inspired cooperative co-evolution algorithm with multi-strategy and its application,” Expert Systems with Applications, vol. 171, Article ID 114629, 2021.

[15] M. Girvan and M. E. J. Newman, “Community structure in social and biological networks,” Proceedings of the National Academy of Sciences, vol. 99, no. 12, pp. 7821–7826, 2002.

[16] Z. Q. Miao and B. Balasundaram, “Approaches for finding cohesive subgroups in large-scale social networks via maximum k-plex detection,” Networks, vol. 69, no. 12, pp. 388–407, 2017.

[17] G. Li, M. F. Liu, Q. Wu, and J. Mao, “A research of characters and identifications of roles among research groups based on the bow-tie model,” Library and Information Service, vol. 61, no. 5, pp. 87–94, 2017.

[18] V. D. Blondel, J. L. Guillaume, R. Lambiotte, and E. Lefebvre, “Fast unfolding of communities in large networks,” Journal of Statistical Mechanics, vol. 2008, no. 10, pp. 1–13, Article ID 10008, 2008.

[19] A. Akbaritabar, V. A. Traag, A. Caimo, and F. Squazzoni, “Italian sociologists: a community of disconnected groups,” Scientometrics, vol. 124, no. 3, pp. 2361–2382, 2020.

[20] V. A. Traag, L. Waltman, and N. J. van Eck, “From louvain to Leiden: guaranteeing well-connected communities,” Scientific Reports, vol. 9, no. 1, pp. 1–12, Article ID 5233, 2019.

[21] V. L. Scibolazza, R. Vacca, T. K. Okraku, and C. McCarty, “Detecting and analyzing research communities in longitudinal scientific networks,” PLoS One, vol. 12, no. 8, pp. 1–23, Article ID e0182516, 2017.

[22] Y. H. Li, J. Q. Wang, X. I. Wang, and Y. L. Zhao, “Community detection based on differential evolution using social spider optimization,” Symmetry, vol. 9, no. 9, pp. 1–20, Article ID 183, 2017.

[23] H. M. Zhao, H. D. Liu, Y. Jin, X. J. Dang, and W. Deng, “Feature extraction for data-driven remaining useful life prediction of rolling bearing,” IEEE Transactions on Instrumentation and Measurement, vol. 70, Article ID 3511910, 2021.

[24] N. Ç. Polat, G. Yaylali, and B. Tanay, “A method for decision making problems by using graph representation of soft set relations,” Intelligent Automation & Soft Computing, vol. 25, no. 2, pp. 305–311, 2019.

[25] Z. F. Wu, P. F. Wang, Z. G. Qin, and S. Q. Jiang, “Improved algorithm of louvain communities partition,” Journal of University of Electronic Science and Technology of China, vol. 42, no. 1, pp. 105–108, 2013.

[26] B. Hu, W. Li, X. Huo, Y. Liang, M. Gao, and P. Pei, “Improving louvain algorithm for community detection,” in Proceedings of the 2016 International Conference on Artificial Intelligence and Engineering Applications, pp. 110–115, Hong Kong, China, November 2016.

[27] R. Q. Zhao, Y. Wu, and X. Chen, “An algorithm for large-scale social network community detection and visualization,” Journal of Computer-Aided Design & Computer Graphics, vol. 29, no. 2, pp. 328–336, 2017.

[28] Z. Zhang, P. Pu, D. Han, and M. Tang, “Self-adaptive Louvain algorithm: I,” Physica A: Statistical Mechanics and Its Applications, vol. 506, pp. 975–986, 2018.

[29] X. Li, J. Yang, Z. L. Sun, and J. P. Zhang, “Differentially private release of the distribution of clustering coefficients across communities,” Security and Communication Networks, vol. 2019, pp. 1–9, Article ID 2518714, 2019.

[30] G. Anania and A. Caruso, “Two simple new bibliometric indexes to better evaluate research in disciplines where publications typically receive less citations,” Scientometrics, vol. 96, no. 2, pp. 617–631, 2013.

[31] L. Lü, T. Lü, Q. M. Zhang, and H. E. Stanley, “The H-index of a network node and its relation to degree and coreness,” Nature Communications, vol. 7, Article ID 10168, 2016.

[32] J. E. Hirsch, “An index to quantify an individual’s scientific research output,” Proceedings of the National Academy of Sciences of the United States of America, vol. 102, no. 4, pp. 16569–16572, 2005.

[33] L. Zheng, “Community discovery based on local incremental hyper point louvain pruning technique,” Control Engineering China, vol. 24, no. 10, pp. 2184–2188, 2017.

[34] Y. R. Run, S. Y. Lao, J. D. Wang, L. Bai, and L. D. Chen, “Node importance measurement based on neighborhood similarity in complex network,” Acta Physica Sinica, vol. 66, no. 5, pp. 383–392, 2017.

[35] E. A. Lightweight, P. Holme, and M. E. Newman, “Vertex similarity in networks,” Physical Review E - Statistical, Nonlinear and Soft Matter Physics, vol. 73, no. 2, Article ID 026120, 2006.

[36] M. E. Newman and M. Girvan, “Finding and evaluating community structure in networks,” Physical review. E, Statistical, Nonlinear, and Soft Matter Physics, vol. 69, no. 2, pp. 1–16, Article ID 026113, 2004.

[37] L. C. Freeman, “Centrality in social networks conceptual clarification,” Social Networks, vol. 1, no. 3, pp. 215–239, 1978.

[38] D. Chen, L. Lü, and M.-S. Shang, “Identifying influential nodes in complex networks,” Physica A: Statistical Mechanics and Its Applications, vol. 391, no. 4, pp. 1777–1787, 2012.

[39] G. Cordasco and L. Cargano, “Community detection via semi-synchronous label propagation algorithms,” in Proceedings of the 2010 IEEE International Workshop on: Business Applications of Social Network Analysis (BASNA), Bangalore, India, Dec. 2010.

[40] A. Prat-Perez, D. Dominguez-Sal, and J. L. Larriba-Pey, “High quality, scalable and parallel community detection for large real graphs,” in Proceedings of the 23rd international conference on World Wide Web, pp. 225–236, Seoul, Korea, April 2014.

[41] M. Bailey, “Snowball sampling in business oral history: accessing and analyzing professional networks in the Australian property industry,” Enterprise and Society, vol. 20, no. 1, pp. 74–88, 2019.
Humanized Computing, vol. 2020, pp. 1–11, Article ID 7307, 2020.

[43] U. N. Raghavan, R. Albert, and S. Kumara, “Near linear-time algorithm to detect community structures in large—scale networks,” Physical Review E, vol. 76, no. 3, pp. 1–11, Article ID 036106, 2007.

[44] J. Zheng, J. Gong, and R. Li, “Community evolution analysis based on co-author network: a case study of academic communities of the journal of Annals of the Association of American Geographers,” Scientometrics, vol. 113, no. 2, pp. 845–865, 2017.

[45] R. Khalid, N. Idrees, and M. Jawwad Saif, “Topological characterization of book graph andstacked book graph,” Computers, Materials & Continua, vol. 60, no. 1, pp. 41–54, 2019.

[46] A. H. Aric, A. S. Daniel, and J. S. Pieter, ”Exploring network structure, dynamics, and function using NetworkX,” in Proceedings of the 7th Python in Science Conference (SciPy2008), pp. 11–15, Pasadena, August 2008.

[47] W. Fan, G. Li, and R. Law, “Analyzing co-authoring communities of tourism research collaboration,” Tourism Management Perspectives, vol. 33, pp. 1–11, Article ID 100607, 2020.