Hindawi
Mobile Information Systems
Volume 2021, Article ID 2931801, 10 pages
https://doi.org/10.1155/2021/2931801

Research Article

Community Detection Algorithm Based on Intelligent Calculation of Complex Network Nodes

Yanjia Tian [1,2] **and Xiang Feng** [1,3]

[1]*Department of Computer Science and Engineering, East China University of Science and Technology, Shanghai 200237, China*
[2]*School of Electronics and Information, Shanghai DianJi University, Shanghai 201306, China*
[3]*Shanghai Engineering Research Center of Smart Energy, Shanghai 200237, China*

Correspondence should be addressed to Xiang Feng; xfeng_ecust@163.com

Received 30 August 2021; Revised 21 October 2021; Accepted 26 October 2021; Published 13 November 2021

Academic Editor: Han Wang

With the explosive development of big data, information data mining technology has also been developed rapidly, and complex networks have become a hot research direction in data mining. In real life, many complex systems will use network nodes for intelligent detection. When many community detection algorithms are used, many problems have arisen, so they have to face improvement. The new detection algorithm CS-Cluster proposed in this paper is derived by using the dissimilarity of node proximity. Of course, the new algorithm proposed in this article is based on the IGC-CSM algorithm. It has made certain improvements, and CS-Cluster has been implemented in the four algorithms of IGC-CSM, SA-Cluster, W-Cluster, and S-Cluster. The result of comparing the density value on the entropy value of the Political Blogs data set, the DBLP data set, the Political Blogs data set, and the entropy value of the DBLP data set is shown. Finally, it is concluded that the CS-Cluster algorithm is the best in terms of the effect and quality of clustering, and the degree of difference in the subgraph structure of clustering.

1. Introduction

Complex systems in social life and nature can be intelligently detected by network nodes. When the community algorithm can solve many problems in life, it has been widely used, and it has also promoted the improvement of community detection in complex network systems. A study about directed network module found that its value will penetrate and change [1], and the directed network module overlaps, and the overlap center includes two aspects, inward and outward. Revealing the structure in the community is one of the key issues in the study of complex networks [2]; that is, the node group formed by the module and the community will form a close unit, but the connection between the units is very weak. The current research has found that the network connection partition allows the community to overlap at the node [3], and the connection partition can be generated by the algorithm of multiple node partitions through overlap and through the line chart to show the role of degree heterogeneity. Combined with network dynamics, a new detection community algorithm is proposed [4], through two principles to explain the new algorithm, and used for the detection of strong overlapping communities. Moreover, researchers analyze the real society based on the research of overlapping network structures [5] and then propose a novel general framework to detect the entire community. According to the research, it is found that SLPA has excellent performance in identifying nodes and overlapping communities of different degrees. An algorithm is also proposed, which is very practical and can find overlapping communities in a huge network [6]. The same as the traditional algorithm is that the vertices have labels spread between the same vertices. The main contribution of this paper is to expand the labeling and propagation steps, and this algorithm is very effective in restoring the community. Researchers have found that it is necessary to verify the information when identifying the community structure of the large-scale network real world [7]; in the process of

replacement, densely connected nodes form a consensus on a unique label, and a community is generated. At present, researchers use the new community detection algorithm in the study to delete the high-different unconnected node subgroups in the splitting process [8]. To provide a general framework for the realization of this method, the application of this method to computer-generated networks and different real-world networks shows the effectiveness of this method. The identification of building blocks is very important for understanding the network structure and functional characteristics [9]. Use new technologies to find overlapping communities on a large scale, and define new features through community statistics. Based on the studied topological characteristics [10], researchers proposed a new penetration algorithm SCP, which performs rapid community detection in selected modules of weighted and unweighted networks. With recognition network being used more and more [11], it has led to the lack of feature consensus in overlapping communities, so the previous algorithm scan was modified accordingly to solve some attribute defects. Furthermore, researchers have found a completely solvable social network model [12] that provides models for simple one-party and two-party networks and also compares the prediction results of the model with social networks in the real world. The hidden relationship is revealed by the complex network [13], but the detection method at this time is unstable, and the result is affected by many factors. In the case of self-consistent combination with any method, the stability and accuracy of the partition result will be greatly improved. A study proposes a simple and intuitive network cohesion method [14], which can be implemented in a few lines of code. According to research, node separation can achieve quite obvious effects on the best graph and clustering method in terms of quality and efficiency. The network community structure algorithm proposed in this paper densely divides connected subgroups [15]. After a series of calculations, it is proved that this method also has a significant effect on network data in the real world research on the community detection algorithm based on the above-mentioned complex network intelligence.

With the increasing maturity of data mining technology [16], and the wide application of complex networks in various fields, clustering technology has been developed day by day. Analyzing the community structure in complex networks can not only make researchers have a deeper understanding of the structural and performance characteristics of nodes in complex networks, but also help find the evolution rules of complex networks.

2. Basic Community Detection Algorithm

Nowadays, with the advent of the era of big data explosion, data mining technology has made rapid progress, and complex networks have become a hot research direction in the field of data mining. A community detection method is proposed [17]; it is a hierarchical clustering method. It uses the random walk model to calculate the dissimilarity of nodes in the network to complete the division of the network. Radicchi et al. used the idea of edge-based clustering coefficient to complete the identification of community structure [18]; this method first calculates the clustering coefficient of edges, then selects lower-value edges from them, and then gradually deletes them to identify the community. The advantage of this algorithm is that it can be successfully applied to large networks. A community detection algorithm (GN algorithm) is proposed in 2002 [19, 20]; this algorithm is an algorithm based on the idea of splitting. It initially calculates the betweenness values of the edges contained in the network and keeps updating the betweenness values of the remaining edges in the network until all edges in the network are deleted, thereby obtaining community structure. A network community detection algorithm based on minimum spanning tree and modularity is proposed [21]; it realizes community detection through a hierarchical idea. In addition, a detection algorithm that recognizes communities based on the idea of tag propagation is proposed [22]; this algorithm initially defines a label for all nodes in the graph and then sequentially replaces the label of node i with the label owned by most of the neighboring nodes around it. And a new algorithm is to minimize the number of edges that exist outside the community to divide the network graph [23] and finally complete the recognition of the community, but the algorithm must know the number of communities in advance. Furthermore, a hierarchical clustering algorithm is developed [24]; according to the rule of information centrality, this method uses information centrality to detect edges in communities and gradually remove the edge with the highest value of information centrality in the graph to identify the community structure. A detection algorithm, NMF algorithm, is also proposed [25], which uses the overlap of the Laplacian matrix of a given network to realize community recognition.

In this chapter, we introduce the degree of node dissimilarity-node proximity. Based on the dissimilarity, a new clustering algorithm, CS-Cluster, is proposed. This algorithm comprehensively considers the topological structure and semantic information of the node to compare complete clustering process with high efficiency. Compared with other algorithms, this algorithm has the following characteristics:

(1) Use the structural and semantic characteristics of the node to calculate the proximity of the node;

(2) According to the different connection modes of nodes, the contribution degree and the matching degree are, respectively, added to participate in the calculation of the structural dissimilarity;

(3) The algorithm can automatically determine the initial clustering center point, which improves the accuracy of clustering and reduces the inaccuracy caused by human subjective judgment, because the selection of different initial cluster centers will have a greater impact on the final clustering results.

In the network graph to be explored in this chapter, the number of semantic types contained in nodes is constant,

and it is impossible for the number of semantic types to be inconsistent. An undirected weighted graph with semantic information can be represented $G = (V, E, W, \Gamma)$, which represents$V = \{v_1, v_2, \ldots, v_n\}$ as the collection of all nodes in the graph; E represents the set of all connected edges in the graph. Each edge in the edge set E of the network corresponds to two nodes in the node set V in the graph $E = \{e_1, e_2, \ldots, e_m\}$, that is, $\{e_1, e_2, \ldots, e_m\} \subseteq V \times V; W$, the weight set of the undirected edges in the graph, The weights of the connecting edges of nodes v_m and nodesv_n represent the set of semantic information (attributes) of the nodes $\omega_{mn} > 0; \Gamma = \{s_1, s_2, \ldots, s_n\}$ in the graph; the value set of the node v_m on the semantic information Γ can be expressed as$\{s_1(v_m), s_2(v_m), \ldots, s_k(v_m)\}$. The total number of nodes $v_k$directly connected to the node is called the degree of this node $d(v_k)$.

The goal of graph clustering algorithm is to divide a large graph into several closely connected and disjoint subgraphs (i.e., communities in the following) through the clustering process and should satisfy the following:

(1) The internal nodes of the same community are in the structure the above is closely connected, and the object connection between different communities is relatively sparse;

(2) The semantic features of the nodes in the same community are similar, but in different communities, the semantic information of the nodes is different.

3. Calculation and Replacement of Community Test Algorithm

3.1. Calculation of Node Proximity

3.1.1. Relevance and Matching. This paper proposes a community detection algorithm based on the proximity of nodes. The closeness of the relationship between nodes depends on the correlation between nodes. It shows the characteristics of information transfer between nodes. For interconnected nodes, there may be multiple paths between the initial node and the target node. The IGC-CSM algorithm is proposed and considers the selection of weighted connecting edges [26]. The path with the smallest weight participates in the calculation of structural dissimilarity instead of all paths participating in the calculation, which reduces the amount of calculation. The algorithm proposed in this chapter introduces the correlation degree and the matching degree on the basis of the IGC-CSM algorithm to complete the calculation of structural dissimilarity.

Definition 1. (degree of association). In an undirected network graph with weights, a pair of directly connected nodes isvm and$v$. The degree of connection intimacy is defined as the degree of association. Then, the calculation formula for the degree of association between the node vmandvn the node is $H(vm, vn)$ as follows:

$$T(v_m, v_n) = 1 - e^{-\left(\left(\omega_{mn}/\sum_{k-1}^{d(v_m)}\omega_{ma_k}\right) * d(v_m)\right)}, \quad v_m \leftrightarrow v_n,$$

$$H(v_m, v_n) = \frac{T(v_m, v_n) + T(v_n, v_m)}{2}, \tag{1}$$

which defined $T(v_m, v_n)$ as the correlation coefficient between v_m nodes v_n. It indicates the degree to which the node is associated with v_m the node v_n:$d(v_m)$represents the degree of the node v_m, and ω_{mn} represents the weight of the edge connecting the node v_m and the node v_n. It represents the sum of the weights of $\sum_{k=1}^{d(v_m)}\omega_{ma_k}$, all the edges connected to the nodev_m; because the degrees $v_m$of the nodes $v_n$are different, the weights on the respective connected edges are not the same, so $T(v_m, v_n) \neq T(v_n, v_m)$. For example, in the DBLP network, an author may be associated with many authors. These authors are other authors who are associated with them, and the correlation coefficients of these authors are therefore different. The larger the value $H(v_m, v_n)$, the higher the degree of association between the two nodes, and the closer the relationship between the two nodes. Suppose that there are two directly connected nodes v_m and v_n, $d(v_m)$ and $d(v_n)$ mean $v_m$sum, respectively; v_n, w_{mm} represent the node v_m and the node v_n, the weight on the connecting edge. Then, the attraction factors v_m and v_n of the sum $f(v_m, v_n)$ are calculated as follows:

$$f(v_m, v_n) = \ln\left(1 + \frac{d(v_m)}{\sum_{j=1}^{d(v_m)} w_{mj}} * w_{mn}\right), \quad v_m \leftrightarrow v_n. \tag{2}$$

Definition 2. (matching path). A pair of indirectly connected nodes v_m, from the source node through several intermediate nodes v_n to the end point, a path passed is called a matching path $OP(v_m, v_k, \ldots, v_n)$, and all nodes pass only once on the path.

$$R(v_m, v_n) = \sum_{i-\text{srcnode}}^{\text{desnode}} f(v_i, v_{i+1}), \quad v_m \Theta v_n. \tag{3}$$

which represents *sno* as the initial node, *dno* represents the end node, and $T(v_i, v_j)$ represents the correlation coefficient between the two directly connected nodes. In the undirected weighted network graph, there may be multiple matching paths in the path from the initial node v_m to the end point v_n. We choose the path with a high degree of matching as the best path from the initial node v_m to the end point v_n.

3.1.2. Structural Dissimilarity. The calculation method of the adjacency between nodes maps the relationship between the nodes. An undirected weighted network graph is composed of the structural characteristics and semantic information of the nodes, so when calculating the proximity of the nodes, it should be composed of the structural characteristics and semantic characteristics of the nodes., And add the concept of relevance and matching in the calculation process of structural dissimilarity. We decided to

use the Jaccard correlation coefficient to calculate the basic structural dissimilarity. The formula is as follows:

$$D(A,B) = \frac{|A \cap B|}{|A| + |B| - |A \cap B|},$$
$$\text{sim}(X,Y) = \frac{|X \cap Y|}{|X \cup Y|}. \tag{4}$$

The IGC-CSM algorithm uses the weighted Jaccard correlation coefficient for calculation. Due to the differences in the connection methods between nodes, the corresponding structural dissimilarity calculation methods in the IGC-CSM algorithm are also different. The specific calculation formula is as follows:

(1) When two nodesv_m are directly connectedv_n, the calculation method is expressed as

$$D(v_m, v_n)_{\text{struct}} = \frac{\omega_{mn}}{\sum_{c=1}^{dc\ g} \omega_{mc} + \sum_{c=1}^{dc\ g(v_n)} \omega_{nc} - \omega_{mn}}, \quad v_m \leftrightarrow v_n. \tag{5}$$

(2) For the sum of two indirectly connected nodes v_m and v_n, the IGC-CSM algorithm is calculated in the form of the product of the directly connected node sequence pair. The specific calculation method is as follows:

$$D(v_m, v_n)_{\text{struct}} = \prod_{sno}^{dn\ o} D(v_j, v_k), \quad v_j \in V \text{ and } v_m \Theta v_n. \tag{6}$$

And its characteristic inspection method:

$$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta(C_i, C_j). \tag{7}$$

Among them, (v_j, v_k) represents a pair of directly connected nodes. Since there are multiple paths in the process from the initial node v_m to the target node v_n, IGC-CSM selects the path with the least weight and does not take all paths into the calculation, which greatly simplifies the amount of calculation increased, and the efficiency of calculation is improved.

Among them, m is the total number of edges in the network, k_i and k_j the degree of the node i and j respectively, C_i and C_J the node and the community, respectively; if $C_i = C_j$, then $\delta(C_i, C_j) = 1$, and the value is 0. In a specific network structure, the result of the division of the community is not the same, so the obtained modularity values are also different; if the modularity is larger, the algorithm partition is larger.

(3) When the two nodes are relatively independent, the structural dissimilarity is 0.

After the concepts of relevance and matching are introduced, the structural dissimilarity and semantic dissimilarity of the nodes in the graph can be calculated. The calculation method is based on the IGC-CSM algorithm to add the correlation degree and matching degree of the node. The calculation of structural dissimilarity can be divided into three cases according to the connection mode of the nodes: (1) For two directly connected nodes, the dissimilarity calculation is completed by combining the weighted Jaccard correlation coefficient and the correlation between the nodes. (2) For two indirectly connected nodes, the structural dissimilarity and the matching degree between the nodes in the IGC-CSM algorithm are used to complete the calculation of the structural dissimilarity. In the path between the node and the indirect connection, there may be more matching paths; we choose a path with a high degree of matching as the best path between the initial node and the end to participate in the calculation. (3) For unconnected nodes (independent nodes), the structural dissimilarity in the IGC-CSM algorithm is 0. The calculation formula is as follows:

$$D(v_m, v_n)_{str} = \begin{cases} D(v_{mv}, v_n)_{con} + H(v_{m,}, v_n), & v_m \leftrightarrow v_n, \\ D(v_m, v_n)_{in\ di\ r} + R(v_m, v_n), & v_m \Theta v_n, \\ 0, & v_m \otimes v_n. \end{cases} \tag{8}$$

Among them, $D(v_m, v_n)_{\text{sem}}$ means the degree of structural dissimilarity, $\longrightarrow$ means that two nodes are directly connected, and Θ means that the source nodes v_m and v_n are indirectly connected to the end point, which means that the node has no connected edges and belongs to an independent node.

The structural similarity calculation of indirect nodes is realized through the product of the node sequence on the path, and the calculation method is

$$\text{sim}(v_m, v_n)_{\text{indirconn}} = \prod_{i-\text{srcnode}}^{\text{desnode}} \text{sim}(v_i, v_{i+1}) \text{connected}, \quad v_m \Theta v_n, v_i \in V. \tag{9}$$

Among them, (v_i, v_{i+1}) is the sequence pair meaning two points v_m and v_n directly connected. There may be multiple paths from the source node to the node, but IGC-CSM uses the shortest weighted path strategy to avoid a large number of calculation processes.

In the process of graph clustering according to the dissimilarity of nodes, if only the structural features between nodes are concerned, the result of the division may be inaccurate, because, in real life, a node may carry multiple semantic information. For example, in a social network, a

user can perform different roles in different places. In addition, it may also include attributes such as occupations and hobbies. Therefore, when we compare node dissimilarity, we should also consider their semantic dissimilarity. A node contains multiple semantic information, and each semantic may take different values. We use the method of calculating dissimilarity in the K-Modes algorithm as the semantic similarity between nodes. For the convenience of calculation, we set the semantic weight to 1.

$$D(v_m, v_n)_{\text{sem}} = \begin{cases} \sum_{i=1}^{j} \dfrac{\text{com}(v_m, v_n, \text{sem}) * \omega_{\text{sem}}}{j}, & v_m \leftrightarrow v_n \text{ or } v_m \otimes v_n, \\ \sum_{i=\text{sno}}^{\text{dno}} D\left(v_i, v_j\right)_{\text{sem}}, & v_m \otimes v_n, \end{cases} \tag{10}$$

$$\text{com}(v_m, v_n, \text{sem}_l) = \begin{cases} 1, & \text{the node } v_m \text{ and } v_n \text{ the } l \text{ semantic value are equal.} \\ 0, & \end{cases}$$

Table 1 shows the meaning of symbols in complex network structure.where $D(v_m, v_n)_{\text{sem}}$ represents the degree of semantic dissimilarity, and ω_{sem} is the semantic weight; $j$represents the number of semantics; when the semantic value is the same, the value is 1; when the semantic value is different, the value is 0; so the semantic value range is [0.1].

3.1.3. Node Proximity. The calculation of similarity between nodes should comprehensively consider the structural characteristics of nodes and the semantic characteristics between nodes, so we draw the proximity of nodes to complete the calculation of dissimilarity between nodes. The formula is shown in (13):

$$D(v_m, v_n) = \lambda\, D\left(v_{m,} v_n\right)_{\text{str}} + (1-\lambda) D(v_m, v_n)_{\text{sem}}. \tag{11}$$

Among them, λ is the balance factor, which aims to adjust the proportion of structure and semantics in the calculation of dissimilarity. Its value range is [0, 1]; $D(v_m, v_n)_{\text{str}}$ represents the structural dissimilarity; $D(v_m, v_n)_{\text{sem}}$ represents the semantic dissimilarity. In the later part of the experiment, this chapter will give the most suitable value λ.

When the value $y_i$is larger, it will be more likely to become the cluster center; when the noncluster center transitions to the cluster center, the value will undergo a jump change, so we define a transfer ϕ_i function to indicate that the jump has found that the specific calculation formula of the clustering center point in the process is as follows:

$$\phi_i = \left|\gamma_{i-1} + \gamma_{i+1} - 2\gamma_i\right|. \tag{12}$$

It can be seen from the formula that the greater the value ϕ_i, the greater the change in the value of the node γ_i; then, the node ϕ_i should become the initial cluster center, so the results are sorted in descending order, and the point with the largest value z is taken as ϕ_i the initial cluster center point.

After calculating the node proximity, we use the node distance to complete the clustering process. Since the distance is inversely proportional to the proximity, that is, the greater the proximity, the closer the relationship between the nodes, and the smaller the corresponding distance value, so we use the inverse of the proximity to calculate the distance. The calculation formula is as follows:

$$\text{distance}(v_m, v_n) = \frac{1}{D(v_m, v_n)}. \tag{13}$$

According to the algorithm framework in this chapter to achieve clustering, the calculation formula is as follows:

$$f(w, s) = \sum_{l=1}^{k} \sum_{j=1}^{n} \omega_{lj}\, \text{distance}\left(s_l, v_j\right), \quad 1 \le l \le k \text{ and } 1 \le j \le n, \tag{14}$$

which represents the adjacency matrix of order. At that time, the representative node belongs to the first subcommunity; at that time, it means that the node does not belong to the community. It represents a collection of cluster center points.

The selection of the cluster center point should meet the following two criteria:

(A) The local density of the center point should be large enough

(B) The centers of different subassociations are far apart from each other

Regarding the local density definition of the node, the calculation formula can be expressed as

$$\partial_i = \sum_{v_m \in V/(v_n)} e^{-\left(\text{distance}(v_m, v_n)/d_c\right)}. \tag{15}$$

Among them, d_c is the cutoff distance, v_m is its value and is the average value of the distances from other nodes. $V/\{v_n\}$ represents other nodes except for v_n. It can be seen from the definition that the greater the number of nodes v_m whose distance to the node d_c is smaller, the larger the value ∂_i.

3.2. Iterative Update of the Algorithm. The iterative update steps of the CS-Cluster algorithm are as follows, in which $s_{(l)}$, $w_{(l)}$, respectively, represent the cluster center point and adjacency matrix at the first update.

Table 1: Common symbols in this chapter.

Symbol	Meaning
$\longrightarrow$	Represents two nodes directly connected
Θ	Indicates that the source node is indirectly connected to the end point
$\otimes$	Indicates that the node has no connected edges
ω_{sem}	Represents semantic weight
j	Represents the number of semantics
$D(v_m, v_n)_{\text{str}}$	Indicates the degree of structural dissimilarity
$D(v_m, v_n)_{\text{sem}}$	Represents the degree of semantic dissimilarity

(1) The point z with $\phi_i$the largest previous values$_{(l)}$ is calculated by formula (8) as the initial cluster center point$w_{(l)}$, and the adjacency matrix is calculated$f(w_{(l)}, s_{(l)})$ according to formula (9)

(2) When w does not change, calculates$_{(l+1)}$ and obtain the minimum value$f(w_{(l)} \cdot s_{(l+1)})$ according to formula (12); if it is$f(w_{(l)}, s_{(l+1)}) = f(w_{(l)}, s_{(l)})$, the algorithm iteration ends; otherwise, perform step (3)

(3) When s does not change, calculate $w_{(l+1)}$ and obtain $f(w_{(l+1)} \cdot s_{(l)})$, the minimum value according to formula (9). If it is$f(w_{(l+1)}.s_{(l)}) = f(w_{(l)}, s_{(l)})$, the algorithm update ends; otherwise, proceed to step (2)

When s does not change, the update method$w$ is as shown in formula (9), where the value range is $[1, z]$ and the value range is $[1, n]$.

$$w_{(lj)} = \begin{cases} 1, & \text{distance}\left(s_l, v_j\right) \leq \text{distance}\left(s_{\phi_i}, v_j\right), \\ o, & \text{Others.} \end{cases} \tag{16}$$

When w is not changed, the update method s is as follows:

$$s_l = v_l. \tag{17}$$

During the execution of the algorithm, the cluster center point will be updated after each iteration, and finally the remaining nodes will be divided into the nearest category with higher density from each cluster center.

The pseudocode of the clustering algorithm CS-Cluster mentioned in this chapter is shown in Algorithm 1:

where the balance factor$\lambda = 0.6$. In this paper, different values of balance factor are applied to the data set to observe the influence of balance factor on the density and entropy. When the balance factor is 0.6, there are higher density value and lower entropy value. See Section 4.3 for details.

4. Experimental Verification

In this experiment, five algorithms are used as comparison algorithms, namely, IGC-CSM, SA-Cluster, W-Cluster, S-Cluster, and CS-Cluster. Among them, IGC-CSM, SA-Cluster, and W-Cluster are three algorithms in the cluster. The topological structure of the node and the semantic information of the node are comprehensively considered in the class process. The IGC-CSM algorithm uses the K-Medoids framework to achieve clustering, and the SA-Cluster uses random walk technology. The iteration process is very time-consuming. The W-Cluster algorithm passes the weight function used to calculate the degree of dissimilarity between nodes. The S-Cluster algorithm also uses random walk technology, but the algorithm only considers the structural characteristics of the node. The CS-Cluster algorithm is the algorithm proposed in this chapter. The concept of correlation and matching is given in the algorithm to complete the calculation of the structural dissimilarity between the nodes in the graph and integrate the structural dissimilarity of the nodes; the degree of semantic dissimilarity calculates the proximity of nodes, redefines the selection of the initial clustering center point, and finally realizes the division of communities.

4.1. Data Set. This chapter uses two classic data sets to verify the effectiveness of the algorithm.

(1) Political blogs data set: this data set is composed of links between different blogs. For each blog contained in the data set, there is a semantic meaning to describe political orientation, with 0 for members of the Liberal Party and 1 for the Conservative Party member. The data on political orientation comes from blog directories.

(2) DBLP data set: we used one of the sub-data-sets-author collaboration network; we created a graph that reflects the cooperative relationship between authors. In addition, we added two phases to each node included in the network graph.

Related semantic information: the information is reflected by the number of evaluation indicators and themes of the papers. There are three optional values for the number of papers, which are less than 10, between 10 and 20, and greater than or equal to 20, to determine whether the author is "prolific." The other semantic is the topic. For topic attributes, we obtained the paper titles of the selected authors and organized these paper titles into documents. Finally, we extracted research topics containing 100 possible values from the documents, which were randomly assigned the keywords that represent each topic. The theme attributes of each author will correspond to these 100 themes in Table 2.

4.2. Evaluation Index. The comparison algorithm we used evaluates the results of clustering based on density and entropy, so this chapter also uses density and entropy to measure the effect of clustering.

(1) Density: It is defined as the ratio of the sum of edges in the subnetwork to the edges in the entire network.

Input: an undirected weighted graph containing attributes G, Number of clusters$z$;
Output: get a category $M_1, M_2, \ldots, M_z$;
Step 1: initialize the distance of each node (v_m, v_n) in the graph distance $[V_m][V_n] = 0$
Step 2: cluster center point centroid = 0, Semantic extremum $\omega_{\text{sem}_1}, \omega_{\text{sem}_2}, \ldots, \omega_{\text{sem}_l} = 1$, Balance factor $\lambda = 0.6$
Step 3: each node included in the set of all nodes V in the For graph
Step 4: each node v_n and $v_m \neq v_n$ contained in the set of all nodes in the For graph
Step 5: calculate the proximity of nodes according to formula (8)
Step 6: calculate the node distance according to formula (8)

Algorithm 1: The pseudocode of the clustering algorithm.

The higher the density, and the clustering of communities, the better the division effect for the clustering of communities, and the formula is as follows:

$$\text{Density}\left(\{V_c\}_{c=1}^{k}\right) = \sum_{c=1}^{k} \frac{\left|\left\{\left(v_{m,} v_n\right)\right\}\right| v_m, v_n \in V_c, \left(v_m, v_n\right) \in E}{|E|}, \tag{18}$$

where $E = \{e_1, e_2, \ldots, e_m\}$ represents the set of connected edges of all nodes in the graph, and m represents the total number of edges.

(2) Entropy: It is a measure of the semantic similarity between nodes, when the semantic similarity between nodes is a category.

The higher the value, the smaller the entropy value in the subnetwork, and the better the result of community division. Its definition is shown in formulas (19) and (20):

$$\text{Entropy}\left(\{V\}_{cc=1}^{k}\right) = \sum_{i=1}^{m} \left(\frac{w_i}{\sum_{s=1}^{m} w_s} \sum_{c=1}^{k} \frac{|V_C|}{|V|}\right) \text{entropy}, \tag{19}$$

$$\text{entropy}\left(a_i, V_c\right) = -\sum_{n=1}^{n_j} \text{prcnt}_{icn} \log\left(\text{prcnt}_{icn}\right). \tag{20}$$

Among them, $i = \{1, 2, \ldots, m\}$ and $s = \{1, 2, \ldots, m\}$ represent the number of semantics, and w represents the semantic weight; $|V_c|$ represents the percentage of nodes whose semantic values are in the divided subgraph; and V_C represents the number of semantic values $n = \{1, 2, \ldots, n\}$.

4.3. Experimental Results. Figure 1 is on the Political Blogs data set. We show the comparison results of the density values of the five algorithms when the number of clusters is 3, 5, 7, and 9, respectively. The density value of CS-Cluster changes slowly along with the increase in the number of clusters and has always been higher than the values of the other four comparison algorithms. When $Z = 7$, its density value is the highest 0.93, indicating that the CS-Cluster has reached the structural dissimilarity among nodes in the same subgraph in the cluster division and is the best. Among the five algorithms, the density value of W-Cluster is the lowest among the five algorithms, indicating that the structural dissimilarity of nodes in the same subgraph is the worst. As the number of clusters increases, the density value of IGC-CSM remains basically unchanged, and the density value is relatively high, so the clustering effect of IGC-CSM is second only to CS-Cluster.

In Figure 2, we show the comparison of the density values of the five algorithms in the DBLP data set. As the number of clusters continues to increase, the density value of the W-Cluster algorithm changes more obviously, and compared with the other four comparison algorithms, its density value reaches the lowest value. Therefore, the detection results obtained by the W-Cluster algorithm are used. The structural dissimilarity of nodes in the same subgraph is the worst; IGC-CSM and CS-Cluster have roughly the same density values at $Z = 50$, and the density values are all above 0.85; but the number of clusters is higher than 50 when the density value of CS-Cluster is slightly higher than that of IGC-CSM; the density value of SA-Cluster is in the middle of all algorithms; from the overall situation, the average density of CS-Cluster algorithm is higher than the values of the other four algorithms, which shows that, compared with other comparison algorithms, CS-Cluster has the best structural dissimilarity of the nodes in the subgraph in the clustering results obtained.

In Figure 3, we show the comparison of the entropy values of the five algorithms used on the Political Blogs data set. The number of clusters set in the experiment is $z = 3, 5, 7$, and 9. From the experimental figure, we can see that the entropy value of the CS-Cluster algorithm is always 0; as the number of clusters continues to increase, the entropy value of IGC-CSM is almost stable at about 0.1 and remains unchanged, so it can explain the two algorithms of IGC-CSM and CS-Cluster. It can accurately classify the semantically similar nodes into the same category; but from the comparison of the density values in Figure 3, it can be concluded that the average density of CS-Cluster is slightly higher than that of the IGC-CSM algorithm, so CS-clustering quality of the Cluster algorithm has priority over IGC-CSM. When the z value increases from 3 to 7, the entropy value of SA-Cluster basically remains unchanged, and the value remains at about 0.1, but when the z value increases to At 9 o'clock, the entropy value of SA-Cluster suddenly increased, indicating that when the number of clusters is set to 9, the semantic dissimilarity of nodes in the same subnetwork is poor. Among all the comparison algorithms, the

Table 2: Data subset meeting of DBLP.

Research areas	Meeting name
Database	SIGMOD, VLDB, PODS, ICDE, EDBT
Data mining	KDD, ICDE, SDM, PAKDD, PKDD
Information retrieval	SIGIR, CIKM, ECIR, WWW
Artificial intelligence	IJCAI, AAAI, VAI, NIPS

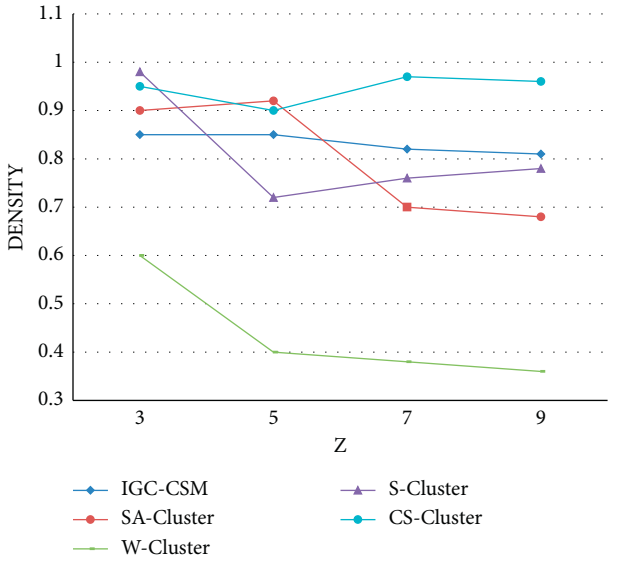

Figure 1: Density analysis on political blogs dataset.

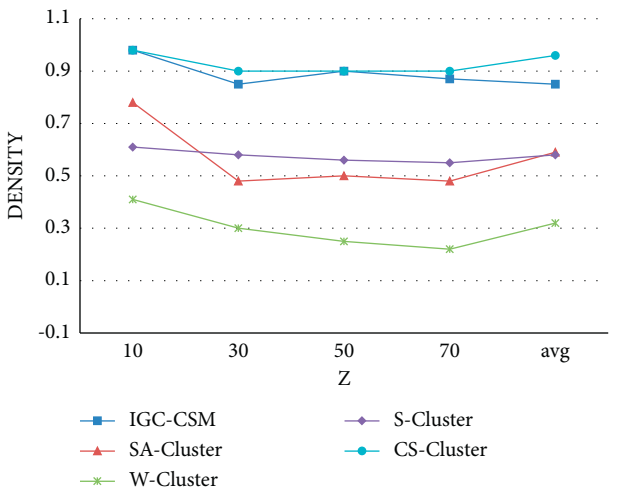

Figure 2: Density analysis on DBLP data.

S-Cluster algorithm has the highest entropy value, indicating that the subgraphs of the clustering division obtained by the S-Cluster algorithm have the worst semantic dissimilarity between nodes.

Figure 4 is the comparison result of the entropy of the four algorithms on the DBLP data set. We set the number of clusters to $z = 10, 30, 50, 70$. Among the five comparison algorithms, the W-Cluster algorithm has the lowest entropy, and the average value remains at 0.3; it means that, in the

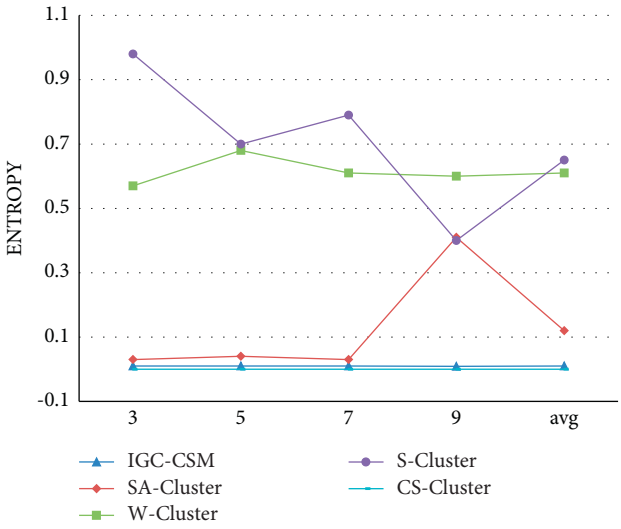

Figure 3: Entropy analysis on political blogs dataset.

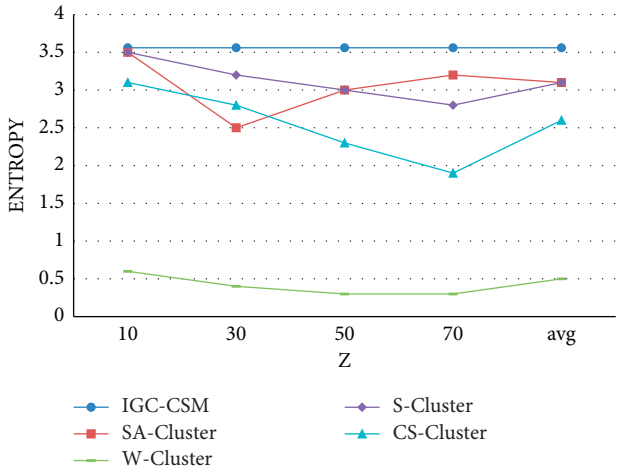

Figure 4: Entropy analysis on DBLP data set.

clustering results obtained, the semantic dissimilarity of the nodes in the subgraph is very high, but in Figure 2, we can see that the density value of W-Cluster is also relatively low. Therefore, it can be judged that the structural dissimilarity between nodes is poor; that is, the connection between nodes is relatively distant. The entropy of the IGC-CSM algorithm is the highest. As the number of clusters increases, the entropy value of the CS-Cluster algorithm shows a downward trend. When the number of clusters is higher than 30, compared with SA-Cluster and CS-Cluster, in the clustering results obtained, the semantic dissimilarity between nodes is relatively high, and the accuracy of the classification results is relatively high.

Figures 5 and 6 show the influence of the balance factor λ on the density and entropy on the Political Blogs dataset when the number of clusters z is 15. It can be seen from Figure 5 that, with the increase of x, the density as a whole shows a trend of gradual increase first, then decrease, and then increase. The density value in the range of λ from 0.5 to

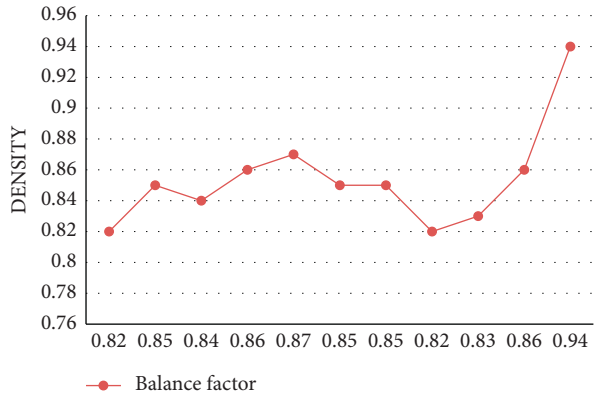

Figure 5: The influence of balance factor λ on density.

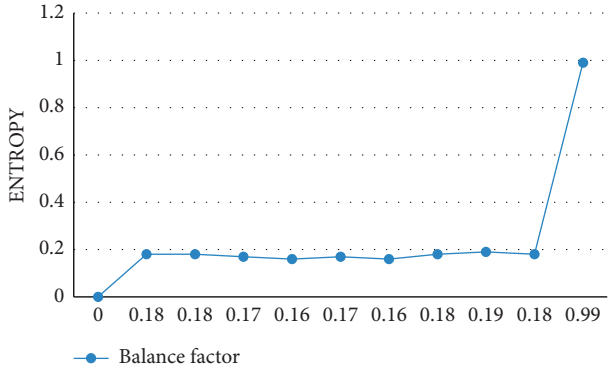

Figure 6: The influence of balance factor λ on entropy.

0.6 is almost unchanged. It can be seen from Figure 6 that the entropy value slowly decreases during the process of the value of a from 0.5 to 0.6, so the value of λ is 0.6 is more appropriate.

5. Conclusion

This article first summarizes the conventional community clustering algorithm and believes that the structural characteristics and semantic information between nodes should be considered comprehensively in the clustering process. Based on this, this article proposes node proximity to complete the calculation of dissimilarity between nodes, introduces associations of the concepts of degree and matching degree, and completes the calculation of structural dissimilarity between nodes. Afterwards, the method of selecting the initial clustering center point was redefined. This method avoids the drawbacks caused by human judgment and improves the accuracy of clustering. Finally, the CS-Cluster algorithm uses the K-Medoids framework to achieve community division. In other algorithm ideas, the experimental comparison on two practical and effective data sets shows that the algorithm proposed in this paper has achieved good clustering results. Data mining and community detection in complex network are a very meaningful research topic. This paper conducts relevant research on the characteristics of nodes in the network and the community detection in the complex network. However, there are still many problems that need to be improved. In the community detection algorithm based on node proximity, this paper only studies the undirected weighted network graph and ignores the clustering of nodes in directed network graph. Secondly, how to effectively select semantic categories in the case of different semantic weight values is also not discussed in this study, which can be further discussed in the future.

Data Availability

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

Conflicts of Interest

The authors declared that they have no conflicts of interest regarding this work.

References

[1] G. Palla, I. J. Farkas, P. Pollner, I. Derényi, and T. Vicsek, "Directed network modules," *New Journal of Physics*, vol. 9, no. 6, p. 186, 2007.

[2] A. Lancichinetti, S. Fortunato, and J. Kertész, "Detection of overlapping and hierarchical community structures in complex networks," *Acta New Physics*, vol. 11, no. 3, pp. 19–44, 2009.

[3] T. S. Evans and R. Lambiotte, "Line graphs, link partitions, and overlapping communities," *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics*, vol. 80, no. 80, pp. 145–148, 2009.

[4] R. Cazabet, F. Amblard, and C. Hanachi, "Detection of overlapping communities in dynamic social networks," in *Proceedings of the 2010 IEEE Second International Conference on Social Computing (Social-Com)*, pp. 309–314, IEEE, Minneapolis, MN, USA, August 2010.

[5] J. Xie, B. K. Szymanski, and X. Liu, "SLPA: uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process," in *Proceedings of the 2011 IEEE 11th International Conference on Data Mining Workshops*, IEEE, Vancouver, Canada, December 2012.

[6] S. Gregory, "Searching for overlapping communities in the network through tag propagation," *New Journal of Physics*, vol. 12, no. 10, pp. 2011–2024, 2009.

[7] U. Raghavan, R. Albert, and S. Kumara, "Near-linear time algorithm to detect community structure in large-scale networks," *Physical Review E*, vol. 76, p. 36106, 2007.

[8] B. Saoud and A. Moussaoui, "Community detection in networks based on minimum spanning tree and modularity," *Physica A: Statistical Mechanics and Its Applications*, vol. 460, pp. 230–234, 2016.

[9] G. Palla, I. Derényi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *Nature*, vol. 435, no. 7043, pp. 814–818, 2005.

[10] J. M. Kumpula, M. Kivela, and K. Kaski, "Sequence algorithm for fast clique penetration," *Physical Review E*, vol. 78, no. 2, pp. 1–7, 2008.

[11] M. Goldberg, S. Kelley, M. Magdon-Ismail, K. Mertsalov, and A. Wallace, "Finding communities in overlapping social networks," in *Proceedings of the Second IEEE Conference on*

Internal Society, pp. 104–113, Minneapolis, MN, USA, August 2010.

[12] D. S. Callaway, M. E. J. Newman, S. H. Strogatz, and D. J. Watts, "Network robustness and fragility: percolation on random graphs," *Physical Review Letters*, vol. 85, no. 25, pp. 5468–5471, 2000.

[13] A. Lancichinetti and S. Fortunato, "Consensus clustering in complex networks," *Scientific Reports*, vol. 2, no. 13, p. 336, 2012.

[14] J. Kim and T. Wilhelm, "Spanning tree separation reveals community structure in networks," *Physical Review E*, vol. 87, no. 3, p. 32816, 2013.

[15] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical Review E*, vol. 69, no. 2, p. 26113, 2004.

[16] S. Deng, C. Wang, M. Wang, and Z. Sun, "A gradient boosting decision tree approach for insider trading identification: an empirical model evaluation of China stock market," *Applied Soft Computing*, vol. 83, Article ID 105652, 2019.

[17] P. Pons and M. Latapy, "Computing communities in large networks using random walks," *Journal of Graph Algorithms and Applications*, vol. 10, no. 2, pp. 191–218, 2006.

[18] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi, "Defining and identifying Communities in networks," *Proceedings of the National Academy of Sciences*, vol. 101, no. 9, pp. 2658–2663, 2004.

[19] M. Girvan and M. E. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, no. 6, pp. 7821–7826, 2002.

[20] M. E. J. Newman, "Goldon Fast algorithm for detecting community structure in networks," *Physical Review E, Statistical, Nonlinear, and Soft Matter Physics*, vol. 69, no. 6, p. 66133, 2003.

[21] D. Saoud and A. Moussaoui, "Community detection in networks based on minimum spanning Applications: S0378437116301996.tree and modularity," *Physica A Statistical Mechanics*, vol. 460, 2016.

[22] U. Brunn, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," *Physical Review E*, vol. 76, p. 36106, 2007.

[23] B. W. Kernighan and S. Lin, "An efficient heuristic procedure for partitioning graphs," *Bell System Technical Journal*, vol. 49, no. 2, pp. 291–307, 1970.

[24] S. Fortunato, V. Latora, and M. Marchiori, "Method to find community structures based on Information centrality," *Physical Review E, Statistical, nonlinear, and soft matter physics*, vol. 70, no. 5, p. 56104, 2004.

[25] M. Zarei, D. Lzadi, and K. A. Samani, "Detecting overlapping community structure of networks based on vertex-vertex correlations," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 11, p. 11013, 2009.

[26] W. Nawaz, K.-U. Khan, Y.-K. Lee, and S. Lee, "Intra graph clustering using collaborative similarity measure," *Distributed and Parallel Databases*, vol. 33, no. 4, pp. 583–603, 2015.