Hindawi
Security and Communication Networks
Volume 2023, Article ID 9857863, 1 page
https://doi.org/10.1155/2023/9857863

Retraction

Retracted: Overlapping Community Detection Based on Node Importance and Adjacency Information

Security and Communication Networks

Received 26 December 2023; Accepted 26 December 2023; Published 29 December 2023

This article has been retracted by Hindawi, as publisher, following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of systematic manipulation of the publication and peer-review process. We cannot, therefore, vouch for the reliability or integrity of this article.

Please note that this notice is intended solely to alert readers that the peer-review process of this article has been compromised.

Wiley and Hindawi regret that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] P. Wang, Y. Huang, F. Tang, H. Liu, and Y. Lu, "Overlapping Community Detection Based on Node Importance and Adjacency Information," *Security and Communication Networks*, vol. 2021, Article ID 8690662, 17 pages, 2021.

Hindawi
Security and Communication Networks
Volume 2021, Article ID 8690662, 17 pages
https://doi.org/10.1155/2021/8690662

Research Article

Overlapping Community Detection Based on Node Importance and Adjacency Information

Ping Wang,[1] Yonghong Huang ,[1,2] Fei Tang ,[1,2] Hongtao Liu,[1] and Yangyang Lu[1]

[1]*College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China*
[2]*School of Cyber Security and Information Law, Chongqing University of Posts and Telecommunications, Chongqing 400065, China*

Correspondence should be addressed to Yonghong Huang; huangyonghong@cqupt.edu.cn

Received 12 November 2021; Accepted 9 December 2021; Published 31 December 2021

Academic Editor: Jian Su

Detecting the community structure and predicting the change of community structure is an important research topic in social network research. Focusing on the importance of nodes and the importance of their neighbors and the adjacency information, this article proposes a new evaluation method of node importance. The proposed overlapping community detection algorithm (ILE) uses the random walk to select the initial community and adopts the adaptive function to expand the community. It finally optimizes the community to obtain the overlapping community. For the overlapping communities, this article analyzes the evolution of networks at different times according to the stability and differences of social networks. Seven common community evolution events are obtained. The experimental results show that our algorithm is feasible and capable of discovering overlapping communities in complex social network efficiently.

1. Introduction

With the development of information technology, social networks such as microblogs and forums have developed rapidly in recent years and gradually become an important platform for people to exchange feelings, share experiences, and transmit information. At present, there are many community detection algorithms based on node set partition, e.g., Kernighan–Lin algorithm [1] based on greedy algorithm theory, spectral bisection algorithm [2] based on spectral thought, GN algorithm [3] based on splitting thought, and Newman fast algorithm [4] based on cohesion thought. However, the ultimate goal of these community detection algorithms is to divide the network into several independent communities. The above algorithms strictly divide each node into specific communities and can not find overlapping communities. In an in-depth study of real networks, researchers have found that many real networks not only have community structures but also have the characteristics of overlapping and correlation among communities. The network is composed of many overlapping and interconnected communities [5]; e.g., in interpersonal networks, everyone belongs to several different communities such as school, family, and friends according to different classification methods. As the basis of studying the network structure, revealing the community structure in the network is of great significance to study the function of the network and analyze the composition structure of the network. Overlapping community detection has become the key issue in researching opportunity network structure. Many scholars have proposed a lot of overlapping community detection algorithms, e.g., LFM [6], DEMON [7], and SLPA [8]. However, the accuracy and complexity of current overlapping community detection algorithms need to be improved.

In this article, overlapping community detection algorithms are divided into four categories. They are the clique penetration algorithms, the label propagation algorithms, the link partition algorithms, and the local expansion algorithms. We mainly focus on the overlapping community detection algorithms based on local expansion. It is found that many local expansion algorithms take node centrality as

the evaluation indicator. The initial nodes largely determine the division results of overlapping communities. The random selection of initial nodes leads to a certain deviation in the membership of overlapping nodes after community division. For the interest communities, when the user belongs to both the tennis and basketball communities, they have inconsistent preference evaluation of different communities, which makes the number of overlapping node members deviate. If all nodes in the network are used as the initial nodes of expansion, it is easy to form an independent community, which will make the community detection inaccurate. Through the research on the diversity of complex networks, we find that only a single evaluation indicator is prone to the high overlap of local communities. More factors need to be considered for nodes selection. We also find that it needs to have the following characteristics if the nodes can be located in the community center to stabilize the structure of the whole community. (a) The node must be a certain number of neighbor nodes. (b) Both the influence of neighbor nodes on the node itself and the high connectivity between neighbor nodes should be considered. (c) Whether the neighbor node is also of high importance.

Social network analysis [9] has become an important research field and plays an important role in data mining, information dissemination, network modeling, behavior analysis, knowledge discovery and so on. In the real world, most social networks are dynamic networks. Their nodes will change and the relationship between nodes will evolve over time. The relationship density between nodes is different. Some nodes have relatively dense connections, while others have relatively sparse connections. A group of nodes with relatively dense connections is regarded as a community [10]. Nodes in the community usually have some common attributes. This reflects the local rules and global order of the social network to a certain extent. Extracting community structure can effectively reveal the characteristics of social networks and the general laws of groups. It is the basis for analyzing the evolution of social networks. It also contributes to promoting the development of relevant applications, e.g., friend recommendation, privacy protection, network marketing, and role analysis [11]. Therefore, community extraction is an important research area in the field of social network analysis, which is of great significance for analyzing and understanding the structural attributes and group characteristics of social networks.

In the study of dynamic networks changing with time, it is necessary to evaluate the community evolution between the upper and lower time slices. Establishing accurate and effective evaluation criteria is important to judge whether there is an evolutionary relationship between communities. The common method is to calculate the similarity of adjacent time communities. There are also many studies of similarity calculation, e.g., the evaluation method based on the Jaccard coefficient and Normalized Mutual Information (NMI). Choosing a reasonable and effective evaluation criterion and setting an evaluation threshold are challenging in the current research. We think that both the similarity and integrity of community evolution in the upper and lower time slices should be considered aiming at the evaluation of community evolution in the adjacent time in a dynamic network. Based on the idea of galaxy formation in the universe [12], this article gives mass to all nodes in the network and tries to establish the center of gravity relationship matrix between nodes. Finally, it selects the nodes with local characteristics globally and divides them into different node chains as the initial structure of the community. For static networks, communities attract surrounding nodes through multipoint iteration under the action of gravity. It can be found that for the community structure of adjacent time slices, the change of core node gravity chain can be used to observe the evolution process of community integrity. Then, it combines with the local dissimilarity to jointly evaluate the evolutionary behavior.

Based on the above analysis, we present an overlapping community detection algorithm based on node importance and adjacency information. The main contributions of this work are listed as follows:

(1) Many indicators of node importance often consider a single factor. Our indicator considers node importance and adjacency information, i.e., the importance of the node itself, the importance of node neighbors, and the connectivity between neighbors.

(2) We design an overlapping community detection algorithm based on node importance and adjacency information. It uses the proposed node relevance centrality to evaluate the importance of nodes.

(3) Based on the results of the overlapping community detection algorithm, we further study the community structure at different moments.

The rest of this article is organized as follows. In Section 2, we introduce the related work of our proposed algorithm (ILE). The proposed node relevance centrality and related definitions of the ILE algorithm will be defined in Section 3. Then, in Section 4, we present the ILE algorithm. In addition, the experimental results and analysis are given in Section 5. Last but not least, we draw a conclusion in Section 6.

2. Related Work

2.1. Network Research. The natural world can be abstracted as a complex network, e.g., the ecological network with a clear division of labor in the ant nest, the citation network of academic papers, the mysterious neural network in the human brain, and the colorful public opinion network on the Internet. The complex network refers to a network with a certain degree of complexity in structure or attributes. Networks with self-organization, self-similarity, attractor, small world and scale-free partial or complete characteristics are called complex networks [9]. Generally, complex networks have the characteristics of large scale, complex connection structure, complex nodes, sparse connections, complex spatiotemporal evolution process, and so on. For many complex networks in the real world, we only need to treat the entities in the network as nodes and the relationship between entities as edges. A complex network is a tool used to explore complex systems. It is also a research hotspot in

Mathematics in graph theory. The whole network graph is usually represented by $G = (V, E)$, where V represents the set of nodes in the network graph, E represents the set of edges, and the two nodes in set V just correspond to one edge in set E. The research of complex networks mainly focuses on the structural analysis of graphs composed of nodes and edges.

2.2. Overlapping Community Detection. Since community detection algorithms are used in complex networks, a large number of algorithms have been proposed. According to whether there are overlapping nodes or not, they can be divided into overlapping community detection algorithms and nonoverlapping community detection algorithms. Some representative nonoverlapping community detection algorithms include GN [1], CNM [5], and LPA [13]. Each node can only belong to a separate community, and all communities in the network are not connected. With the in-depth study of community detection, people also find that this community structure is obviously not common in the real world. This hard division cannot really reflect the actual relationship between nodes and communities.

In the real world, many communities in complex networks are not isolated, and they constantly communicate and overlap. Like our own friends, relatives, and other different relationships, we can belong to multiple social networks. A person's title can be the professor, the university dean, or visiting scholar; i.e., a person may have different attributes in different fields. When any node in the network belongs to two communities at the same time, it is shared by the two communities. These communities with shared nodes are called overlapping communities [14]. Overlapping community detection is more in line with the law of community organization in the real world. In recent years, it has become a new hotspot in community detection research. Many overlapping community detection algorithms have emerged.

The clique penetration algorithm proposed by Palla et al. [15] is the first algorithm that can detect overlapping communities. This algorithm considers that the community is composed of a series of mutually reachable k-clique. It realizes overlapping community detection by merging adjacent k-communities. The nodes in multiple k-communities are the overlapping part. Kumpula et al. [16] further proposed a fast clique penetration algorithm (SCP), which greatly improves the speed of the clique penetration algorithm. These algorithms based on the idea of clique penetration need to take the group as the basic unit to find overlapping parts. Many real networks, especially sparse networks, can find few overlapping communities [17]. Zhou et al. [18] combined clique percolation algorithm and K-means algorithm for detection, which, however, is mainly suitable for dense networks. In the label propagation algorithms, Gregory et al. [19] improved the LPA algorithm and proposed the COPRA algorithm. It allows each node to store multiple tags through the tag list; i.e., each node can belong to a community at the same time. The SLPA algorithm proposed by Xie et al. [8] divides the tags meeting the threshold frequency into corresponding communities by iterating the node tags many times. In the link partition-based algorithms, Evans et al. [20] transformed the overlapping community division into link partition using edges to represent nodes. It incorporated the node-based community detection algorithm to detect the structure of links. The L-Attractor algorithm proposed by Chen et al. [21] transformed the original graph into a link graph and introduced a dynamic interaction process to simulate distance dynamics. All distances converge through the dynamic interaction process. Disjoint community structures appearing in the link graph transformed into overlapping community structures of the original graph.

Among the local expansion algorithms, Lancichinetti et al. [6] proposed the LFM algorithm. It expands the community by randomly selecting initial nodes. A community is formed after an iteration and meets the requirements of the threshold function. It randomly selects new seed nodes outside the community and divides them according to the above method. Coscia et al. [7] proposed the DEMON algorithm. It selects all nodes in the network as the initial nodes and continuously expands the neighbors on the premise of meeting the threshold. Then, similar extended communities are merged to obtain the final overlapping community. Cao et al. [22] realized local community expansion by minimizing the conduction value of the cluster. If the conduction value of the cluster decreases when removing the nodes in a given cluster, it is removed outside the cluster and the iteration is repeated until the conduction value of the cluster reaches a stable state. Zhou et al. [23] proposed a local community detection algorithm based on minimum cluster. The algorithm selects one of the K initial nodes randomly and finds its neighbor nodes. If the found node has the same neighbors as the given initial node, the three constitute the smallest community. Although there have been many studies in overlapping community detection so far, the complexity and accuracy of existing overlapping community detection algorithms still need to be improved. Therefore, it is necessary to study the detection of overlapping communities [24].

However, a very important step in using the local expansion algorithm to detect community is to select the initial node. The accuracy of the local extension community detection algorithm largely depends on the quality of the initial node. Different community detection algorithms based on local extension have different seed selection schemes. LFM algorithm selects the initial nodes in a random way. It is simple and fast at the beginning. If the initial node is an edge node, a large number of highly overlapping communities and overlapping nodes will be formed. The randomly selected nodes often have some uncertain factors, so the results are inconsistent every time. It needs to run many times to get better division results. The initial nodes in the DEMON algorithm are all nodes in the network. On the premise of meeting the threshold, the DEMON algorithm continuously expands the neighbors around the seed nodes. It merges the communities with high similarity to obtain the final overlapping community. When the network scale is large, it will be very time-consuming to detect the largest complete

subgraph in the network. In addition, when there are multiple edge nodes, it is easy to have highly overlapping communities.

In most local expansion algorithms, the selection of initial nodes largely determines the detection of overlapping communities. There are many indicators to evaluate the importance of nodes. Node centrality is one of the early representative indicators. Many local expansion detection algorithms take node centrality as an evaluation indicator due to the node at the center of the network can well stabilize the structure of the whole community; e.g., Wang et al. [25] proposed the concept of structural centrality node, based on which local expansion community was carried out. It is highly accurate but only works on smaller networks. Degree centrality is the main measurement standard of node centrality. Generally, the greater the degree of nodes in the network, the higher the centrality of nodes. Betweenness centrality measures the ability of a node to act as a medium. If the node acts as a medium more frequently, its value is greater. Closeness centrality is defined as the length of the shortest path for a node to reach any other node in the network. The larger the value, the higher the importance of the node. In addition, the clustering coefficient is an indicator to evaluate the connectivity between nodes and adjacent nodes in the network. It is measured by calculating the ratio of the number of triangles formed by nodes in the actual network to the number of triangles expected to be surrounded by nodes. The larger the clustering coefficient, the closer the relationship between the node and its neighbors.

2.3. Overlapping Community Evolution. The main goal of social network evolution is to find the community structure of different time slices. People can mine the real-time changing community structure by studying the dynamic community evolution. According to the analysis of the existing social network evolution algorithms, they can be divided into similarity-based evolutionary algorithms and core node-based evolutionary algorithms.

In the evolution analysis based on similarity algorithms, the similarities of communities in adjacent time slices are usually compared to determine the evolution events. Yang et al. [26] used the Jaccard coefficient to calculate the ratio of the intersection and union of community nodes in two adjacent time slices, as it can better obtain the similarity of the community. Yu et al. [27] added community activity and influence to the Jaccard coefficient. Based on a three-way decision, the law of community evolution is judged by these three parameters. Zhu et al. [28] proposed the concept of community attribute based on the Jaccard coefficient and reconstructed each event according to community attribute. The above evolutionary analysis algorithms are based on the Jaccard coefficient or improved similarity analysis. They failed to consider the topology of specific networks, so the accuracy of evolutionary event results is reduced.

Bhat et al. [29] proposed a density-based evolutionary algorithm. The algorithm selects the core nodes by calculating the density value. It observes the community structure of adjacent time slices and updates the attributes of nodes in a log-based manner. The above steps iterate until a complete community evolution path is formed. Dhouiou et al. [30] identified evolution events based on edge nodes. They found out the core nodes and nodes with few edges in each community. Then, they put the least edge nodes into the existing core community set. Finally, they observed the changes of core nodes to determine the category of evolution events. Starting from the initial community structure defined by the group membership relationship, Karan et al. [31] described the time evolution process from the initial community structure to the current network topology according to the intensity and frequency of interaction between members and the degree of overlap between different communities. Yu et al. [32] presented a new evolutionary model framework based on orthogonal nonnegative matrix decomposition. The essence of the framework is to assume that the community structure obeys the local evolution pattern (LEP) in each snapshot. These local evolution patterns come from the common global evolution pattern (GEP). It can synchronously detect the temporal community structure, extract the evolution pattern, and predict the structure, including future snapshots. Wang et al. [33] proposed a new dynamic overlapping community evolution tracking method. This method detects the initial overlapping community structure of peak valley structure in the topological potential field based on node location analysis. It updates the dynamic community structure incrementally and tracks the community evolution events based on the changes of core nodes. Through the analysis of existing evolutionary algorithms based on core nodes, it can be found that these algorithms consider the global topology information of the network and effectively improve the accuracy of the current community evolution results. However, the core nodes of these algorithms have different characteristics and are closely related to the type of evolution events. Mining the characteristics of different core nodes is necessary so as to better analyze more different evolution events through core nodes.

When analyzing the evolution of dynamic networks, dynamic networks are usually transformed into static networks. In addition to community extraction, the study of community evolution evaluation criteria is also an indispensable part of dynamic network evolution analysis, i.e., to study the relationship between communities with time characteristics and to determine whether there is an evolution between communities.

The most classic evaluation criteria Jaccard evaluates the similarity between communities by calculating the proportion of common nodes and setting a threshold. It is considered that there is an evolutionary relationship between communities when the similarity is greater than the threshold. The similarity based on Jaccard still judges the similarity of the community according to the proportion of shared nodes, regardless of the weight of nodes. Gliwa et al. [34] added the node weights to the similarity formula for the first time. Although the importance of nodes is considered, the importance of any node can be selected to calculate the similarity. The similarity will be affected by the selection of

node importance; e.g., Anna et al. [14] evaluated the community similarity from the change degree of node importance ranking. Generally, most studies do not consider the impact of node weights on the similarity evaluation. Some consider the node weight and ignore the node weight calculation method. Both of them have a great impact on the quality and accuracy of community evolution evaluation results.

3. Preliminaries

3.1. Node Relevance Centrality. Most of the indicators to evaluate the centrality of nodes take the degree of nodes as the main measurement standard. Degree centrality often ignores the relationship between the selected nodes and their neighbors. Closeness centrality and betweenness centrality can not be used in large-scale networks because the information of the global network topology is considered. Although the clustering coefficient considers the connection relationship between nodes, it only considers the direct connection. Therefore, the effect of the initial node selected on certain networks is not obvious.

We hold that a single evaluation indicator of node centrality is prone to a high overlap of local communities. In order to ensure that the node is at the center of the network, we consider the importance of the node and the adjacency informaton, i.e., the importance of the node itself, the number of node neighbors, and the connection between neighbors. The proposed formula is as follows:

$$I_k = \frac{2\left|e_{ij}\right|}{d_k\left(d_k - 1\right)} \sum_{m \in N(k)} \delta(m, k), \tag{1}$$

where i and j are the neighbor nodes of the node k; e_{ij} represents the edge connected by node i and node j; for the function $\delta(m, k)$, m is the neighbor nodes of k. It represents the maximum number of paths that neighbor nodes of m can reach the node k, excluding the neighbor nodes of k. d_k is the degree of the node k. I_k indicates the importance of node k. If the node is more important, the value is larger. The importance score I_k of node k is equivalent to formula (1) and is defined as follows:

$$I_k = \frac{\left|e_{ij}\right|}{d_k\left(d_k - 1\right)} \sum_{m \in N(k)} \left(\frac{\left|e_{ij}\right|}{d_k\left(d_k - 1\right)} \cdot \frac{\left|N(k) \cap N(m)\right|}{\left|N(k) \bigcup N(m)\right|} \right). \tag{2}$$

$N(k)$ and $N(m)$ are the neighbor sets of k and m, respectively.

The proposed node relevance centrality avoids the problems of the above evaluation indicators. Our indicator considers the local information of network topology. Calculations are greatly reduced in a large-scale network. Especially when some nodes are around many neighbors, but the neighbors have no other connections, the node relevance centrality value is 0.

3.2. Similarity Evaluation. At present, similarity-based methods are used to compare the similarity of communities before and after time slices [35]. Although the overall structure of the community can be compared simply and quickly, the accuracy of the results is difficult to be guaranteed due to the lack of consideration of the network topology. Taka et al. [36] introduced the vertex comparison strategy. This method only considers the changes of the core nodes of the community. It does not explain the selection strategy of the core vertices and lacks the consideration of the overall evolution. Bródka et al. [37] gave a more reasonable group evolution detection algorithm (GED). A tolerance change indicator is proposed to dynamically balance and transform the number of nodes and importance. GED defines the event classification of community evolution comprehensively. However, it is sensitive to the scale of the community, especially for some smaller networks. Although Magnien et al. [38] combined the importance of nodes into the similarity evaluation formula, there are still some problems.

Giving the mass to the nodes, the concept of universal gravitation [12] is introduced into the complex network. The nodes with greater influence can usually affect their neighbor links. The link gravitation coefficient defined in this article represents the influence of links in the network. The larger the gravity coefficient of the link, the more the nodes in the core area. The greater its influence is, the more it can attract the neighbor nodes of the nodes on the link to join its community link. Given any link in the network, the link gravity coefficient is as follows:

$$G_c = \frac{\sum_{i \in l} l_i^{(g)} + 1}{\min\left(m_i\right)}, \tag{3}$$

where $l_i^{(g)}$ is the number of g -polygons with node i on the link l. The denominator is the minimum degree of node i on link l. Adding 1 to the molecule is to prevent the number of g-polygons on link l from being 0.

The gravity chain formula of the core node is as follows:

$$F_i^t = G_c \cdot m_k, \tag{4}$$

where m_k is the degree of the core node.

Considering the global network community, we combine the dissimilarity to measure the community evolution in the upper and lower time slices. The dissimilarity formula is as follows:

$$\begin{aligned} \mathrm{Dis}\left(C_i^t, C_j^{t+1}\right) &= \mathrm{norm}\left(\frac{\sum_{k \in C_i^t \cap C_j^{t+1}} \eta \cdot \mathrm{avg}\left(I_k\right)}{C_i^t \bigcup C_j^{t+1}} \right), \\ \eta &= \left| \mathrm{index}_{C_i^t}(k) - \mathrm{index}_{C_j^{t+1}}(k) \right|, \end{aligned} \tag{5}$$

where η is the difference of node importance between communities in adjacent time slices. Anna et al. [14] took the square of η. The minimum of the result can reach $10 - e3$ and the maximum can reach $10 - e4$. The data are unevenly

distributed, resulting in the lack of good community division after normalization. The above dissimilarity formula is combined with the node variation range in the core node gravity chain to jointly evaluate the evolution trend of the community. If the difference is larger, the community in the time slice C_i^t is more different from that in the time slice C_j^{t+1}. $\mathrm{avg}(I_k)$ is the average node importance. The denominator is the number of common nodes. The denominator is the union of the nodes in [14]. If the node exists at the last time and disappears at the next time, the node dissimilarity can not be calculated. norm function is used to normalize the formula. It can avoid the non-standard problem of threshold selection. Typically, the value range of data normalization is $(0, 1)$.

Community evolution analysis mainly completes two tasks: first, whether there is evolution between communities; second, what is the type of evolution. We propose a community evolution algorithm based on the above similarity evaluation method. The algorithm uses the core node selected by the proposed node correlation centrality to construct the core node gravity chain. According to the change of gravity chain, we observe the evolution of community integrity. The dissimilarity evaluates the evolution of community nodes outside the core node gravity chain. Therefore, we analyze the aggregation behavior of the community and establish a new evolution analysis model. The algorithm makes up for the defect that the GED algorithm is sensitive to community scale. It is more general in data selection and can well mine the types of evolutionary events of different time slices. The event-based overlapping community evolution model is as follows:

Forming: some unconnected nodes in the network form a community at time t due to the increasing contact. The similarity between any community at time $t-1$ and the community at time t satisfies the following:

$$\mathrm{Dis}\left(C_i^t, C_j^{t-1}\right) < \alpha \wedge \frac{C_i^{t-1} \cap C_j^t}{C_i^{t-1}} > \beta. \tag{6}$$

Continuing: the gravity chain of the core node of the community C_i^t is not disconnected at time $t+1$, and the community continues in the next time window if and only if the community C_i^{t+1} exists and satisfies the following:

$$\mathrm{Dis}\left(C_i^t, C_j^{t+1}\right) < \alpha \wedge \frac{C_i^{t-1} \cap C_j^t}{C_i^{t-1}} > 90\%. \tag{7}$$

Growing: the gravity chain of the core node of the community C_i^t is not disconnected at time $t+1$, and and the community size is bigger than that at time t if and only if community C_i^{t+1} exists and satisfies the following:

$$\mathrm{Dis}\left(C_i^t, C_j^{t+1}\right) < \alpha \wedge \frac{F_j^{t+1} - F_i^t}{F_i^t} > 10\%. \tag{8}$$

Shrinking: the nodes on the gravity chain of the core nodes in the community C_i^t decrease at time $t+1$, but the community as a whole remains in a continuous state if and only if community C_i^{t+1} exists and satisfies the following:

$$\mathrm{Dis}\left(C_i^t, C_j^{t+1}\right) < \alpha \wedge \frac{F_i^t - F_j^{t+1}}{F_i^t} > 10\%. \tag{9}$$

Splitting: the gravity chain of the core node of the community C_i^t is broken at time $t+1$; if and only if there are multiple (greater than or equal to 2) communities $S^{t+1} = C_1^{t+1}, \ldots, C_n^{t+1}$ at time $t+1$, $\forall C_k^t t+1 \in S^{t+1}$ satisfies the following:

$$\frac{F_i^t - F_k^{t+1}}{F_i^t} > \beta > \mathrm{Dis}\left(C_i^t, C_k^{t+1}\right) < \gamma \wedge \left|C_i^t\right| > \left|C_k^{t-1}\right|,$$
$$\gamma = \alpha + 0.1. \tag{10}$$

Merging: there are multiple gravity chains of core nodes in community C_i^t at time $t-1$. If and only if multiple (greater than or equal to 2) communities $S^{t-1} = C_1^{t-1}, C_n^{t-1}$ exist at time $t-1$, $\forall C_k^t t+1 \in S^{t+1}$ satisfies the following:

$$\frac{F_i^t - F_k^{t-1}}{F_i^t} > \beta > \mathrm{Dis}\left(C_i^t, C_k^{t-1}\right) < \gamma \wedge \left|C_i^t\right| > \left|C_k^{t-1}\right|,$$
$$\gamma = \alpha + 0.1. \tag{11}$$

Dissolving: the community C_i^t disappears at the time $t+1$ only if any community C_i^{t+1} at the time $t+1$ does not exist and satisfies the following:

$$\mathrm{Dis}\left(C_i^t, C_j^{t+1}\right) < \alpha \wedge \frac{C_i^t \cap C_j^{t+1}}{C_i^{t+1}} > \beta. \tag{12}$$

3.3. Related Concepts. With the in-depth study of complex networks, it is found that the problem of community detection mainly focuses on the limited information of unauthorized and undirected networks. How to mine more valuable information? How to define the community more accurately after obtaining rich information and correctly assigning nodes to the corresponding community? The following mainly introduces the concepts of structure information, community boundary, and attributes mentioned in relevant overlapping community detection algorithms.

Definition 1 (community neighbor set). Community neighbor set $N_s(C)$ is a combination of nodes that have direct connection edges with community C.

$$N_s(C) = \bigcup_{v \in C} Z(v) - C,$$
$$Z(v) = \{u: u \in V, (v, u) \in E\}, \tag{13}$$

where C represents a community and $Z(v)$ represents the neighbor set of node v.

Definition 2 (Jaccard coefficient [39]). The Jaccard coefficient J_{uv} between two nodes is defined as follows:

$$J_{uv} = \frac{|Z(v) \cap Z(u)|}{|Z(v) \cup Z(u)|}. \tag{14}$$

Jaccard coefficient J_{uv} can be used to measure the closeness between nodes. The lager the J_{uv}, the more similar the two nodes are.

Definition 3 (node and community similarity). The similarity between node k and community C is defined as follows:

$$S_{kc}(k, C) = \frac{\left|N_s(C) \cap Z(k)\right|}{\left|N_s(C) \cup Z(k)\right|}, \tag{15}$$

where $N_s(C)$ refers to the node set that has a direct connection edge with community C. $S_{kc}(k, C)$ reflects the degree of similarity between node k and community C. The larger the value, the higher the similarity between the node and the community.

Definition 4 (community similarity). $S_{cc}(C_m, C_n)$ is the similarity between community C_m and community C_n:

$$S_{cc}(C_m, C_n) = \frac{\left|C_m \cap C_n\right|}{\min\left(\left|C_m\right|, \left|C_n\right|\right)}. \tag{16}$$

The larger the value of $S_{cc}(C_m, C_n)$ is, the greater the similarity between community C_m and community C_n will be. The two communities will merge if they meet a certain threshold range.

Definition 5 (clustering coefficients [40]). The clustering coefficients among communities represent the relationship between communities. It is defined as follows:

$$C_c = \frac{2F_c}{n_c(n_c - 1)}, \tag{17}$$

where n_c denotes the number of nodes in the community c and F_c is the number of actual edges. The clustering coefficient between communities is similar to the clustering characteristics in complex networks. It is equal to the ratio of the actual number of edges to the theoretical maximum number of edges in community c.

Definition 6 (adaptive function [41]). The adaptive fitness function measures the tightness of nodes in the community. The specific formula is defined as follows:

$$\mathrm{CQ} = \frac{C_{in}^2}{\left(C_{\mathrm{in}}^2 + C_{\mathrm{out}}^2\right)^{\alpha}}, \tag{18}$$

where C_{in} and C_{out} represent the sum of node degrees in the community and the sum of node degrees outside the community, respectively. The parameter α controls the size of the community, where $\alpha \in Z_+$. The greater the value of CQ, the higher the compactness between nodes in the community.

Definition 7 (transition probability matrix). The nodes of the network matrix are weighted by the importance of the nodes, and the weighted adjacency matrix M normalized by the row vector is used as the one-step transition probability matrix P of the random walk strategy. Its expression is as follows:

$$P_{ij} = \frac{M_{ij}}{\sum_{r=1}^{G} M_{ir}}, \tag{19}$$

where M_{ij} represents the importance of nodes; P_{ij} represents the transition probability; and G represents the global network.

Definition 8 (node probability distribution). Assuming that the z-step arrival probability distribution is e_0. $\lambda_s^l(i)$ represents the probability of an agent starting from node s to node i after z-step transfers. It can be expressed by iterative equation as follows:

$$\lambda_s^l = \sum_{r=1}^{|G|} \lambda_s^l(r) P_{ij}. \tag{20}$$

Definition 9 (average probability). The average probability a of all nodes is adopted in this article to divide nodes i and s greater than it into the same community.

$$a = \frac{\sum_{i=1}^{|G|} \lambda_s^l(i)}{G_n}. \tag{21}$$

Definition 10 (stability [42]). The stability formula based on Jaccard mainly calculates and compares common nodes and their number of the two communities in adjacent time windows. The definition is as follows:

$$S_t\left(C_i^t, C_j^{t+1}\right) = \max\left(\frac{C_i^t \cap C_j^{t+1}}{\left|C_j^t\right|}, \frac{C_i^t \cap C_j^{t+1}}{\left|C_j^{t+1}\right|}\right), \tag{22}$$

where C_i^t and C_j^{t+1} correspond to the number of communities at different times. Overlapping nodes are also represented by this formula.

Definition 11 (modularity [43]). The overlapping community modularity EQ function is an extended function of the modularity function. The calculation formula is as follows:

$$\mathrm{EQ} = \frac{1}{2|E|} \sum_{C} \sum_{i,j \in C_k} \frac{1}{Q_i Q_j}\left(A(i, j) - \frac{d_i d_j}{2|E|}\right), \tag{23}$$

where the number of communities to which node i and node j belong is represented by $Q1$ and $Q2$. d_i and d_j represent the degrees of node i and node j. $A(i, j)$ is the adjacency matrix of the network. EQ represents the quality of the result of

community detection. The larger its value, the better the result of community detection.

Definition 12 (normalized mutual information [44]). Normalized mutual information evaluates the similarity between the real network and the community structure detected by the overlapping community detection algorithm. Assuming that community A and community B are the results of the detection in the network, the hybrid matrix C stores the number of nodes divided into community i in A and community j in B simultaneously:

$$I(A,B) = \frac{-2\sum_{i=1}^{C_A}\sum_{j=1}^{C_B} C_{ij}\log\left(C_{ij}N/C_iC_j\right)}{\sum_{i=1}^{C_A} C_i\log\left(C_i/N\right) + \sum_{j=1}^{C_B} C_j\log\left(C_j/N\right)}, \quad (24)$$

where N represents the number of nodes in the network; C_i represents the sum of all node elements in a row of the matrix, i.e., the number of nodes divided into community i in A; C_A represents the number of communities in the A network; and C_{ij} is the number nodes owned by communities i and j jointly. The value range of $I(A,B)$ is between 0 and 1. The larger the value, the higher the similarity between the real network and the detected community structure.

4. Proposed Algorithm

The local overlapping community detection algorithm based on node importance and adjacency information mainly consists of four stages: (1) seed communities detection; (2) merger of similar seed communities; (3) community expansion; (4) community optimization. The detection of seed community is mainly by calculating the influence score of each node according to the proposed node relevance centrality and selecting the core seed node according to adjacency information. Then, it forms a seed community with the neighbor nodes with a close structure. In the community expansion stage, it selects the nodes with high similarity to the community and can optimize the adaptive function to join the community. After the above four stages, the detection of the whole network can be realized. Because each seed community expands independently along its neighbor set, the overlapping structure in the network can be found.

4.1. Seed Community Detection. The first step is to calculate the importance of all nodes and store the node importance calculated by formula (1) to the set S_{core}. Then, it counts the number l_{num} of each node whose score is greater than its neighbor nodes and the number n_{num} of neighbors. The nodes are added to the set S_{core} if the ratio of l_{num} to n_{num} is greater than the threshold ρ. It sorts the node importance in the set S_{core} in descending order and selects the first node as the initial core node. The core nodes are expanded by random walk locally. The construction of the initial core community is completed (Algorithm 1).

4.2. Similar Seed Communities Merger. By detecting the seed community, it is possible that the seed communities will be very similar. They need to be merged to avoid unnecessary calculations in the seed expansion phase. The similarity $S_{cc}(C_m, C_n)$ between communities is calculated according to Definition 4. The seed communities will be merged to obtain a more stable and compact seed community set Seed_{sm} if $S_{cc}(C_m, C_n)$ is greater than the threshold ϵ (Algorithm 2).

4.3. Community Expansion. After getting a stable seed community, the expanded step is to obtain the neighbor set N_s of the seed community firstly. It calculates the similarity S_{nc} between each neighbor node $z \in N_s$ and the community using the formula in Definition 1. Then, it selects the node with a similarity greater than the threshold ϵ as the candidate node. It calculates the fitness function of these candidate nodes after joining the local community. The candidate nodes that can increase the value of the fitness function are added to the community; otherwise, they will be regarded as free nodes in the network. The nodes in the community whose fitness function increment is negative are deleted. Finally, it updates N_s and continues to repeat the above steps until N_s is ϕ(Algorithm 3).

4.4. Community Optimization. The community needs to be optimized; i.e., the free nodes are allocated to the community or allowed to form a community independently and communities with higher similarity are merged. The optimization is mainly divided into two steps: the first step is to calculate the similarity S_{nc} of the node to each community. The node is added to the community if S_{nc} is greater than the threshold ϵ; otherwise, it forms an independent community. The second step is to calculate the similarity S_{cc} between the communities. The communities are merged if S_{cc} is greater than the threshold ϵ.

4.5. Community Evolution. Since the division of the time window directly affects the quality of community extraction, the result of the division is particularly important for subsequent analysis. The time span is usually one year or several months. There is no indicator to evaluate what kind of choice is optimal. We choose to partially overlap the social network data of adjacent time windows, usually overlapping 50% [45]. It ensures that the network topology similarity in adjacent time windows is greatly increased and the extracted evolution events are also increased. The evolutionary evaluation method improved by this article is used to judge the evolution results (Algorithm 4).

5. Experiment Results and Analysis

In this section, several experiments are used to analyze and verify the effectiveness of the proposed algorithm. The datasets are synthetic network and real network datasets. The effectiveness of the ILE algorithm is analyzed by modularity (EQ) and standard mutual information (NMI). On the synthetic network dataset, this article sets the network parameters and compares LFM, SLPA, DEMON, and L-Attractor overlapping community detection algorithms. We analyze the performance and efficiency of the ILE

Input: Network $G(V, E)$, ρ
(1) $\text{Seed}_s = \phi$; $I = 0$;
(2) FOR EACH $v \in V$;
(3) $S_{\text{score}}(v) = I_{\text{score}}(v)$; //According to formula (1)
(4) END FOR
(5) $S_{\text{core}} = \phi$;
(6) FOR EACH $v \in V$;
(7) $l_{\text{num}} = 0$; $n_{\text{num}} = |z(v)|$;
(8) FOR EACH $z \in z(v)$;
(9) IF $S_{\text{score}}(v) > S_{\text{score}}(z)$ THEN $l_{\text{num}} += 1$;
(10) END FOR
(11) IF $(l_{\text{num}}/n_{\text{num}} > \rho)$ THEN $S_{\text{score}} = S_{\text{score}} \cup v$;
(12) END FOR
(13) FOR EACH $m \in N_s(S_{\text{core}})$;
(14) Calculate M_{ij}; //Markov dynamics method;
(15) IF $\lambda_s^l(i) > a$ THEN $\text{Seed}_s = S_{\text{core}} \cup m$;
(16) END FOR
Output: Seed community set Seed_s

Algorithm 1: Seed community detection.

Input: Seed community set Seed_s
(1) $\text{Seed}_{sm} = \phi$;
(2) FOR EACH $s \in \text{Seed}_s$;
(3) IF $\text{Seed}_{sm} = \phi$ THEN $\text{Seed}_{sm} \cup s$;
(4) condition = True;
(5) FOR EACH $s_m \in \text{Seed}_{sm}$;
(6) Sim = $S_{cc}(s, s_m)$; //According to Definition 4
(7) IF Sim $> \epsilon$ THEN $S_{\text{mer}} = s \cup s_m$;
(8) $\text{Seed}_{sm} = \text{Seed}_{sm} \cup S_{\text{mer}}$;
(9) $\text{Seed}_{sm} = \text{Seed}_{sm} - s_m$;
(10) condition = False;
(11) BREAK;
(12) END IF
(13) END FOR
(14) IF condition THEN $\text{Seed}_{sm} = \text{Seed}_{sm} \cup s$;
(15) END FOR
Output: The merged Seed community set Seed_{sm}

Algorithm 2: Similar seed communities merger.

algorithm. In the real network, we select different threshold parameters to observe the change of community detection results. We compare the above four algorithms to analyze the proposed ILE algorithm. In order to verify the effectiveness of the proposed evolutionary algorithm, this article uses DBLP and Enron datasets and selects the current representative community evolutionary algorithm for comparative analysis. Experiments show that the ILE algorithm and its evolution algorithm have good performance and can get good community detection results and evolution results.

5.1. Detection on Synthetic Network. Most networks do not have a natural community detection scale in the actual scene. Many scholars use the data of synthetic artificial networks to verify the algorithm. In this article, we use the LFR [6] network because its parameter setting conditions are more in line with the real network application scenario and more suitable for the experimental analysis of the proposed algorithm. The meanings of LRF parameters are shown in Table 1.

μ represents the hybrid parameter in the network. It can adjust the proportion of edge connections between nodes inside the community and nodes outside the community. The range is generally between 0 and 1. The more obvious the network community structure is, the smaller it is.

Considering that the LFM algorithm randomly selects the initial node, this article obtains three different synthetic datasets according to the parameter settings of the artificial network, as shown in Table 2.

In the case of different hybrid parameters μ on the D1 dataset, observing the community detection results of various algorithms, the overlapping modularity EQ of each algorithm is shown in Figure 1.

Input: Network $G(V, E)$, Seed set Seed_{sm}
(1) $C = \phi$;
(2) FOR EACH $s \in \text{Seed}_{sm}$;
(3) $C_s = s$;
(4) Calculate $N_s(C_s)$; //According to Definition 1
(5) WHILE $N_s(C_s) \neq ;$
(6) List = ϕ;
(7) FOR EACH $z \in N_s(C_s)$;
(8) Sim = $S_{nc}(z, C_s)$; //According to Definition 3
(9) IF Sim > ϵ THEN List = List $\cup z$;
(10) END FOR
(11) FOR EACH $v \in$ List;
(12) IF $f_g^+ > f_g(C_s)$ THEN $C_s = C_s \cup v$;
(13) FOR EACH $z \in C_s$;
(14) $f_g^- = f_g(C_s - z)$;
(15) IF $f_g^- > f_g(C_s)$ THEN $C_s = C_s - z$;
(16) END FOR
(17) END IF
(18) END FOR
(19) Recalculate $N_s(C_s)$;
(20) END WHILE
(21) $C = C \cup C_s$;
(22) END FOR;
Output: Overlapping Community collection C.

Algorithm 3: Community expansion.

Input: Network $G(V, E)$, α, β
(1) C = calculate overlapping community; $V = \phi$;
(2) FOR EACH $C_s \in C$;
(3) FOR EACH $C_1, C_2 \in C_s$;
(4) $\max_Q = \max(q_1, q_2)$; //According to Definition 10
(5) END FOR
(6) FOR EACH $C_i \in C_s$;
(7) Calculate I_{C_i};
(8) END FOR
(9) FOR EACH $C_1, C_2 \in C_s$;
(10) Calculate Dis; //According to formula (5)
(11) END FOR
(12) $ve = S_{cc}(C_t, C_{t+1})$; //According to Definition 4
(13) $V = \cup ve$;
(14) END FOR
Output: Number and type of evolutionary events

Algorithm 4: Community evolution.

Table 1: LFR benchmark network generation parameters.

Parameter	Implication
N	Number of nodes
k	Average degree of node
$\max k$	Maximum degree of node
$\min c$	Number of nodes in the minimum community
$\max c$	Number of nodes in the maximum community
μ	Hybrid parameter
O_n	Number of overlapping nodes
O_m	Number of communities of overlapping nodes

Table 2: Synthetic network dataset parameters.

Dataset	D1	D2	D3
N	600	600	10000 ~ 150000
k	20	20	20
$\max k$	60	60	60
$\min c$	20	20	20
$\max c$	100	100	100
μ	0.3 ~ 0.8	0.4	0.4
O_n	20	20	20
O_m	2	2 ~ 8	2

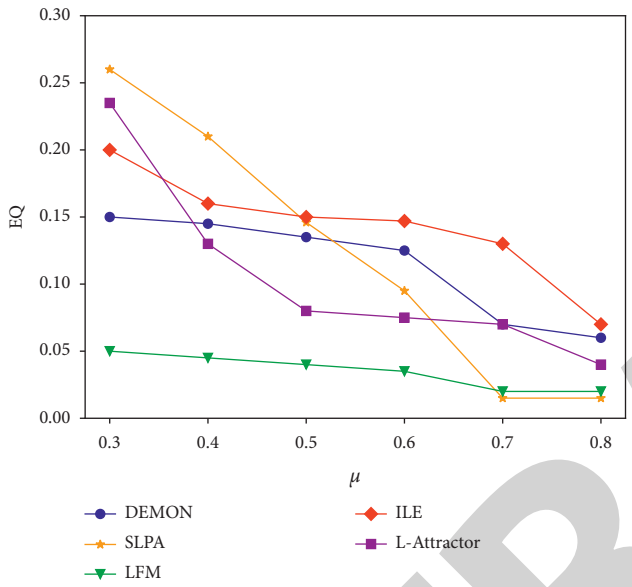

Figure 1: Modularity EQ of each algorithm under different hybrid parameters μ.

As can be seen from Figure 1, SLPA and L-Attractor achieve good results when the hybrid parameter μ is small. The overall performance of SLPA decreased significantly when $\mu > 0.3$. ILE algorithm is best when $\mu = 0.3$. The stability of the ILE algorithm becomes more and more prominent with the increase of μ. It shows that our algorithm has good adaptability to more complex networks. LFM algorithm is worst when the initial nodes are selected randomly. DEMON algorithm expands all network nodes as initial nodes and it is easy to form multiple independent communities, resulting in low accuracy.

Only when communities O_m to which different overlapping nodes belong change on the D2 dataset, a composite network with seven overlapping community structures is generated. The overlapping modularity EQ of each algorithm is shown in Figure 2.

As O_m increase, the modularity EQ of ILE and LFM decreases steadily. The SLPA is best when the number of communities O_m is equal to 2. The ILE algorithm has the advantage of stability when $O_m > 2$. It is more suitable for complex topology networks. This is because ILE algorithm uses a random walk model to initialize the core community and will not fall into local optimization. LFM and L-Attractor show stability to some extent. They are not as effective as the ILE algorithm. The reason for the poor performance of the DEMON algorithm is that it detects many independent communities.

By changing the number of network nodes on the D3 dataset, the running speed of the ILE algorithm for different scale datasets is verified. In this article, the experimental analysis is carried out according to the interval of every 50,000 nodes. The running time comparison of various algorithms is shown in Figure 3.

We find that the SLPA algorithm based on label propagation has the highest time efficiency. The proposed ILE algorithm adds the judgment of node importance and similarity threshold in seed community selection and community expansion. The time efficiency has also been significantly improved. DEMON is sensitive to the size of the dataset. The larger the size, the worse the effect. The time efficiency of DEMON is the lowest. LFM algorithm randomly selects the initial nodes and expands the whole network. It is relatively time-consuming. L-Attractor algorithm has carried out the conversion calculation of the link graph twice, so the time overhead is large.

In addition to the above, we compare different threshold parameters ρ. The NMI of ILE algorithm with different parameter ρ is obtained on the D1 artificial synthesis network. The results are shown in Figure 4.

The threshold parameter ρ determines the selection of the initial node; i.e., it adjusts the ratio of the number l_num of node importance greater than the importance of neighbor nodes to the total number n_num of neighbor nodes. Where the value range of l_num is (0,1), the variation range of the parameter ρ is (0,1). Observing the value of NMI, it can be found that although parameter ρ is different, the performance is roughly the same. It indicates that the selection of the parameters ρ has little effect on the results of the ILE algorithm. The result is very poor when the value of ρ is equal to 1. We can find that it is not easy to find the appropriate initial node when the selection of ρ is large. This leads to a decline in the quality of seed nodes, further reducing the accuracy of the algorithm.

5.2. Detection on Real Network. In the real network, Karate public dataset is selected to verify the effectiveness of the ILE algorithm. Karate dataset is the network of Karate Taekwondo clubs. The data includes 34 nodes and 78 edges. The node represents the members of the club, and the edge represents the friend relationship between each member. On the Karate dataset, we select different thresholds ρ and observe the change of data detection results. The results of the ILE algorithm are shown in Figure 5.

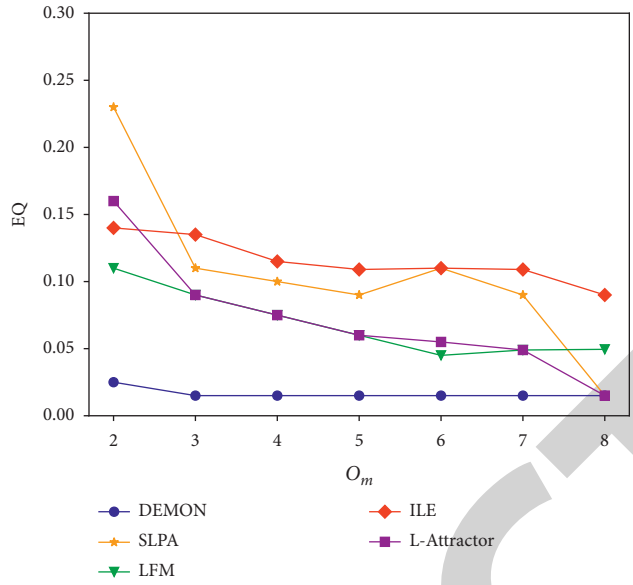

Figure 2: Modularity EQ of each algorithm under different O_m.

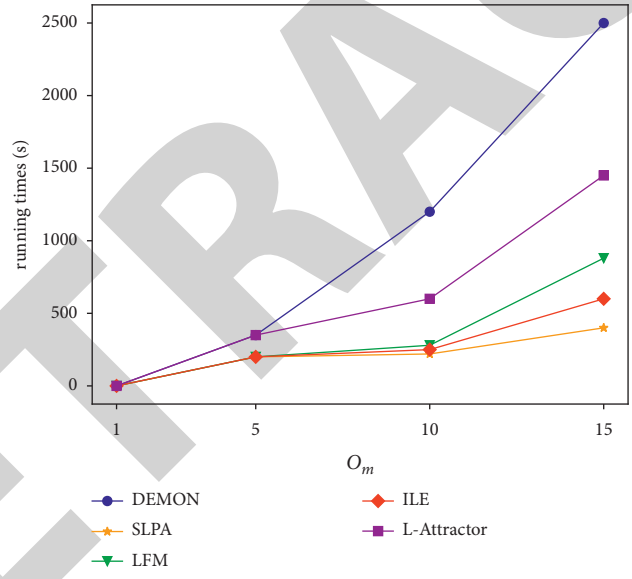

Figure 3: Comparison of running time of ILE algorithm on different scale datasets.

In Figure 5, colors other than green represent the overlapping nodes. And Figures 5(a–d) represent the detection results under $\rho = 0.2, 0.5, 0.9$, and 0.7 respectively. By comparing different parameter ρ, it can be seen that the result of the ILE algorithm is very similar to the actual result of Karate when $\rho = 0.7$. ILE detects three overlapping community nodes, node 1, node 9, and node 31. When ρ is equal to 0.2, 0.5, and 0.9, respectively, it is found that different thresholds ρ have little impact on the overall structure of Karate community detection, but there are great differences in the selection of overlapping nodes. It also shows that the proposed ILE algorithm has certain stability to the network structure itself. Due to different thresholds ρ, the quality of the initial nodes is different, and finally, the overlapping nodes after community detection are different.

The modularity EQ of the ILE algorithm, LFM algorithm, SLPA algorithm, DEMON algorithm, and L-Attractor algorithm running on a real network are shown in Table 3.

From the perspective of modularity EQ, DEMON and LFM perform poorly. In some networks, SLPA and L-Attractor have achieved good results. In most cases, the ILE algorithm proposed in this article has achieved good results. It shows that the ILE algorithm can correctly divide nodes into corresponding communities. The performance of LFM is not very good. It randomly selects the nodes as seed nodes so that the results are different every time, resulting in not finding enough complete subgraphs to cover the whole network.

The NMI of various algorithms on the different datasets is shown in Figure 6.

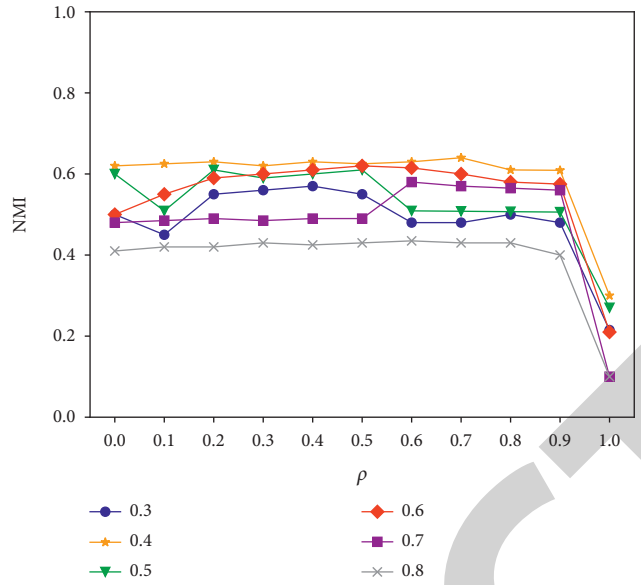

Figure 4: NMI of ILE algorithm under different threshold parameters ρ.

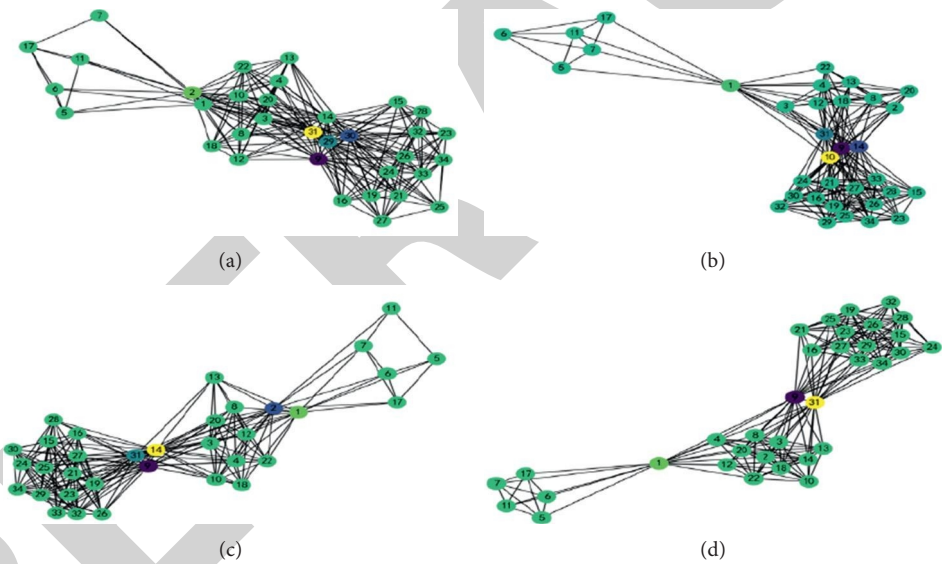

Figure 5: The detection results on Karate dataset under different parameters ρ. (a) $\rho = 0.2$. (b) $\rho = 0.5$. (c) $\rho = 0.9$, (d) $\rho = 0.7$.

Table 3: EQ of real complex networks.

Dataset	Karate	Dolphins	Football	Facebook
ILE	0.243	0.265	0.388	0.168
SLPA	0.253	0.182	0.423	0.155
LFM	0.203	0.206	0.352	0.095
DEMON	0.132	0.143	0.115	0.125
L-Attractor	0.245	0.176	0.354	0.135

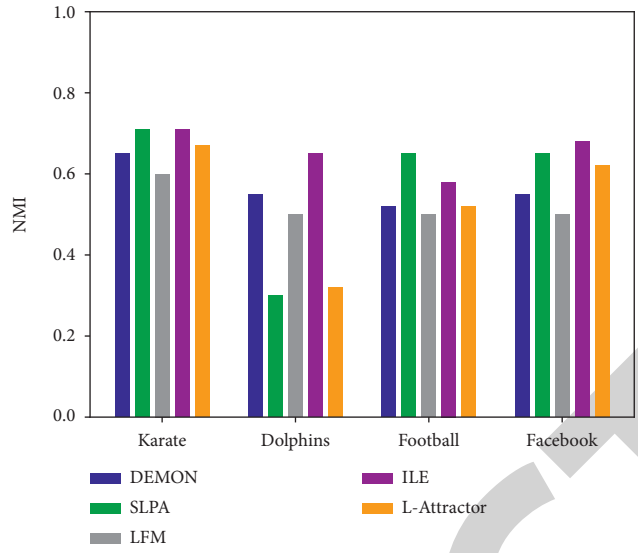

Figure 6: NMI of each algorithm on real network.

We can see that the NMI of the ILE algorithm is slightly lower than that of the SLPA algorithm on the football dataset, but it has stable performance on other real network datasets. It shows that the ILE algorithm has a good detection effect on the whole community and can effectively discover the community structure of the real network, attributing to the high-quality initial nodes selected by the node relevance centrality.

5.3. Evolution Results. This article analyzes the structure and characteristics of the complex network community and preprocesses the DBLP and Enron datasets to verify the proposed evolutionary algorithm. The DBLP dataset includes 497,014 pieces of data, representing the citation relationship between different article authors. The experimental data are from 2001 to 2010 and divided into ten time slice snapshots. Each time snapshot is set as 1 year. The Enron dataset describes the data information exchanged by Enron employees. The data of the whole year of 2001 are selected, including 2359 employee nodes and 136,876 e-mail messages. Moreover, it is divided into 12 time slices by month.

The ILE algorithm detects communities structure in different networks. For DBLP datasets, the time snapshot interval is set to 1 year with 10 snapshots. The detection results of different time snapshots are shown in Figure 7. For the Enron dataset, the time snapshot interval is set to 1 month with 12 snapshots. The detection results of different time snapshots are shown in Figure 8.

Figure 7 records the number of communities detected in the DBLP network from 2001 to 2010. It can be found that the number of communities changes over time. The number of communities showed an upward trend before 2010.

For Enron datasets, we select the overlapping data of adjacent time slices and set community data overlap to 50% considering the relatively small amount of data. As can be seen in Figure 8, the number of communities shows uncertain dynamic changes over time.

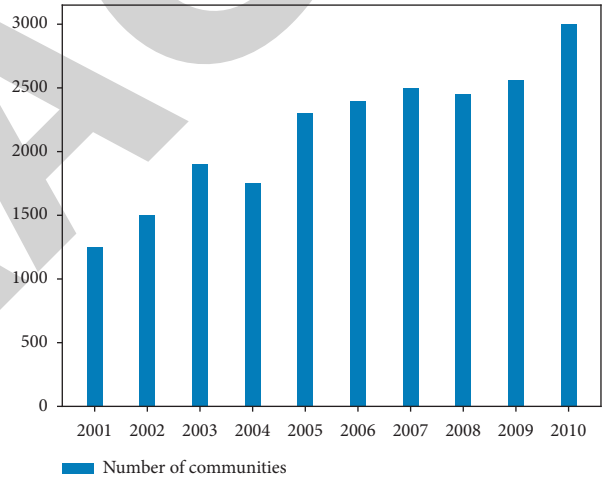

Figure 7: Number of communities per year in the DBLP network.

In the evolutionary algorithm of this article, the optimal parameters α, β, and γ are obtained by selecting different parameters many times and analyzing the experimental results.

It can be observed from Figure 9 that the amount of events such as continuing, growing, shrinking, merging, and dissolving is small when the conditions are stricter, i.e., when α is smaller and β is larger. This is because α and β are inversely proportional when setting the conditions of the evolution model. The shrinking event can not be detected and the number of detected growing, merging, splitting, and other events decreases linearly if β exceeds 0.2. In the experiment, α and β are set to 0.2 and γ is set to 0.3, which increased by 0.1 on α. The setting is optimal. The type and number of evolution events detected by the proposed algorithm are the best. Like the Enron network, the optimal

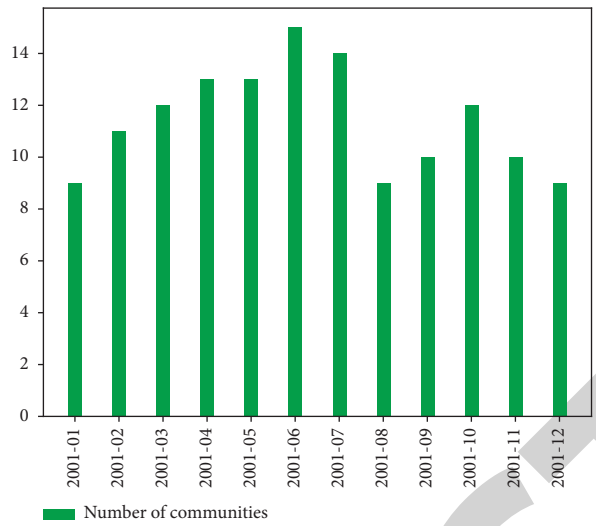

Figure 8: Number of communities per month in the Enron network.

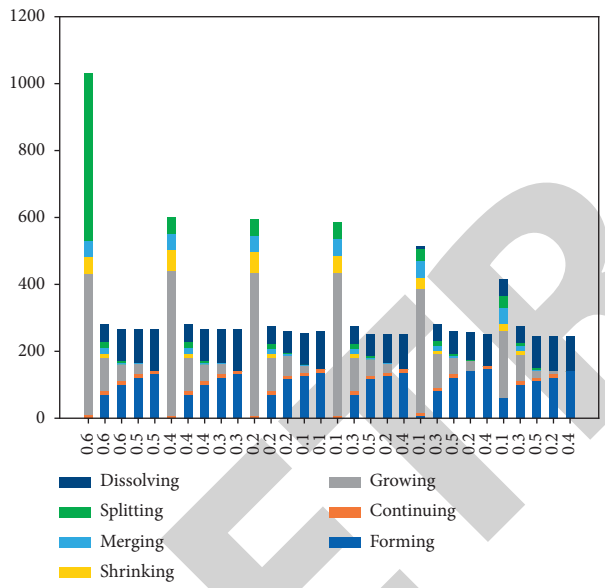

Figure 9: Accumulation diagram of evolution results of different parameters in Enron network. The abscissa represents the value of parameter α and parameter β, respectively, and the ordinate represents the number of community evolution events.

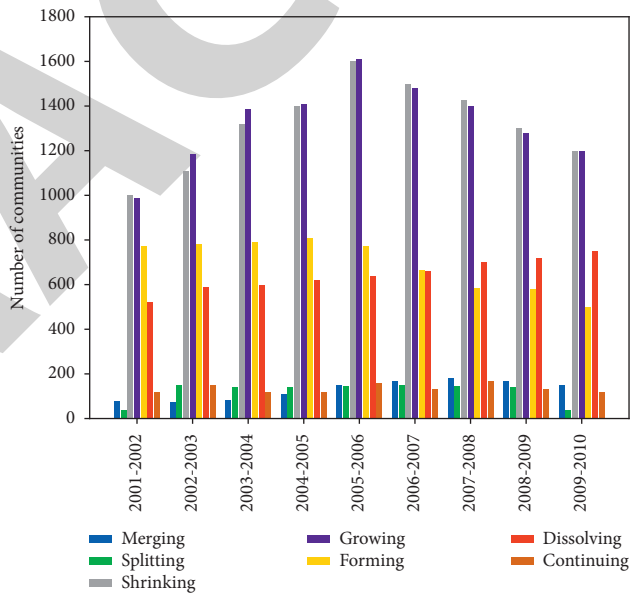

Figure 10: Community evolution events in DBLP network.

parameters in the DBLP network are 0.3, 0.2, and 0.4, respectively. Figure 10 shows the trend of community evolution time over time in the DBLP network.

It can be seen that growing events gradually decrease and dissolving events gradually increase over time. After years of collecting DBLP data, the separated and new members have reached a relatively stable state. The frequency of various evolutionary events presents a stable trend. It can also be found that the events of shrinking and growing detected by the proposed algorithm will have a high frequency, which is more consistent with the actual phenomenon.

The proposed evolutionary algorithm is compared with GED [37], MODEC [46], and Tajeuna [47]. The experimental results on the Enron dataset are shown in Table 4.

By comparing the results, it is found that the GED algorithm can not detect the forming events and dissolving events. The reason is that the Enron dataset is relatively small, so GED can not extract each evolution event well. Tajeuna algorithm can detect all kinds of evolution events as a whole, but few splitting events are detected. Moreover, MODEC algorithm detects too few continuing events. Compared with the above algorithms, our algorithm detects fewer continuing events. This is because we set the threshold condition of the growing event to be greater than 90%, so the detection conditions of the continuing, growing, and shrinking events do not intersect completely. It is a harsh condition, but this threshold setting matches the actual data. Generally, the evolutionary detection model proposed in this article can extract various evolutionary events well, and it has a better community evolutionary detection ability.

Table 4: Evolution results of different algorithms in Enron network.

Algorithm	Tajeuna [47]	MODEC [46]	GED [37]	Ours
Forming	162	201	0	170
Continuing	45	6	37	21
Growing	222	130	140	212
Shrinking	25	45	164	35
Merging	120	76	236	53
Splitting	13	33	3	46
Dissolving	168	53	0	201

6. Conclusion

In the article, an overlapping community detection algorithm (ILE) and its evolution algorithm are presented in the mobile opportunity network. Based on node relevance centrality and local expansion, it can detect the community structure of network. Firstly, it calculates the influence score of each node and finds the most influential node in the network as the core seed. Then, it forms a seed community together with its closely connected neighbors. Secondly, it merges similar seed communities to reduce counting and calculates the similarity between the nodes in the neighbor set of seed community and the community. Thirdly, it uses the fitness function to extend the community. Finally, it optimizes the community by adding nodes do not belong to any community in the network to the community with the highest similarity and merging the communities to improve the quality of community detection. Compared with other algorithms on real and artificial datasets, the proposed algorithm can accurately detect overlapping nodes while having approximate linear time complexity and can detect overlapping communities effectively and stably.

Data Availability

The datasets used to support this study are obtained from http://snap.stanford.edu/data/.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (nos. 61 772 096, 61 876 201, and 61 876 027) and in part by the National Natural Science Foundation of Chongqing (no. cstc2019jcyj-cxttX0002).

References

[1] 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.

[2] R. V. Driessche and D. Roose, "Dynamic load balancing with an improved with an improved spectral bisection algorithm," in *Proceedings of the IEEE Scalable High Performance Computing Conference*, pp. 494–500, Knoxville, TN, USA, May 1994.

[3] 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.

[4] M. E. Newman, "Fast algorithm for detecting community structure in networks," *Physical review. E, Statistical, nonlinear, and soft matter physics*, vol. 69, no. 6, Article ID 66133, 2004.

[5] A. Clauset, M. E. Newman, and C. Moore, "Finding community structure in very large networks," *Physical review. E, Statistical, nonlinear, and soft matter physics*, vol. 70, no. 6, Article ID 66111, 2004.

[6] A. Lancichinetti, S. Fortunato, and J. Kertész, "Detecting the overlapping and hierarchical community structure in complex networks," *New Journal of Physics*, vol. 11, no. 3, Article ID 33015, 2009.

[7] M. Coscia, G. Rossetti, and F. Giannotti, "DEMON: a local-first detection method for overlapping communities," in *Proceedings of the Eighteenth ACM SIGKDD International Conference On Knowledge Discovery And Data Mining*, pp. 615–623, Beijing, China, August 2012.

[8] 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 Eleventh International Conference on Data Mining Workshops*, pp. 344–349, Vancouver ,Canada, December 2011.

[9] P. Kazienko, "Process of Social Network Analysis," *Encyclopedia Of Social Network Analysis And Mining*, Springer, New York, NY, USA, pp. 1418–1432, 2014.

[10] M. Cordeiro, R. P. Sarmento, and J. Gama, "Dynamic community detection in evolving networks using locality modularity optimization," *Social Network Analysis and Mining*, vol. 6, no. 1, 2016.

[11] G. Costa and R. Ortale, "Integrating overlapping community detection and role analysis: Bayesian probabilistic generative modeling and mean-field variational inference," *Engineering Applications of Artificial Intelligence*, vol. 89, Article ID 103437, 2020.

[12] G. Yin, K. Chi, Y. Dong, and H. Dong, "An approach of community evolution based on gravitational relationship refactoring in dynamic networks," *Physics Letters A*, vol. 381, no. 16, pp. 1349–1355, 2017.

[13] U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," *Physical review. E, Statistical, nonlinear, and soft matter physics*, vol. 76, no. 3, Article ID 36106, 2007.

[14] Z. Anna, E. Nawarecki, J. Koźlak, and A. Mika, "Determining life cycles of evolving groups," *Procedia Computer Science*, vol. 35, pp. 1102–1111, 2014.

[15] 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.

[16] J. M. Kumpula, M. Kivelä, K. Kaski, and J. Saramäki, "Sequential algorithm for fast clique percolation," *Physical review. E, Statistical, nonlinear, and soft matter physics*, vol. 78, no. 2, Article ID 26109, 2008.

[17] S. Zhang, X. Ning, and X. S. Zhang, "Identification of functional modules in a PPI network by clique percolation clustering," *Computational Biology and Chemistry*, vol. 30, no. 6, pp. 445–451, 2006.

[18] Z. Zhou, Z. Xiao, and W. Deng, "Improved community structure discovery algorithm based on combined clique percolation method and K-means algorithm," *Peer-to-Peer Networking and Applications*, vol. 13, no. 6, pp. 2224–2233, 2020.

[19] L. Donetti and M. A. Muñoz, "Detecting network communities: a new systematic and efficient algorithm," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2004, no. 10, Article ID P10012, 2004.

[20] 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. 1, Article ID 16105, 2009.

[21] L. Chen, J. Zhang, and L. Cai, "Overlapping community detection based on link graph using distance dynamics," *International Journal of Modern Physics B*, vol. 32, no. 3, 2018.

[22] J. Cao, S. Wang, and H. Wang, "Detecting communities on topic of transportation with sparse crowd annotations," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 4, pp. 1017–1022, 2017.

[23] Y. Zhou, G. Sun, R. Zhou, and Z. Wang, "Local Community Detection Algorithm Based on Minimal Cluster," *Applied Computational Intelligence and Soft Computing*, vol. 2016, Article ID 3217612, 11 pages, 2016.

[24] T. Chakraborty, S. Kumar, N. Ganguly, A. Mukherjee, and S. Bhowmick, "GenPerm: a unified method for detecting non-overlapping and overlapping communities," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 8, pp. 2101–2114, 2016.

[25] X. Wang, G. Liu, and J. Li, "Overlapping community detection based on structural centrality in complex networks," *IEEE Access*, vol. 5, Article ID 25258, 2017.

[26] K. Yang, Q. Guo, S. N. Li, J. T. Han, and J. G. Liu, "Evolution properties of the community members for dynamic networks," *Physics Letters A*, vol. 381, no. 11, pp. 970–975, 2017.

[27] H. Yu, L. Jin, B. Zhou, B. Xiao, and X. Zeng, "An Event-Based Approach to Overlapping Community Evolution by Three-Way Decisions," in *Proceedings of the 2017 IEEE Second International Conference on Big Data Analysis (ICBDA)*, pp. 772–778, Beijing, China, March 2017.

[28] J. Zhu, J. Liu, X. Zhang, and Y. Zhao, "A Reconstructed Event-Based Framework for Analyzing Community Evolution," in *Proceedings of the 2016 IEEE International Conference On Big Data Analysis (ICBDA)*, pp. 1–4, Hangzhou, China, March 2016.

[29] S. Y. Bhat and M. Abulaish, "HOCTracker: tracking the evolution of hierarchical and overlapping communities in dynamic social networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 4, pp. 1019–1013, 2015.

[30] Z. Dhouioui, R. Toujani, and J. Akaichi, "Tracking Dynamic Community Evolution and Events Mobility in Social Networks," *Encyclopedia of Social Network Analysis and Mining*, Springer, New York, NY, USA, pp. 3159–3170, 2018.

[31] R. Karan and B. Biswal, "A model for evolution of overlapping community networks," *Physica A: Statistical Mechanics and Its Applications*, vol. 474, pp. 380–390, 2017.

[32] W. Yu, W. Wang, P. Jiao, H. Wu, Y. Sun, and M. Tang, "Modeling the local and global evolution pattern of community structures for dynamic networks analysis," *IEEE Access*, vol. 7, Article ID 71350, 2019.

[33] Z. Wang, Z. Li, G. Yuan, Y. Sun, X. Rui, and X. Xiang, "Tracking the evolution of overlapping communities in dynamic social networks," *Knowledge-Based Systems*, vol. 157, pp. 81–97, 2018.

[34] B. Gliwa, S. Saganowski, A. Zygmunt, P. Bródka, P. Kazienko, and J. Kozak, "Identification of Group Changes in Blogosphere," in *Proceedings of the 2012 IEEE/ACM International Conference On Advances In Social Networks Analysis And Mining*, pp. 1201–1206, Istanbul, Turkey, August 2012.

[35] Y. Hong, C. Fu, Q. Huang, Z. Fang, J. Zeng, and L. Han, "Communities evolution analysis based on events in dynamic complex network," in *Proceedings of the 2016 IEEE 14th Intl Conf on Dependable, Autonomic and Secure Computing, 14th Intl Conf on Pervasive Intelligence and Computing, 2nd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress(DASC/PiCom/DataCom/CyberSciTech)*, pp. 497–503, Auckland, New Zealand, August 2016.

[36] M. Takaffoli, F. Sangi, J. Fagnan, and O. R. Zäıane, "Community evolution mining in dynamic social networks," *Procedia - Social and Behavioral Sciences*, vol. 22, pp. 49–58, 2011.

[37] P. Bródka, S. Saganowski, and P. Kazienko, "GED: the method for group evolution discovery in social networks," *Social Network Analysis and Mining*, vol. 3, no. 1, pp. 1–14, 2013.

[38] C. Magnien and F. Tarissan, "Time evolution of the importance of nodes in dynamic networks," in *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pp. 1200–1207, Paris, France, August 2015.

[39] P. Jaccard, "Etude comparative de la distribution florale dans une portion des Alpes et des Jura," *Bulletin de la Societe Vaudoise des Sciences Naturelles*, vol. 37, no. 142, pp. 547–579, 1901.

[40] Y. Cui, X. Wang, and J. Li, "Detecting overlapping communities in networks using the maximal sub-graph and the clustering coefficient," *Physica A: Statistical Mechanics and Its Applications*, vol. 405, pp. 85–91, 2014.

[41] P. Kim and S. Kim, "Detecting overlapping and hierarchical communities in complex network using interaction-based edge clustering," *Physica A: Statistical Mechanics and its Applications*, vol. 417, pp. 46–56, 2015.

[42] G. Palla, A. L. Barabási, and T. Vicsek, "Quantifying social group evolution," *Nature*, vol. 446, no. 7136, pp. 664–667, 2007.

[43] H. Shen, X. Cheng, K. Cai, and M. B. Hu, "Detect overlapping and hierarchical community structure in networks," *Physica A: Statistical Mechanics and Its Applications*, vol. 388, no. 8, pp. 1706–1712, 2009.

[44] L. Danon, A. D. Guilera, J. Duch, and A. Arenas, "Comparing community structure identification," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2005, no. 9, Article ID P09008, 2005.

[45] S. Ghosh and N. Ganguly, "Structure and evolution of online social networks," *Intelligent Systems Reference Library*, vol. 65, pp. 23–44, 2010.

[46] M. Takaffoli, J. Fagnan, F. Sangi, and O. R. Zaïane, "Tracking Changes in Dynamic Information Networks," in *Proceedings of the 2011 International Conference on Computational Aspects of Social Networks (CASoN)*, pp. 94–101, Salamanca, Spain, October 2011.

[47] E. G. Tajeuna, M. Bouguessa, and S. Wang, "Tracking the Evolution of Community Structures in Time-Evolving Social Networks," in *Proceedings of the 2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 1–10, Paris, France, October 2015.