

An Improved Algorithm based on Time Domain Network Evolution

Guanghui Yan^{a,*}, Qingqing Ma^a, Yafei Wang^a, Yu Wu^a, and Dan Jin^b

^a*School of Electronic and Information Engineering, Lanzhou Jiaotong University, Lanzhou, 730070, China*

^b*Gansu Electric Power Information Communication Centre, Lanzhou, 730070, China*

Abstract

Community evolution is the highlight in the field of complex network. The current typical tracking community algorithms largely focus on adopting the traditional similarity functional measurements to capture the similarity between communities at temporal snapshots. However, it doesn't take into account the actions accumulated with the events and the effects of community members in evolutionary networks. Meanwhile, different communities use traditional tracking methods with a simple similarity function, and as a result, many analogous communities cannot be effectively extracted in the network. To address these shortcomings, in this paper, we propose a much more powerful similarity function to catch and evaluate communities or groups in a successive time frame. We implement a community tracking method in our new function on the basis of previous research, in which we improve accuracy in network structure by taking the diversity corresponding to the active node in network-evolution into consideration. Finally, we find an interesting phenomenon and give a new method to weigh out the relationships involving active nodes within community evolution over time frames. Eventually, the performance of our algorithm is measured by applying it to real datasets and it is tested on tracking community structure and assessing the experimental results that inhibit active nodes extracted from the community. The experimental results show that our algorithm can effectively keep track of community structure and outperform other algorithms.

Keywords: temporal networks; community evolution; similarity function; active node

(Submitted on January 18, 2018; Revised on March 9, 2018; Accepted on April 21, 2018)

© 2018 Totem Publisher, Inc. All rights reserved.

1. Introduction

Complex network, which is a powerful method to model the complex system by viewing the individual elements of systems as nodes and the relations between individual events as edges in graphs, has become a hot topic in recent years. The method of complex networks can be traced back to the ER random graph model that was established by Erdos and Renyi [10]. They proposed various important properties of ER random graphs in their research work. In 1998, the WS small world model was proposed by Watts and Strogatz [18] in terms of a large number of real-world network topologies. Albert-Laszlo and Barabasi investigated the scale-free feature of complex networks and proposed the BA-scale-free model [2]. Due to the introduction of the WS small world model and the BA scale-free model, the study of complex networks has entered a new era. In 2002, Girvan and Newman discovered that there are community structures in complex networks [13], that is, nodes are connected closely in the same community and sparsely connected among different communities. Moreover, Fortunato and other scholars adopted a novel definition of community [12] to promote the efficiency of community discovery in complex networks. As a result, a large number of community detection methods and community quality assessment models have emerged. In 2007, Gergely and other scholars gave a new algorithm based on clique percolation, which allowed investigation of the time dependence of overlapping communities on a large scale to uncover basic relationships for characterizing community evolution. It also proposed the six basic evaluation standards of the community in the process of evolution [1].

Community is usually described dynamically rather than firmly according to construction structure (the nodes and edges). In essence, the network structure in the real world will change with time. It reflects the events or changes about the network properties [4]. Therefore, tracking and detecting communities in complex networks is an effective method for complex network evolution [8]. In contrast to traditional similarity functions, we propose a more dominant pattern, which considers

* Corresponding author.

E-mail address: yanguanghui@mail.lzjtu.cn

the exclusiveness and inclusiveness of distinct communities at successive snapshots in order to track community. It performs better than other algorithms [1,9,11]. Consecutively, we distinguish the evolutionary relationships between two different communities over time slices so that we can effectively track the temporal network and find the community structure that truly participates in evolution. In this paper, we also present a new denotation called Active Node, which describes the common members of a temporary community that contributes to network constructional structure changes in successive window time slices. On the basis of the above similarity function, we find an interesting phenomenon between the relationships involving active nodes within community evolution over time in DBLP datasets. Eventually, the experiments show that the algorithm can better track the evolution of the network structures and can detect more underlying behaviors.

This paper is organized as follows: in the next section, we first introduce the research trends in this field and the improvement of the algorithm in this paper. In section 3, we present propose a new similarity function and community tracking method based on active nodes. In Section 4, we give an exhaustive experiment to test the actual datasets and provide a comparative analysis. Finally, in Section 6, we draw our conclusions and sketch possible future improvements of our research.

2. Related Work

In this section, we compare proposed methods for tracking community and discuss the implements of community evolution.

There are quite a few perfect research results in community tracking [15] that introduced several measurements to evaluate the degree of web community evolution, such as growth rate, novelty, and stability, etc. Hopcroft [5] obtained the community structure through a cooperative network and found a fixed structure in the network by randomly moving 5% of the members of the network, which was called a natural community. Gergely [1] tracked the community by changing the topology of networks over time, giving a specific definition of community morphological changes in community evolution for at the first time. Asur [9] studied the changes in the structure, characteristics, and behavior of community evolution. Greene and Doyle [11] adopted the generalized Jaccard coefficients for community tracking and proposed the concept of evolutionary interruption. Takaffoli's research team [3] improved the Jaccard coefficients to track community in different time slices. Saganowski [14] and other scholars used the observation of the time domain network to predict the evolution of the community. Goldberg [7] introduced the minimum common point of community evolution to track the community, presented three axioms of community evolution, and predicted community development according to axioms of community. Gauvin [17] detected the network community activity pattern by a method called non-negative tensor decomposition. Ley [16] identified the central structure of the community according to the degree of the nodes. As the above studies, we can perceive that the systematic concept of the community is not explicit, and much of the work often focuses on the evolution of the community with specific metric measurements in the early years. The research of the complex network has the following deficiencies:

- The previous tracking community evolution algorithms are partially considered. As shown in Figure 1, the black dot stands for individuals that have high degrees of nodes in the community that play an important role in tracking network structure. They maybe belong to some district community (they are not presented in figure). The white dot stands for common nodes during the evolutionary network. After a transient time frame, they perhaps insignificantly disappear. The evolution of the two processes will be calculated in the same way due to ignoring the differences between trends of changes. Though the selection of different similarity functions in different evolutionary forms carries out the evolutionary matching computation, these communities represent the expansion and contraction of the different behaviors of the two communities.

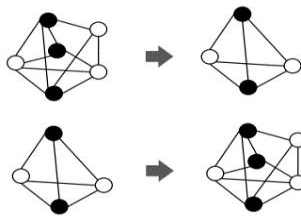


Figure 1. Two different forms of evolution

- The community tracking method gives some explanations, but it still lacks the objective standards. As in [6], the degree is used as the basis for the importance of the node in the community, though in the network of cooperative relations, some unimportant members may get a higher degree through a larger cooperation project. The community tracking of members affects the accuracy of the study.

Through the introduction above, we can see that the current community tracking algorithms still have some shortcomings in the evolutionary computation, which may bring some errors and issues in research. Therefore, we design a new community tracking method by adding the time domain information to similar functions. We consider the differences between the network and the nodes, and extract the active nodes in the network to carry out community tracking. As a class of open complex system, we extend the concept of traditional network survivability to large-scale networks. For a large-scale network system S , when one or more subsystems S_i suffer internal or external disturbances, such as external attack or system failure etc., S maintains continuous services through adaptation, configuration, restoration, and evolution etc., and makes the entire network avoid failure status. The behavioral characteristic is called survivability. Survivability is a fundamental characteristic of large-scale networks and would not disappear due to system evolution or external environment changes. In large-scale networks, there exists information exchange between the failed subsystems and other subsystems, and thus, this may cause cascaded failures. This increase of failed subsystems may cause the whole large-scale network to fail.

3. Preliminaries

As aforementioned, a general framework to deal with the issues of data reliability in complex networks is very important. Here, we present a framework and introduce the notations to present the issues we will resolve, which are mainly denoted according to community change features at dynamic window snapshots during their lifetime.

3.1. Notion Description

For a Network or Graph $G = \{V, E\}$, V is a set of vertexes and E is a set of edges. The dynamic network model established in this paper is composed of static snapshots of network G at time slices, which include $G = \{G_1, G_2, \dots, G_n\}$, (as shown in Figure 2). We use the index n to represent some time slices. Collection $C_t = \{C_t^1, C_t^2, \dots, C_t^n\}$ represents a set of communities found in graph G_t at time t . The symbols used in this paper are shown in Table 1.

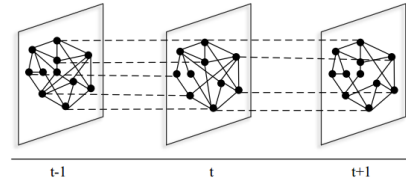


Figure 2. Temporal network model

Table 1. Notion and Description

| Notion | Description |
|--------------------------|--|
| $\text{Node}(V_t)$ | The set of G_t nodes |
| $\text{Edge}(E_t)$ | The set of G_t edges |
| $\text{Node}(C_t^i)$ | The collection of nodes in community C_t^i |
| $\text{Edge}(C_t^i)$ | The collection of edges in community C_t^i |
| $ \text{Node}(C_t^i) $ | The number of node in community C_t^i |
| $\text{Father}(C_t^i)$ | Direct Father community in community C_t^i |
| $\text{Child}(C_t^i)$ | Community C_t^i Direct child community collection |
| $ \text{Father}(C_t^i) $ | The number of members included in the direct father community collection for Community C_t^i |
| $ \text{Child}(C_t^i) $ | The number of members included in the direct child community collection for Community C_t^i |

There are two concepts we need to clearly describe. $\text{Child}(C_t^i)$ is represented as the Community C_t^i direct child community collection illustrated in Figure 3. We assume that two groups of static snapshots are tracked at different time frames t and $t-1$. Most of the nodes in group C_t^i are found in group C_{t-1}^i . In another word, group C_t^i and C_{t-1}^i have common nodes. But, our proposed operator denotes group C_t^i by means of all nodes in C_t^i , corresponding to common nodes in the forehead community C_{t-1}^i . This means that perhaps some nodes in C_t^i don't exist in community C_{t-1}^i . The operator $\text{Child}()$ obtains a series of sets that exist in C_t^i and not in C_{t-1}^i . The operator $\text{Father}()$ obtains a series of sets that exist in C_{t-1}^i and not in C_t^i at the next time snapshot window.

3.2. Community Evolution

To quickly capture the community evolution, we introduce six evolution events: birth, death, growth, contraction, merging, and splitting as proposed in paper [1]. Several forms of community evolution are shown in Figure 3. Symbolic representations of community evolution are as follows:

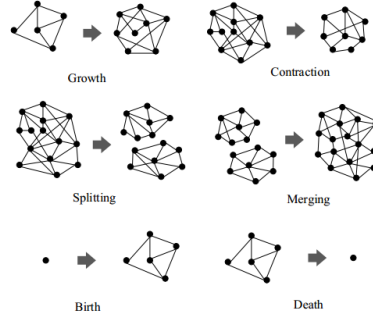


Figure 3. Basic forms of community evolution

- Community Birth: a community C_t^i , if the $|\text{Father}(C_t^i)|=0$, then the community C_t^i for the appearance of the community is in time t .
- Community Death: a community C_t^i , if the $|\text{Child}(C_t^i)|=0$, then the community C_t^i for the disappearance of the community is in time t .
- Community Growth: a community C_t^i , if the $\text{Father}(C_t^i)=\{C_{t-1}^i\}=0$ and $|\text{Node}(C_t^i)| \geq |\text{Node}(C_{t-1}^i)|$, then the community C_t^i evolved from community C_{t-1}^i through growth.
- Community Contraction: a community C_t^i , if the $\text{Father}(C_t^i)=\{C_{t-1}^i\}$ exists and $|\text{Node}(C_t^i)| < |\text{Node}(C_{t-1}^i)|$, then the community C_t^i evolved from community C_{t-1}^i through growth.
- Community Merging: a community C_t^i , if the $\text{Father}(C_t^i)=\{C_{t-1}^1, \dots, C_{t-1}^m\}$ exists and $\text{Child}(C_{t-1}^m)=\dots=\text{Child}(C_{t-1}^1)=\{C_t^i\}$, then the community C_t^i evolved from community $C_{t-1}^m, \dots, C_{t-1}^1$ through merging.
- Community Splitting: a community C_t^i , if the $\text{Child}(C_t^i)=\{C_{t+1}^1, \dots, C_{t+1}^m\}$ exists and $\text{Father}(C_{t+1}^m)=\dots=\text{Father}(C_{t+1}^1)=\{C_t^i\}$, then the community C_t^i splits into community $C_{t+1}^m, \dots, C_{t+1}^1$.

3.3. Community Similarity Function

Definition 1: Let C_m^i and C_n^j be communities that are detected at snapshot m and n ($m \neq n$). Equation (1) describes that the communities are similar if common nodes meet the following formulation:

$$\text{Sim}(C_m^i, C_n^j) = \begin{cases} \frac{|V_m^i \cap V_n^j|}{|V_m^i| * |V_n^j|} * \frac{|V_m^i \cap V_n^j|}{\min(|V_m^i|, |V_n^j|)}, & \text{if } s' \geq s \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

Where $s' = \frac{|V_m^i \cap V_n^j|^3}{|V_m^i| * |V_n^j| * \min(|V_m^i|, |V_n^j|)}$, s is a threshold that represents a margin precluding unpredictable or missed deviation through different snapshots. In this paper, we set s to be a number in the limit scope $[0.1, 0.3]$. We newly denote similarity by $\text{Sim}(C_m^i, C_n^j)$. In previous similarity functions [11,15], $\text{Sim}(C_m^i, C_n^j)$ is equal to $\frac{|V_m^i \cap V_n^j|}{\max(|V_m^i|, |V_n^j|)}$, which is only slightly precise in characterizing analogous communities or group. For example, as illustrated in Figure 3, we take a merging evolution to explain why it is not precise. Let C_m^i and C_n^j meet and note that $\text{Node}(C_m^i)$ is much greater than $\text{Node}(C_n^j)$ and $C_m^i \in C_n^j$. We can obtain that $\frac{|V_m^i \cap V_n^j|}{\max(|V_m^i|, |V_n^j|)}$ is approximately close to 0. But, we expect that value was too low. Our proposed similarity function can better feature the above issues. In an illustrated case, $\frac{|V_m^i \cap V_n^j|}{\min(|V_m^i|, |V_n^j|)}$ is equal to 1 and $\frac{|V_m^i \cap V_n^j|^2}{|V_m^i| * |V_n^j|}$ stands for C_m^i and C_n^j , which make up a proportion of share members corresponding to their respective number of nodes in the community. Consequently,

$\text{Sim}(C_m^i, C_n^j)$ reaches an acceptable compromise between two communities in which one side unevenly accounts for the others.

3.4. Active Nodes in network

In an evolutionary network, the nodes appear continuously or intermittently. What's more, the common nodes play a significant part in community structure changes at snapshot t and $t-1$. The individuals that appear at both time frames t and $t-1$ are defined as the Active Nodes. Specific definitions are as follows:

Definition 2: (active node) For Node set $V_t^{t-1} = \text{Node}(G_t) \cap \text{Node}(G_{t-1})$, we denote such a set V_t^{t-1} as an active node set that exists at $t-1$ and t moments. There is an obvious relationship: $V_t^{t-1} = V_{t-1}^t$. Equation (2) describes that there are a set of Active Nodes during their lifetime to make contributions to evolutionary community. So, we denote Active Node S_{active} Contributor Factor by $\text{ANCF}(S)$.

$$S_{\text{active}} = V_{t-m}^{i_1} \cap V_{t-m+1}^{i_2} \cap \dots \cap V_{t+n-1}^{i_{j-1}} \cap V_{t+n}^{i_j} \quad (2)$$

Definition 3: Active Node Contributor Factor. For a series of community snapshots, we can obtain a collection of nodes that exist over community evolution. It is formally formulated as the following:

$$\text{ANCF}(S, i) = \frac{t_n * \exp(\text{Sim}(C_i^m, C_{i+1}^n))}{\sum_{i=0}^{t_n} \exp(\text{Sim}(C_i^m, C_{i+1}^n))} \quad (3)$$

It is worth noting that S_{active} commonly behaves with some nodes, which means that active nodes play an important role in participating in community. According the Equation (3), we don't distinguish which active nodes are significant because they have the same ANCF value.

4. Algorithm

4.1. Algorithm of community tracking

Input: Graph G_t, G_{t-1}

Output: the evolution of communities in Graph G_t, G_{t-1}

1. Calculate the active point V_t^{t+1} in Graph G_t, G_{t-1}
2. Evaluate community collection C_t, C_{t+1} in Graph G_t, G_{t-1}
3. **for** $i=0:n-1$ in graph G_t **do**
4. **if** node(i) in C_t , but all nodes in C_t are not in V_t^{t+1}
5. **then**
6. C_t is labeled as "disappearing"
7. **end if**
8. **if** The nodes in V_t^{t+1} **then**
9. select t in C_t^i, t in C_t^j
10. **if** $\frac{\text{Node}(C_t^i)}{\text{Node}(C_{t+1}^j)} > l$ **then**
11. the community C_t^i and C_{t+1}^j are not similar
12. **else if** $|\text{Node}(C_t^i)| \geq |\text{Node}(C_{t+1}^j)|$ **then**
13. **if** $\frac{\text{Node}(C_{t+1}^j) \cup \text{Node}(C_t^i)}{\text{Node}(V_t^{t+1}) \cup \text{Node}(C_{t+1}^j)} > k$ **then**
14. the community C_t^i and C_{t+1}^j are not similar
15. **end if**
16. **else if** $|\text{Node}(C_t^i)| < |\text{Node}(C_{t+1}^j)|$ **then**

```

17.   if  $\frac{\text{Node}(C_{t+1}^j) \cup \text{Node}(C_t^i)}{\text{Node}(V_t^{t+1}) \cup \text{Node}(C_{t+1}^j)} > k$  then
18.       the community  $C_t^i$  and  $C_{t+1}^j$  are not similar;
19.   end if
20. end if
21. if  $C_t^i$  is similar to  $C_{t+1}^j$  then
    if  $\frac{\text{Node}(C_t^i)}{\text{Node}(C_{t+1}^j)} > l$  then
22.       the community  $C_t^i$  is labeled as "contraction"
23.   else
24.       the community  $C_t^i$  is labeled as "growth"
25.   end if
26. end if
27. if  $C_t^i$  is similar to one of  $C_{t+1}^m, \dots, C_{t+1}^n$  then
28.       the community  $C_t^i$  is labeled as "splitting"
29.   end if
30. if  $C_t^i$  is similar to  $C_{t+1}^i$  then
31.       the community  $C_t^i$  is labeled as "merging"
32.   end if
33. end if
34. end for

```

4.2. Complexity Analysis

Let the number of nodes in the network be N_t at t time, and denote the number of nodes in the network by N_{t+1} at $t+1$ time. V_k is the number of active nodes in the network. Time complexity: Step 1, calculate whether the nodes appear in G_{t+1} and not in G_t . The use of hash table gives the time complexity $O(n_{t+1})$. For Steps 2 and 3, t time for all nodes is traverse and the time complexity is $O(n_t)$. Each active node $v \in V_t^{t+1}$ must exist in the community $C_t^i \in G_t$ and $v \in C_t^i$, $C_t^j \in G_t$ and $v \in C_t^j$. Community C_t^i and community C_{t+1}^j may exist between the evolution of the need for similarity calculation. So, the part-time complexity is $O(v)$. For Steps 4 to 6, for each active node $v \in V_t^{t+1}$ that corresponds to community C_t^x and community C_{t+1}^y between the evolution of the need for similarity calculation, the time complexity is $O(v)$. Step 7: to determine a new network at time $t+1$ of the community, the time complexity is $O(n_{t+1})$. So, the overall time complexity of the algorithm is $O(n_t) + 2O(n_{t+1}) + O(V_k)$. Space complexity: The general hash table space complexity and the number of storage members have a linear relationship. Therefore, the spatial complexity of this paper is $O(n_t) + O(n_{t+1}) + O(V_k)$.

5. Experiments

DBLP is a very imperfect "authority file" for computer science researchers. It lists more than 3.66 million journal articles, conference papers, and other publications on computers science. All important journals on computers can be tracked. The DBLP datasets maintained by Stanford University is one of the classic datasets in data mining. The datasets contain papers published from 1936 to the present, including main conference journals in the computer field. We try to identify the authors behind research papers and treat synonyms and homonyms as precise as possible. We work on the DBLP datasets and compare with the MODEC algorithm and GP algorithm [1].

We use the datasets of the 2015.11.1 release. The metadata are pretreated by means of traversing all of $\langle \text{www} \rangle$ nodes and dealing with situations where the same person has different names. After that, the datasets are grouped according to different behaviors. According to the actual demand, the data generation network structure from 1980 to 2014 is extracted, and the algorithm is introduced to calculate the community.

5.1. Active nodes

We employ $\langle \text{article} \rangle$ labels as individual nodes in the network, and the two different articles in where common authors both

participated are relative. If there are links between articles because of several common authors, we regard it as a link. We only obtain relative groups despite every author composing multiple papers that are grouped into different communities. What's more, groups or communities that authors are involved in can diversify the research filed. Yet, the evolution reflects the researcher's work changes. So, a big network is given by the above operations. Based on the DBLP datasets <article> nodes, the experiment calculates the total data of 35-time slices from 1980 to 2014 and validates the importance of active nodes. The concrete results are shown in Figure 4. Figure 4 represents the statistics of the growth of network scale. Experimental results show that the speed of the network grow from low to high, and the network scale grows gradually from 4761 nodes in 1980, to 195096 nodes in 2013. The number of inactive nodes and active nodes increased steadily each year, and the number of active nodes increased from 964 nodes in 1980, to 76,708 nodes in 2015. At the same time, we counted the active members of the network. The results show that the proportion of active nodes changed from less than 20% in the first part of the 1980s, to nearly 30% in 2013.

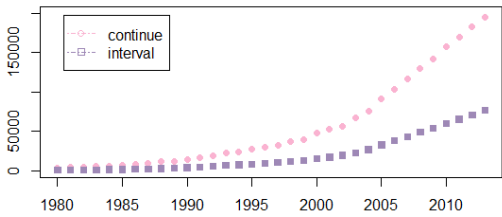


Figure 4. Network volume growth

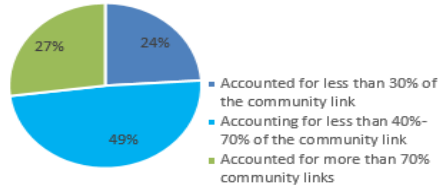


Figure 5. Active point percentage proportion community links

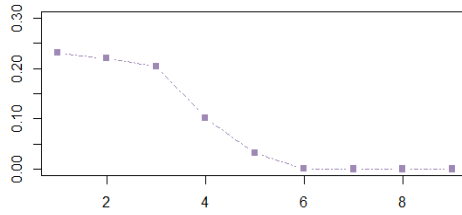
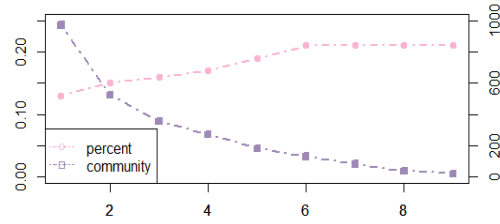
Figure 5 shows the proportion of active nodes in the total number of links in the community. It is obvious that, although the total number of active points is rather small, the links among the active nodes in the community account for a large proportion of the whole links. In most cases, links to active nodes exceed 60% of the community's internal connections. The experiment verifies the hypothesis that the active nodes occupy the dominant position in the community. Firstly, we verify the information similarity of individual nodes in the network by adopting the method of extracting keywords from multiple article titles. In the trial, the authors who publish only one article are excluded, and the authors of the top 10% of articles published between 2014 and 2013 are relative with about 9 other authors per year. The key words are sorted out from the articles and we use the Jaccard coefficient to calculate the information similarity. Meantime, the information similarity of any two nodes are arranged in contrast with previous similarities. The results are shown in Table 2. It shows that the similarity of the articles published by the same author in two consecutive years is 0.24, which is more than that of any other author by about 4%. Analyzing experimental results, we reach a conclusion that the information of the same node changes little in adjacent time, whereas the diversity of the information of different nodes is larger in the same time slices.

Table 2. Active point information similarity

| | Similarity | Diversity |
|------------------------------------|------------|-----------|
| The similarity of node information | 0.24 | 0.04 |

5.2. Threshold Settings

The evolution of the community can gradually change. We utilize the threshold to exclude the excessively drastic changes between the two successive snapshots in tracking communities. As shown in Figure 6, the information similarity stays steady when the ratio l , which is the volume threshold of two of the numbers of nodes, is less than 3 in the evolutionary community at successive snapshots. After the ratio exceed 3 times, it changes sharply. Moreover, when the ratio is higher than 6 times, the similarity ratio flattens and stays at a very low value. In view of Table 2, the similarity of the calculated node information is 0.24, the volume threshold l is set as 3, and more similar communities are tracked. We set the threshold k value from 0.1 to 0.9 for 9 spans, as shown in Figure 7. The experimental results show that when the threshold k increases, the number of communities tracked decreases. While the threshold varies from 0.1 to 0.3, the number of community matched decreases rapidly. If the threshold is over 0.3, the number decreases slowly. The similarity of communities is gradually increased from 0.1 to 0.6. If the threshold k is over 0.6, the similarity is basically stable at 0.21. The similarity of communities of successive snapshots is 0.24 (see Table 2). We think that the appropriate threshold of the algorithm should be between 0.5 and 0.6.

Figure 6. Volume threshold l Figure 7. Similarity threshold k

5.3. Comparison of Algorithm

In order to evaluate the efficiency of our proposed method and the availability of community tracking, the Community Tracking Method is compared with two other classical algorithms. The datasets used in the comparison are the network data sets of the <article> node in the DBLP datasets from 2010 to 2011. The similarity between communities is obtained by different algorithms. Meanwhile, we take a brief statistical result where the different numbers of similar communities are given by using different tracking methods. The experimental results are shown in Table 3.

Table 3. Comparison of Algorithm

| Algorithm | The number of communities | Similarity of information |
|-----------------------------|---------------------------|---------------------------|
| GP [6] | 948 | 0.227 |
| MODEC [12] | 1338 | 0.221 |
| The algorithm of this paper | 3515 | 0.218 |

According to the result data, the number of communities that can be obtained by means of our proposal in this paper is nearly four times than that of the GP [6] algorithm, which is nearly three times in contrast with the MODEC algorithm. At the same time, the algorithm of this paper is the superset of the GP algorithm and MODEC algorithm. This displays that the algorithm in this paper does not omit the results of other algorithms in community tracking and evolutionary research. However, in the similarity of community information, this algorithm is slightly lower than the other two algorithms, but the deviation is not obvious. The experimental results show that the proposed algorithm has a better result than the other two community tracking algorithms.

5.4. Evaluations

In this paper, we use the proposed method to analyze the DBLP datasets and selected the data from 1980 to 2014 for a total of 35 years. We selected a relatively large number of <article> nodes and <inproceedings> nodes for Community Tracking to count statistics of the evolution of the community. When we tracked the communities, we found that a large number of communities appeared only in a given year and are not involved in evolution. These communities are referred to as temporary communities and are classified separately. The dynamic evolution of temporary communities that only appear one time is illustrated in Figure 8 and Figure 9. It can be seen that the number of temporary communities and the growth trend in the two types of networks are higher than that of non-temporary communities.

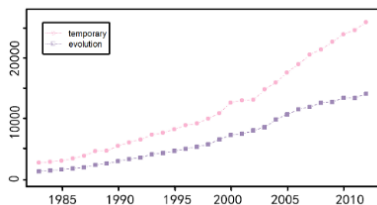


Figure 8. Article node

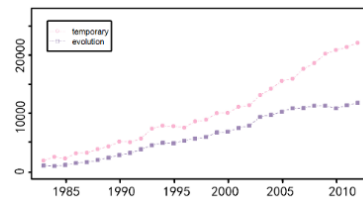


Figure 9. Inproceedings node

Focusing on non-temporary communities, the distribution of different evolutionary types changes each year. We pay attention to the evolution patterns of these six basic types. The experimental results are shown in Figure 10 and Figure 11. The distribution of each type of network is dynamic. Although there are many forms of evolution in the community, in the evolution process, most community life is relatively short. In other words, as large the community size is, most don't participate in community evolution. They only appear in a little short period while a great deal of communities showed a growth trend. When a community appears continuously in a network, it will be more attractive to other nodes. So, the

community allows other individual members to join it partly because of merging or growth structures. The number of contracting, splitting and merging in the community is relatively low, indicating that there is little interaction. In the co-author community, although different research institutions willingly work together, it is difficult to merge them into a group because they are affiliated to different institutions.

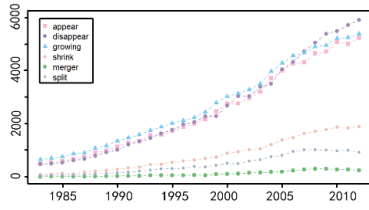


Figure 10. Article node

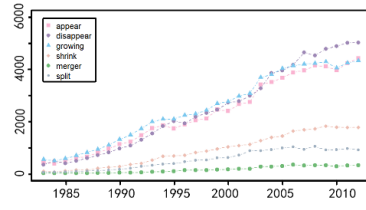


Figure 11. Inproceedings node

In section 3, we propose a concept about Active Node Contribution Factor (ANFC). To some extent, it shows that several active nodes play a significant role in the evolution of communities. During the active node lifetimes, they may take part in different evolutions. As in Figure 12 and Figure 13, we select two different group types. The one network is an organized Article node whose link attributes are about connections with common authors. The other network consists of inproceedings nodes. With an increasing number of nodes, the ANFC value also increases. We consider that the number of nodes partly reflects the type of community evolution. If the community behaves a growth tendency, the number of nodes corresponding to the community gradually increases. On the contrary, if some communities start to merge into a relatively large community, the number of new communities also rises.

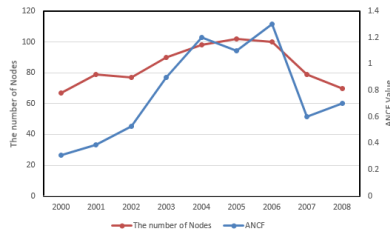


Figure 12. Article node

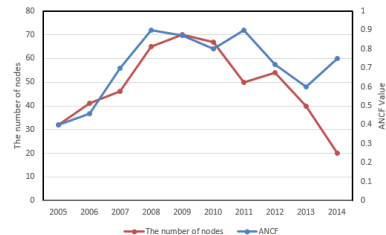


Figure 13. Inproceedings node

6. Conclusions

In this paper, we have proposed a community tracking algorithm based on time domain information and the active nodes of the network. The main advantage of our method is its ability to solve problems by distinguishing nodes in the process of tracking the communities and the time domain considerations in the matching function. We then applied our framework on the DBLP co-authorship datasets. Existing community tracking algorithms do not distinguish the nodes in the tracking process, and the matching function lacks the time domain. Through testing the DBLP datasets, our algorithm is compared with that of other papers [1,11]. Extensive experimental analysis demonstrated that the proposed method is very effective in tracking the community structure. At the same time, the performance of our framework on the datasets is also compared with other algorithm frameworks. The results show that our algorithm outperforms the others in terms of community information similarity. Hence, we intend to analyze the evolutionary process of the community and use the active part to track the non-adjacent time community to study the stable structure of the community.

Acknowledgements

This work is sponsored by the National Natural Science Foundation. In particular, we are grateful to Professor Yan and the team for their support in data gathering, data analysis, and direction. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the National natural Science Foundation. Funding was from the Research on the Equivalent Relations of Complex Network Models and Construction of High Quality Networks (grant number 61662066), the Research of the Function and Topological Structures of the Internet of Things based on Graph Theory (grant number 61363060), and the Science Project of Gansu Electric Power Information & Communication Centre (grant number SGTYHT /15- JS -194).

References

1. B. Albert-Laszlo, P. Gergely, T. Vicsek, "Quantifying Social Group Evolution," *Nature*, vol. 446, no. 7136, pp. 664-671, May 2007
2. R. Albert, A. L. Barabasi, "Emergence of Scaling in Random Networks," *Science*, vol. 286, no. 5439, pp. 509-512, October 1999
3. Z. Anna, G. Bogdan, K. Jarosław, B. Piotr, K. Przemysław, S. Stanisław, "Predicting Community Evolution in Social Networks," *Entropy*, vol. 17, no. 5, pp. 924-925, May 2015
4. S. Aral, D. Walker, "Identifying Influential and Susceptible Members of Social Networks," *Science*, vol. 337, no. 6092, pp. 337-41, July 2012
5. S. Asur, S. Parthasarathy, D. Ucar, "An Event-based Framework for Characterizing the Evolutionary Behavior of Interaction Graphs," *Acm Transactions on Knowledge Discovery from Data*, vol. 3, no. 4, pp. 913-921, August 2009
6. M. Bajec, L. Šubelj, "Robust Network Community Detection using Balanced Propagation," *European Physical Journal B*, vol. 81, no. 3, pp. 353-362, June 2011
7. C. Cattuto, L. Gauvin, A. Panisson, "Detecting the Community Structure and Activity Patterns of Temporal Networks: a Non-negative Tensor Factorization Approach," *Plos One*, vol. 9, no. 1, pp.1-13, January 2014
8. V. Colizza, L. Ferreri, C. Poletto, E. Valdano, "Analytical Computation of the Epidemic Threshold on Temporal Networks," *Physical Review X*, vol. 18, no. 3, pp. 503-512, April 2015
9. P. Cunningham, D. Doyle, D. Greene, "Tracking the Evolution of Communities in Dynamic Social Networks," *International Conference on Advances in Social Networks Analysis and Mining*, pp. 176-183, August 2010
10. P. Erdos, A. Renyi, "On the Evolution of Random Graphs," *Magyar Tud Akad Mat Kutato Int Kozl*, vol. 5, pp.17-61, January 1960
11. J. Fagnan, F. Sangi, M. Takaffoli, O. R. Zaiane, "Tracking Changes in Dynamic Information Networks," *Computational Aspects of Social Networks (CASoN)*, 2011 International Conference on. IEEE, pp. 94-101, October 2011
12. S. Fortunato, "Community Detection in Graphs," *Physics Reports*, vol. 486, pp. 75-174, June 2010
13. M. Girvan, M. E. J. Newman, "Community Structure in Social and Biological Networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, no. 12, pp. 7821-7826, April 2002
14. M. Goldberg, M. Magdonismail, S. Nambirajan, J. Thompson, "Tracking and Predicting Evolution of Social Communities," *IEEE Third International Conference on Privacy*, vol. 780-783, October 2011
15. J. Hopcroft, O. Khan, B. Kulis, B. Selman, "Tracking Evolving Communities in Large Linked Networks," *Proceedings of the National Academy of Sciences*, vol. 1, pp. 5249-53, May 2004
16. M. Ley, "DBLP: Some Lessons Learned," *Proceedings of the vldb endowment*, vol. 2, no. 2, pp. 1493-1500, June 2009
17. G. Rauret, A. Rigol, M. Vidal, "Empirical Analysis of An Evolving Social Network," *Science*, vol. 311, no. 5757, pp. 88-90, January 2006
18. S. H. Strogatz, D. J. Watts, "Collective Dynamics of 'Small-world' Networks," *Nature*, vol. 393, no. 6684, pp. 440-442, March 1998

Guanghui Yan is a Professor at the School of Electronic and Information Engineering of Lanzhou Jiaotong University. His current research interests include complex network, data mining, and social network.

Qingqing Ma is a Master's student from the School of Electronic and Information Engineering, Lanzhou Jiaotong University. Her research interests include complex network, time domain network and link prediction.

Yafei Wang is a Master's student from the School of Electronic and Information Engineering, Lanzhou Jiaotong University. Her research interests include complex network and time domain network.

Yu Wu is a Master's student from the School of Electronic and Information Engineering, Lanzhou Jiaotong University. His research interests include complex network, tensor decomposition and parallel computing.

Dan Jin is an expert in electricity and works at the Gansu Electric Power Information & Communication Centre. His research interests include complex network and time domain networks.