Filomat 32:5 (2018), 1559–1570
https://doi.org/10.2298/FIL1805559C

Published by Faculty of Sciences and Mathematics,
University of Niš, Serbia
Available at: http://www.pmf.ni.ac.rs/filomat

Similarity-based Local Community Detection for Bipartite Networks

Dongming Chen[a], Wei Zhao[a], Dongqi Wang[a], Xinyu Huang[a]

[a] *Software College, Northeastern University, 110169, Shenyang, China*

Abstract. Local community detection aims to obtain the local communities to which target nodes belong, by employing only partial information of the network. As a commonly used network model, bipartite applies naturally when modeling relations between two different classes of objects. There are three problems to be solved in local community detection, such as initial core node selection, expansion approach and community boundary criteria. In this work, a similaritybased local community detection algorithm for bipartite networks (SLCDB) is proposed, and the algorithm can be used to detect local community structure by only using either type of nodes of a bipartite network. Experiments on real data prove that SLCDB algorithms output community structure can achieve a very high modularity which outperforms most existing local community detection methods for bipartite networks.

1. Introduction

There are many forms of complex network models, such as one-mode network, bipartite network and multimode network[1]. Bipartite network is commonly and naturally existed in real world, when modeling relations between two different classes of objects in real world, such as paper reference relationship, movie actor-movie relationship. The dichotomy characteristics of target networks can help researchers reveal more details and useful networking features than single model networks[2]. Bipartite networks also share some statistical properties as their single-mode form, which means when regarding the nodes of the two parts of the network as the same, the single model network and original bipartite network share the same degree distribution, clustering coefficient, at the same time, bipartite also maintains more information of the real network being modeled than the single model version. As an important network model which is widely used in reality, researchers have done a lot bipartite network related research, bipartite network models of real application had been built, such as scientists cooperation network, the audience and the music network, book lending network, P2P exchange network and so on. There are mainly two ways to curve the relationship of two different classes of objects, one is projection method, which projects the two parts of the bipartite network into a certain type of node to carry on further study, the other is non-projection method. There are different ways of projecting different nodes into the same category, such as the unweighted projection and weighted projection[3, 4], but existing experiments proved that projection

2010 *Mathematics Subject Classification*. Primary 05C85; Secondary 05C90, 05C99.
Keywords. Bipartite network; Local community detection; Similarity; Subgraph.
Received: 12 September 2017; Accepted: 02 Nov 2017
Communicated by Xinyu Huang
Research supported by Liaoning Natural Science Foundation under Grant No.20170540320 and Research project of Liaoning Department of Education under Grant No.L2015173.

Email addresses: chendm@mail.neu.edu.cn (Dongming Chen), vikrant.zhao@qq.com (Wei Zhao), wangdq@swc.neu.edu.cn (Dongqi Wang), neuhxy@163.com (Xinyu Huang)

operation usually causes loss of information[5–7]. Therefore, the use of non-projection modeling is more reasonable.

Existing researches focus more on modifying community detection methods which are not specially designed for bipartite network to solve community detection problem, such as clustering algorithm based on edge clustering coefficient[8], K_{ab}-Biclique division method[9], module optimization algorithm[10], etc. These existing methods ignore the difference between the nodes of the two parts of bipartite network, and they also dont make sufficient use of the characteristics of bipartite network. On the other hand, global information are usually needed by these algorithms, which means that these algorithms are difficult to be applied without full structure information of target network. Alzahrani et al.[11] summarized the recent studies in bipartite networks and obtained clusters by Infomap, which are meaningful than those found in Louvain. Beckett et al.[12] introduced two algorithms, LPAwb+ and DIRTLPAwb+, for maximizing weighted modularity in bipartite networks, providing a different and potentially insightful method for evaluating network partitions. There are also some researchers improved the community detection algorithms in bipartite network, for example, Fan et al. studied community partition merging principle in single type of nodes in bipartite networks and proposed an improved bipartite networks community detection method, which was based on Page Rank algorithm, information spreading probability model and combined with the modularity, and they also demonstrated the effectiveness of the proposed method[13]. However, for huge and dynamic bipartite networks, it is difficult to collect global information[14], so taking full advantage of local network structure can be a better choice.

This paper focuses on detecting local community structure of bipartite networks, and reducing dependence on global information turns into the main target of the proposing method.

The rest of the paper is structured as follows: Section 2 describes the key questions of local community discovery in bipartite network. Section 3 introduces the proposed algorithm and analyzes the complexity. Section 4 provides some experiments on the proposed algorithm in different iterative orders. We conclude in Section 5 with possible directions for future work.

2. Key Issues of Local Community Detection in Bipartite Network

In order to solve local community detection problem, there are three basic questions needed to be answered: (1) How to choose initial node, which is called initial core[15]? (2) How to expand member nodes starting from initial core[16]? (3) What criteria should be adopted to determine the boundary of a community[17–20]? This section discusses and analyzes these questions so as to propose a solution to local community discovery in bipartite network.

2.1. *Initial Core of Local Community*

Whether it is a onemode network or a bipartite network, we always concern with the local surrounding structure of some key nodes. According to the general definition of local community, namely in a network G, given an initial node V_0, under the circumstance of referencing local information, the algorithm will find a community that contains the node.

In the light of this definition, the final community started from the core should contain node V_0; thus some algorithms directly use V_0 as the initial core, and then expand from this core node to discover the community.

The second idea takes consideration on community center, not starting from V_0, but selecting a node nearby V_0 which may be the community center. Typically, one approach is to traverse V_0's neighbor nodes from the direction which consists with the degree ascending, until the algorithm finds the node with the largest degree(marked as V_{max}), and V_{max} is employed as its initial core; another approach calculates V_0's neighbor nodes and the central degrees of the neighbor's neighbor nodes, so as to find V_c which holds the highest central degrees and is simultaneously close to V_0, then V_c is regarded as the initial core of the local community.

The initial core selection and subsequent expansion approach are closely correlated. Selecting central node is a prerequisite of subsequent expansion and boundary determination in some local community discovery algorithms. Selecting central node is beneficial to improve the efficiency and stability of local communities mining, while the rationality of the selected initial core remains to be verified. Firstly, the central node nearest to V_0 and V_0 itself may not fall into the same community. Secondly, starting from the node with high clustering coefficient may lead to overlapping communities. Starting from the original node could avoid these problems, however, the subsequent expansion strategy and boundary determination become more difficult to deal with, and is prone to poor stability, that is to say, starting from different nodes in the same community may lead to different results.

The above approaches which start from a single node are defective. The proposed algorithm considers sub-graph, which is a group of nodes belonging to the same community, as the initial core of local communities. To a network, node similarity refers to the closeness between two nodes in certain aspects, such as structural similarity and regular similarity. High similarity between two nodes illustrates that these two nodes share more common features. Thus, the more similar the two nodes are, the more possible they are from the same community. Selected collections of nodes sharing the highest similarity and more common features are regarded as similar sub-graph.

As mentioned above, using a similar sub-graph as the initial core of a local community is reasonable, and a more reasonable choice is to select the most similar group of nodes which should be a complete graph or so called component, which can ensure the stability of initial core and avoid problems resulted from selecting inaccurate central node as the initial core.

2.2. Local Community Expansion Approach

Local community expansion approach is defined as the approach to choose neighbor nodes to be added into current community after obtaining the initial core[21]. Each time you select a node to be added, considering the following approaches: one by one node, level expansion and sub-graph expansion.

The most common expansion strategy is the one by one node approach. A neighbor node is chosen for each existed community. In light to certain criteria, the algorithm decides whether this certain node will be added into the current community or not, repeating this process until the algorithm satisfies community boundary conditions. The new node to be added in accordance with the standard closeness selection to the existed community, such as the external node which has highest connectivity with existed communities, the node which has the highest similarity with existed community boundary nodes and so on. The biggest problem of one by one node approach is that the final result is instable, that is to say, starting from different nodes in the same community may lead to different results. Relatively speaking, compared with starting from boundary nodes, starting from central nodes would obtain a better result. Given that the unknown network structure could cause uncertainty, one by one node approach is not desirable.

Subsequently, considering with level expansion approach to implement local community expansion, the definition of level expansion demonstrates that adding the neighbor nodes of the initial core and the neighbor nodes neighbors into community sequentially, and the algorithm repeats this process until the current node is judged to be a boundary node. Level expansion approach requires the initial core to be selected must be a central node, otherwise the performance will be degraded. Level expansion approach is more efficient than one by one node approach, and the result turns to be better when the network has a regular community structure. However, if the community structure of the network is irregular, level expansion approach would join or exclude all the nodes of a certain level, and it is difficult to ensure the accuracy of community boundaries.

Sub-graph expansion approach refers that the network is divided into several sub-graphs, each time you select a sub-graph and determine whether all nodes in the sub-graph should be included into existed communities. In the prerequisite of all sub-graph nodes belonging to the same community, this approach is more efficient and reasonable than level expansion approach. Meanwhile, aiming at irregular community structure, the result of sub-graph expansion is better than that of the level expansion approach.

The algorithm employs similar sub-graph approach to expand the local community. For a given initial node, we can calculate the closest node to obtain a similar sub-graph and regard the sub-graph as a core, then starting from the core(similar sub-graph) to obtain nearby similar sub-graph. According to the definition of similar sub-graph, the nodes of each similar sub-graph fall into the same community; thus, based on neighboring similar sub-graph to extend sub-graphs holds its rationality and the completeness of similar sub-graphs makes the judgment of boundaries for merging sub-graphs more accurate.

2.3. Local Community Evaluation Criteria

Another key problem lies in the local community discovery, namely how to determine whether a node of an existed local community should be expanded, which is also the criteria of the local community boundariesthe termination conditions of the algorithm. For global community partitioning algorithm, the accuracy of the results can be measured by modularity function or other evaluation indicators; but for local community discovery algorithm, the results can only be judged by the local information, thus, it is difficult to make judgments for highly coupled complex networks.

Some researchers supposed a number of evaluation indicators for local community of onemode network. Clauset et al.[17] proposed a local community discovery algorithm based on local modularity R, the R is defined as follows:

$$R = \frac{B_{in}}{B_{in} + B_{out}}, \tag{1}$$

where Bin represents the number of links between the boundary nodes and internal nodes in a community, Bout indicates the number of links between boundary nodes in the current community and other nodes out of the current community. R value represents steep degree of the community boundary, the larger the R value, the higher the proportion of links between community boundary nodes and inner nodes in the community, namely the more obvious of the community structure. Conversely, if adding a node leads to the R value reducing, representing its links is much closer with outer nodes of the community rather than internal nodes in the community, so the node should be excluded this community. Using R value to measure the community boundary obtains its rationality as well as lower accuracy. Moreover, this algorithm uses node by node expansion approach and needs a prespecified community scale as end condition; apparently it is unable to meet the actual needs.

Bagrow et al. [19] proposed an algorithm of extensions of the local communities in accordance with the level of mining, this paper defined an indicator, and a presetting threshold value α. When the indicator is less than α, it is considered that the algorithm reached community boundaries. Defined as:

$$\Delta K_j^l = \frac{K_j^l}{K_j^{l-1}}. \tag{2}$$

It represents a ratio between the number of external connection links of the l layer which starting from the node j, and the number of external connection links of $l-1$ layer starting from the same node j. The algorithm uses layer by layer expansion approach and the results is not very accurate; meanwhile the presetting threshold value α is subjective, it is not very reasonable.

Luo et al.[18] uses M (the ratio between inner links and outer links) as the indicator of local modularity, repeating the process by deciding each neighbor node to join the current community or not until M reaches a maximum value which is regarded as the end conditions. Defined as follows:

$$M = \frac{E_{in}}{E_{out}}, \tag{3}$$

where E_{in} represents the total number of edges inside the community; E_{out} provides the number of links between boundary nodes and outside nodes of the community. The algorithm has no default end conditions and it judges the end conditions by the extreme value of M, which is obviously more reasonable; however, the stability problems of "one by one node" expansion approach still exists.

Chen et al. proposes L-a ratio [22] between inside and outside degrees which is used as a local community evaluation indicator. Defined as follows:

$$L = \frac{L_{in}}{L_{ex}}, \tag{4}$$

where L_{in} is a ratio between the number of edges and the number of nodes in the community, and L_{ex} is a ratio between the number of edges which connect to the outside community and the number of boundary nodes. The algorithm also has the similar problem which mentioned earlier.

In summary, due to the lack of global information, local community usually incorporates boundary information inside and outside the community as a basis for evaluation; however, in terms of the uncertainty of the expansion process, it is difficult to ensure the accuracy and stability of the results. Our algorithm proposes a solution to these problems.

2.3.1. *Edge Clustering Coefficient in Bipartite Network*

The edge clustering coefficient of a one-mode network is defined as a ration between the number of triangles of a certain edge and all possible triangles [23]. Zhang et al. proposed the definition of edge clustering coefficient in bipartite network as follows[24] :

$$LC_3(i,X) = \frac{1}{k_i + k_X - 2}\left(\sum_{m=2}^{K_x} \frac{t_{mi}}{k_m + k_i - t_{mi}} + \sum_{N=2}^{k_i} \frac{t_{NX}}{k_N + k_X - t_{NX}}\right), \tag{5}$$

where k_i and k_X represent the degrees of node i and node X respectively. m is the number of neighbor nodes of node X excluding node i, t_{mi} shows the triples between node m and node i (the number of common neighbors between node m and node i), node N illustrates the neighbor nodes of node i excluding node X, and the definition of t_{NX} is similar with the definition of t_{mi}.

2.3.2. *Modularity Q_B*

In order to evaluate whether the result of the algorithm is good or not, Newman and Girvan proposed a definition of modularity function[25], that is, Q function. Defined as follows:

$$Q = \frac{1}{2m}\sum\left[A_{ij} - \frac{k_i k_j}{2m}\right]\delta(c_i, c_j), \tag{6}$$

where A represents the adjacency matrix of the network, k_i denotes the degree of node i, m exhibits the total number of edges; c_i is the community which contains node i, c_j represents the community which contains node j. $\delta(c_i, c_j) = 1$satisfies only when c_i equals to c_j.

The Q function represents the close degree of inner edges in each network. The value of Q ranges from -0.5 to 1. In fact, the value of Q usually ranges from 0.3 to 0.7 in common network.

2.3.3. *Similarity Indicators in Bipartite Network*

Similarity measurement has been widely researched in one-mode network. One of the basic similarity measurement indicator is Common Neighbor indicator[26]. Taking the impact between nodes into consideration, similarity can be evaluated in many ways, such as Salton indicator (also known as cosine similarity)[27], Jaccard indicator[28], Srensen indicator[29], hub node promoted indicators (Hub Promoted Index, *HPI*)[30], hub depress indicators (Hub Depress Index, *HDI*)[31]and Leicht indicator[32]. In addition, addressing common neighbors information of two nodes, a set of similarity indicators were proposed, such as the famous Adamic-Adar(*AA*) indicator[33], the idea of Adamic-Adar index is that the greater the degree of common neighbor, the smaller the contribution to the node similarity, and vice versa. From the angle on network resource allocation, indexes, such as *RA* indicator[31] and the similarity indicator based on local Naive Bayes model[34], can also be a good way to evaluate the similarity.

Among various similarity indicators, indicator based on the random walk of similarity holds a good effect and a wide range of applications. A considerable amount of similarity indicators are based on a random walk process definition. For example, indicator based on the random walk of cosine similarity (*Cos*+) [35], the average commute time (Average Commute Time, *ACT*)[36], SimRank indicator[37], restarted the random walk indicator (Random Walk with Restart, *RWR*)[38], these are indicators of global random walk.

Because computational complexity of global random walk indicators is very high, it is difficult to apply this indicator to large-scale networks. To solve this problem, Liu Weiping and Lv Linyuan proposed a random walk similarity indicator with a number of steps in a limited time (Local Random Walk, *LRW*)[39], and proved that the performance of the proposed indicator is much better than common neighbor based approaches and the computational complexity is much smaller than the global random walking based methods, such as Superposed Random Walk, *SRW* based method. The *SRW* indicator considers the probability of the most likely connection between the target node and the adjacent node, and the efficiency and accuracy are quite high. *SRW* similarity is calculated as follows:

$$S_{xy}^{SRW} = \sum_{l=1}^{t} S_{xy}^{SRW} = q_x \sum_{l=1}^{t} \pi_{xy}(l) + q_y \sum_{l=1}^{t} \pi_{yx}(l), \tag{7}$$

where $S_{xy}^{SRW}(t)$ represents random walk similarity of node x and node y after it is superimposed to t step, q_x and q_y are parameters which are initially allocated to node x and node y, $\pi_{xy}(l)$ is the probability starting from node x and arriving node y through l steps.

For the local community detection problems in bipartite network, given the initial node x_0, we use *SimRank* to gain the most similar node to x_0, which marked as x_{sim}. *SimRank* indicator is a commonly used similarity evaluation indicator which is based on a random walk of global information, to solve local community detection problem, we can adopt random walk model based on finite local information.

Our algorithm extended superimposed random walk indicator(*SRW*) to apply to bipartite network. Starting from the initial node with limited depth traverse, it is possible to obtain the corresponding probability of achieving neighboring nodes. The probability of the same kind of nodes is regarded as the comparative probability, the node with largest probability is the most similar node with the original node. Then the process will be repeated to achieve similar node sub-graphs in the original bipartite network, and these sub-graphs are employed as the initial core and merging units for the local communities mining process.

2.3.4. Local Community Modularity in Bipartite Network

For bipartite networks, the evaluation of a local community can only be analyzed by the structural information of the certain community. Only the statistical degree of edge information leads to low accuracy; moreover, betweenness is a global feature for a network, and edge clustering coefficient becomes a more reasonable choice.

Based on triple edge clustering coefficient LC_3 in bipartite network, it characterized the connection tightness of the end nodes. According to the definition of community, clustering coefficient among inner edges of the community is high and low among boundary edges. Figure 1 illustrates the LC_3 clustering coefficient of each edge for a simple bipartite network, it can be seen that, in the figure the network was divided into two parts(left and right) and its internal edge clustering coefficients(5/6, 23/36) were significantly higher than that of the boundary edges (2, C) whose clustering coefficient was 1/4. The figure discloses that it was reasonable to employ edge clustering coefficient as an indicator to distinguish local community boundaries.

Therefore, our algorithm utilizes the accumulated ratio between the external edge clustering coefficient of boundary nodes and the total edge clustering coefficient as the modularity of local community M_B, which is formulated as:

$$M_B(C) = \sum_{a \in C} \frac{\sum_{b \notin C} LC_3(a, b)}{\sum_{i=1}^{n} LC_3(a, i)}, \tag{8}$$

where a is the boundary nodes of the community, b is the neighbor nodes of the community, and n is the number of neighbor nodes of a. Since the edge clustering coefficient uses the structural information of near range, modularity M_B represents the connection tightness between the boundary nodes and the outside nodes of the community. The smaller the M_B, the more sparse links between the community and external, so that the more obvious the community structure presents. Thus, this formulated modularity can be used as the criteria for the sub-graph merging process.

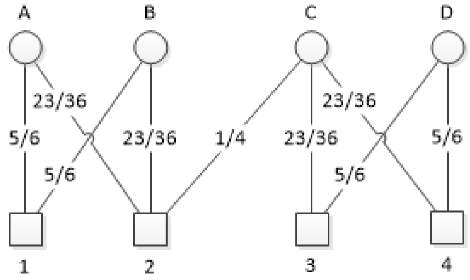

Figure 1: Sample of edge clustering coefficient LC_3

3. Similarity-based Bipartite Local Community Detection Algorithm

3.1. Algorithm Description

SLCDB algorithm is denoted as:

$$BLC = SLCDB(B, v_0), \tag{9}$$

where the input parameters are the bipartite network B and the initial node v_0, the output result is a local community BLC which contains v_0.

Step 1: Based on the conception of similarity sub-graph, the algorithm obtained the initial core which contains v_0.

a) Regarding V_0 as the current node, and the initial core set is marked as: $C_0 = \{v_0\}$;

b) Starting from the current node, calling the random walk function, the algorithm obtains the similarity indicators of the neighbor nodes;

c) Selecting the highest similarity node s of v_0, if node s is in the set C_0, the algorithm turns to Step 2, otherwise continues with Step 1-d);

d) Adding s to C_0, selecting s as the current node then algorithm repeated the Step 1-b).

Step 2: Acquire and expand neighbor similarity sub-graph.

a) Obtaining a neighbor node n which is not being traversed in C_0, the algorithm turns to Step 2-b). If all the nodes in C_0 are traversed, the algorithm turns to Step 2-d);

b) In light of the process of Step 1, starting from node n, obtaining the similarity sub-graph C_n which contains node n, then the algorithm marks the most similar node i while merging it into C_n as traversed; if node i belongs to certain sub-graph C_i (C_i may be the initial core C_0 or another neighbor sub-graph), the algorithm merge C_n into C_i. The algorithm returns to Step 2-a), otherwise continues with Step 2-c);

c) The algorithm obtains new sub-graph C_n and stores it into $ListC$, then returns to Step 2-a);

d) After finishing obtaining all the similar sub-graphs, the algorithm would expand the neighbor sub-graph, namely traversing all the nodes in the neighbor sub-graph which is not traversed; if the most similar node belongs to a certain existed sub-graph, then merge the node into the sub-graph, the algorithm repeated the process until all the boundary nodes dont belong to any neighbor sub-graph.

Step 3: Merge and expand the sub-graph.

a) Calculating the modularity M_B for all the sub-graphs;

b) Selecting a neighbor sub-graph from $ListC$ in turns, calculating the increased modularity ΔM_B with other sub-graph (including core sub-graph C_0). If ΔM_B is less than 0, the algorithm will not merge this sub-graph and repeat current process, otherwise the algorithm will continue to execute Step 3-c);

c) Merging sub-graph, selecting sub-graph with the biggest ΔM_B to merge; if two neighbor sub-graphs are to be merged, the algorithm would trace back to Step 3-b), then selecting next sub-graph; if the core sub-graph C_0 merges with neighbor sub-graph, due to C_0 expanded the neighbor nodes which are not traversed, the algorithm need to trace back to Step 2 to obtain new neighbor sub-graph.

Step 4: After finishing the merge of sub-graphs, the algorithm would expand C_0 into two local communities BLC which contains two types of nodes(e.g. if the bipartite network is company-shareholder network, then the two types of nodes are company nodes and shareholders nodes), and the algorithm finishes and the results are returned.

3.2. *The Complexity of Algorithm*

3.2.1. *Time Complexity*

The SLCDB algorithm is divided into 4 steps. The main computation of the first two steps is to calculate the similarity indicator based on random walk. If starting from a node, then the algorithm executes a random walk within w steps; thus, the complexity is the power of maximum node degree w. However, we only need to obtain maximum similarity of the same type nodes in practice. Therefore, walk path probability is less than the threshold value α and the algorithm stops before the next iteration. For example, the network's average degree $k = 10$, if the threshold of $\alpha = 1/1000$, then the actual average depth of the iteration is $log_k(1/\alpha) + 1 = 4$. Thus, if the core sub-graph and neighbor sub-graphs contain the average of total nodes which is $(t + 1)m$, then the total similarity indicator complexity is $O(ktm/\alpha)$.

The expansion process of the subgraph needs to continue to traverse and add the neighbor nodes, the maximum complexity does not exceed the size of the community m, and all the neighbor sub-graphs expansion is proportional to the maximum complexity of t_m. The merging process of sub-graphs requires to compute the value local modularity M_B, if the size of sub-graph is m, the number of average total edges is km, and the number of community internal edge links is bigger than external edge links; thus, the number of edges which needs to be calculated the clustering coefficient of LC_3 is less than $km/2$; and calculating LC_3 of each edge needs to traverse both of the end vertices neighbor nodes, its complexity is $2k$. Therefore, the local modularity computation complexity of each sub-graph is $O(k^2m)$.

In order to judge the sub-graphs merging process, the algorithm will calculate gain and loss of local modularity after subgraph merging operations. If the average number of sub-graph is $t+1$, then the number of possible combinations is $t \times (t + 1)$, and the computation complexity is $O(k^2t^2m)$. After subgraph merging operation, the algorithm recalculates local modularity gain and loss, assuming that the scale of community is $log(m)$, then, the computation complexity of iterative merge function will be $O(k^2t^2mlog(m))$.

Base on above analysis, the maximum complexity of the algorithm is $O(k^2t^2mlog(m))$, and which is the time computation complexity of SLCDB algorithm.

3.2.2. *Space Complexity*

The data structure of the whole network uses adjacency table. Storage of all the nodes in the base table occupies space n. The length of each node of the linked list is unequal which depends on the number of its neighbor nodes, and the average value is k, so the space complexity is $O(n + kn)$.

In the partition process, the algorithm needs to keep information, degree, similarity indicator etc. of all involved nodes, which may occupy the space $O(ktm/\alpha)$. The algorithm also needs to build a collection of lists used to store the core sub graph and all neighbor graphs and it takes up the space $O(m + tm)$.

Because the size of a community and all its neighbors are not larger than the size of the entire network n, the maximum space complexity of SLCDB algorithm is $O(kn)$, which is less than $O(n^2)$.

4. Experiment and Results Analysis

In order to evaluate the performance of the proposed algorithm, Scotland corporation network[40] dataset is employed in the experiment. This dataset shows the relationship between 136 shareholders of 108 joint-stock companies in early 20th century in Scotland. Each company has a plurality of shareholders and vice versa. The relationship between a company and a shareholder shapes a classical bipartite network. The structure of this network is shown in Figure 2, the nodes in red colour represent companies and the nodes in cyan colour represent shareholders. This algorithm uses the maximal connected subgraph which contains 86 companies and 131 shareholders, including 217 nodes and 348 edges.

To verify the performance, supposing a node set of bipartite network, marked as *setB*, the experiment is carried out in two different traversing orders, node degree ascending and descending. Execute SLCDB algorithm and get an output community result denoted by *listC*, then remove the nodes in *listC* from *setB*.

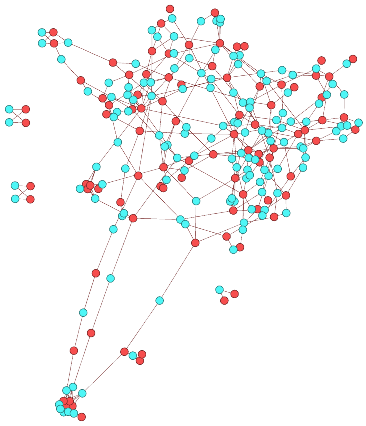

Figure 2: Scotland corporation network

Repeat the above step until every node falls into a certain community. Firstly, the algorithm will take different nodes as starting nodes to execute to verify the stability of the algorithm; secondly, it will take different type of nodes as starting nodes to execute so as to compare the differences and similarities of the two results; finally, it will be verified by Q_B to prove accuracy of the proposed algorithm.

4.1. *Community Results Based on Company Nodes Ascending*

Nodes 1~108 represent companies, each time starting from the company nodes with minimum degree. The local community for each step is shown in Table 1. It shows the algorithm detects 17 local communities by starting from company nodes. There is no overlapping between company nodes; however, some shareholder nodes are overlapped due to such type of nodes have many connected edges belonging to many communities.

4.2. *Community Detection Results Based on Company Nodes Descending*

The experimental result of starting from the company nodes with maximum degree is shown in Table 2. The results are similar as experiments on the company nodes ascending. Algorithms would get different community results when using different nodes as the initial nodes as was proved in the 2nd paragraph of section 2.2, and Label Propagation Algorithm (LPA)[40] is a typical example. Starting from either descending order or ascending order of node degree, the two community detecting results remain very good, and this phenomenon proves the stability of the proposed algorithm.

4.3. *Community Detection Results Based on Shareholder Nodes Ascending and Descending*

Each time starting from the shareholder nodes with minimum degree, the local community for each step is exhibited in Table 3, which shows 22 communities are divided by the algorithm. The results are different from above results, demonstrating that the proposed algorithm is effective on the same type of nodes in spite of descending or ascending. The structure of communities of company nodes is quite different from the structure of communities of shareholder nodes.

Each time starting from the maximum node of shareholder nodes, the algorithm gets 22 communities. The results keep the same with Table 3, which proves the stability of the algorithm is acceptable.

4.4. *Community Detection Results Analysis*

Figure 3(a) shows the community results based on company nodes and Figure 3(b) discloses the community results based on shareholder nodes. The nodes in dark color represent shareholders and the nodes in shallow color indicate company. Both of the two results conform to the community structure. The two community detection results are not the same. Figure 3(a) represents the communities composed by the

Table 1: Partition result of Scotland corporation network by SLCDB (company nodes ascending)

Group	Node	Local community result
1	1	[1, 3, 7, 12, 16, 21, 23, 50, 53, 109, 110, 112, 113, 114, 115, 116, 130, 131, 132, 133, 134, 145, 177]
2	2	[2, 20, 27, 29, 35, 46, 58, 62, 71, 117, 118, 120, 121, 122, 123, 125, 126, 127, 128, 129, 130, 153, 176, 181]
...	...	...
17	96	[96, 99, 105, 221, 222, 228, 229, 238, 241, 242, 243]

companies and their owners. Figure 3(b) reveals the communities composed by shareholders and their common companies. In Figure 3(a), a node in each community could be either a company node or a common shareholder node, the results emphasize the relationship between companies. In Figure 3(b), a node in each community could be either a shareholder node or a company node, the results emphasize the relationship between the shareholders.

Q_B is an indicator to judge whether the algorithm is good or not and it was mentioned in Section 2. LPA is a very famous algorithm due to its time efficiency and widely used in the actual application, the main idea of this algorithm described as follows:

Each node is initialized with a unique label, and during the iteration, each nodes label is changed into the maximum number of its neighbors. As the labels propagate through the network in this manner, densely connected groups of nodes form a consensus on their labels. At the end of the algorithm, nodes with the same label are grouped together as communities. The advantage of this algorithm over the other methods is its simplicity and time efficiency. The time complexity of LPA is $O(n)$.

The comparison of partition modularity is shown in Figure 4. The results prove that SLCDB algorithm gets higher accuracy than LPA. According to the discussion in section 3.2, the time complexity of SLCDB is higher than that of LPA, and this is what were going to do next.

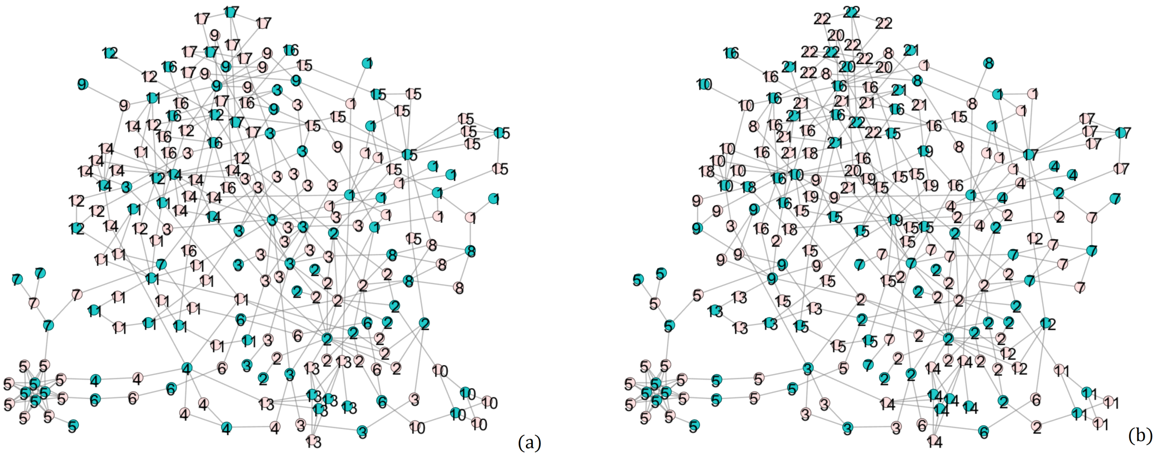

Figure 3: Partition result of Scotland corporation network by SLCDB

5. Conclusion

A similarity based local community detection algorithm for bipartite network is proposed. The algorithm employs random walk strategy as similarity indicator and starts from similar sub-graphs to expand community structure, the modularity of bipartite network is used as the community detection criteria. The accuracy and stability of algorithm was proved by experiment on real data. This research gives a practical

Table 2: Partition result of Scotland corporation network by SLCDB (company nodes descending)

Group	Node	Local community result
1	108	[55, 59, 67, 77, 82, 87, 100, 108, 121, 167, 168, 178, 179, 182, 191, 192, 197, 198, 199, 200, 206, 226, 227, 230, 231, 244]
2	106	[89, 91, 98, 106, 225, 227, 231, 234, 235, 236, 238, 242]
...	...	...
17	39	[38, 39, 172, 173, 174, 175]

Table 3: Partition result of Scotland corporation network by SLCDB (shareholder nodes ascending)

Group	Node	Local community result
1	109	[1, 7, 23, 50, 109, 112, 145]
2	110	[28, 59, 61, 67, 110, 168, 182, 183, 184, 192, 197]
...	...	...
22	225	[89, 90, 91, 98, 106, 225, 231, 234, 235, 236, 237, 238]

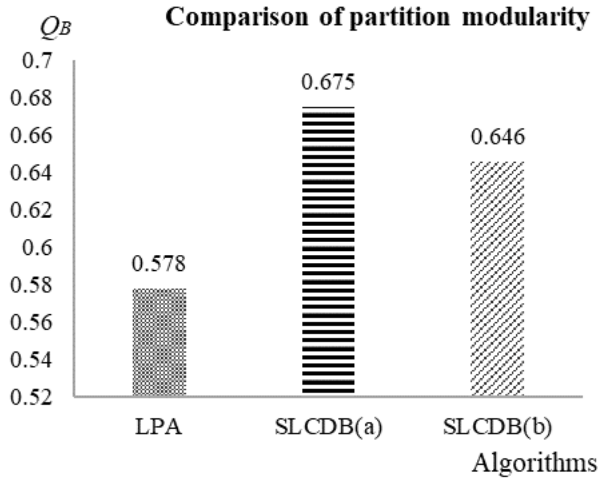

Figure 4: Comparison of partition modularity

way of community detection for bipartite network by using only partial network information. The research on the three key issues (initial core node selection, expansion approach and community boundary criteria) of community detection can be a useful reference for relative researches. The next step is to optimize the time efficiency of this algorithm.

6. Acknowledgement

This work was partially supported by Liaoning Natural Science Foundation under Grant No.20170540320 and Research project of Liaoning Department of Education under Grant No.L2015173.

References

[1] Costa L. Structure and Function in Complex Networks[J]. 2016.

[2] Wang H, Hu J. Modeling and Analysis of Baoji Bus Line-station Network Based on the Bipartite Network[J]. Electrical Automation. 2016.

[3] Qiao J, Meng YY, Chen H, Huang HQ, Li GY. Modeling one-mode projection of bipartite networks by tagging vertex information[J]. Physica A Statistical Mechanics & Its Applications. 2016;457:270-9.

[4] Gao M, Chen L, Yong-Cheng XU. Projection Based Algorithm for Link Prediction in Bipartite Network[J]. Computer Science. 2016.

[5] Kitsak M, Papadopoulos F, Krioukov D. Latent geometry of bipartite networks[J]. Physical Review E. 2017;95.

[6] Wang GX, Liu H, Li Q. Bipartite network projection and its application in recommendation systems2013. 5052-7 p.

[7] Zhou T, Ren J, Medo M, Zhang YC. Bipartite network projection and personal recommendation. Phys Rev E 76(4):046115[J]. Physical Review E. 2007;76(4 Pt 2):046115.

[8] Zhang P, Wang J, Li X, Li M, Di Z, Fan Y. Clustering coefficient and community structure of bipartite networks[J]. Physica A Statistical Mechanics & Its Applications. 2008;387(27):6869-75.

[9] Lehmann S, Schwartz M, Hansen LK. Biclique communities[J]. Physical Review E Statistical Nonlinear & Soft Matter Physics. 2008;78(2):016108.

[10] Guimer R, Sales-Pardo M, Amaral LA. Module identification in bipartite and directed networks[J]. Physical Review E Statistical Nonlinear & Soft Matter Physics. 2007;76(3 Pt 2):036102.

[11] Alzahrani T, Horadam KJ. Community Detection in Bipartite Networks: Algorithms and Case studies[J]. Understanding Complex Systems. 2016;73:25-50.

[12] Beckett SJ. Improved community detection in weighted bipartite networks[J]. Royal Society Open Science. 2016;3(1):140536.

[13] Fan C, Song Y, Song H, Ding G, editors. An Improved Community Detection Method in Bipartite Networks. International Conference on Web-Age Information Management; 2016.

[14] Bu Z, Wu Z, Cao J, Jiang Y. Local Community Mining on Distributed and Dynamic Networks From a Multiagent Perspective[J]. IEEE Transactions on Cybernetics. 2016;46(4):986-99.

[15] Interdonato R, Tagarelli A, Ienco D, Sallaberry A, Poncelet P, editors. Local community detection in multilayer networks. Ieee/acm International Conference on Advances in Social Networks Analysis and Mining; 2016.

[16] Imperiale AJ, Vanclay F. Experiencing local community resilience in action: Learning from post-disaster communities[J]. Journal of Rural Studies. 2016;47(Part A):204-19.

[17] Clauset A. Finding local community structure in networks[J]. Physical Review E Statistical Nonlinear & Soft Matter Physics. 2005;72(2):026132.

[18] Luo F, Wang JZ, Promislow E. Exploring local community structures in large networks[J]. Web Intelligence & Agent Systems An International Journal. 2008;6(4):387-400.

[19] Bagrow JP, Bollt EM. Local method for detecting communities[J]. Physical Review E Statistical Nonlinear & Soft Matter Physics. 2005;72(2):046108.

[20] Barber MJ. Modularity and community detection in bipartite networks[J]. Phys Rev E Stat Nonlin Soft Matter Phys. 2007;76(6 Pt 2):066102.

[21] Whang J, Gleich D, Dhillon I. Overlapping Community Detection Using Neighborhood-Inflated Seed Expansion[J]. IEEE Transactions on Knowledge & Data Engineering. 2015;28(5):1272-84.

[22] Chen J, Goebel R, editors. Local Community Identification in Social Networks. International Conference on Advances in Social Network Analysis and Mining; 2009.

[23] Radicchi F, Castellano C, Cecconi F, Loreto V, Parisi D. Defining and identifying communities in networks[J]. Proceedings of the National Academy of Sciences of the United States of America. 2004;101(9):2658.

[24] Zhang P, Li M, Mendes JFF, Di Z, Fan Y. Empirical Analysis and Evolving Model of Bipartite Networks[J]. Physics - Physics and Society. 2008.

[25] Girvan M, Newman ME. Community structure in social and biological networks[J]. Proceedings of the national academy of sciences. 2002;99(12):7821-6.

[26] Franois Lorrain, Harrison C. White. Structural equivalence of individuals in social networks[J]. Social Networks, 1977, 1(1):67-98.

[27] Salton G MGM. Introduction to Modern Information Retrieval[J]. 1983.

[28] Jaccard P. Etude de la distribution florale dans une portion des Alpes et du Jura[J]. Bulletin De La Societe Vaudoise Des Sciences Naturelles. 1901;37(142):547-79.

[29] Srensen T. A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on Danish commons[J]. Biol Skr. 1957;5:1-34.

[30] Ravasz E, Somera AL, Mongru DA, Oltvai ZN, Barabsi AL. Hierarchical Organization of Modularity in Metabolic Networks[J]. Science. 2002;297(5586):1551.

[31] Zhou T, L L, Zhang YC. Predicting missing links via local information[J]. European Physical Journal B. 2009;71(4):623-30.

[32] Leicht EA, Holme P, Newman ME. Vertex similarity in networks[J]. Physical Review E Statistical Nonlinear & Soft Matter Physics. 2006;73(2 Pt 2):026120.

[33] Adamic LA, Adar E. Friends and neighbors on the Web[J]. Social Networks. 2003;25(3):211-30.

[34] Liu Z, Zhang QM, L L, Zhou T. Link prediction in complex networks: a local naïve Bayes model[J]. Epl. 2011;96(4):48007.

[35] Fouss F, Pirotte A, Renders JM, Saerens M. Random-Walk Computation of Similarities between Nodes of a Graph with Application to Collaborative Recommendation[J]. IEEE Transactions on Knowledge & Data Engineering. 2007;19(3):355-69.

[36] Klein DJ, Randi M. Resistance distance[J]. Journal of Mathematical Chemistry. 1993;12(1):81-95.

[37] Jeh G, Widom J, editors. SimRank: a measure of structural-context similarity. Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining; 2002.

[38] Brin S, Page L. The anatomy of a large-scale hypertextual Web search engine[J]. Computer Networks & Isdn Systems. 2012;56(18):3825-33.

[39] Raghavan UN, Albert R, Kumara S. Near linear time algorithm to detect community structures in large-scale networks[J]. Physical Review E Statistical Nonlinear & Soft Matter Physics. 2007;76(3 Pt 2):036106.

[40] Yongcheng XU, Chen L. Community Detection on Bipartite Networks Based on Ant Colony Optimization[J]. Journal of Frontiers of Computer Science & Technology. 2014.