Article

Detecting the Structural Hole for Social Communities Based on Conductance–Degree

Zhifang Liao [1], **Lite Gu** [1], **Xiaoping Fan** [2],*, **Yan Zhang** [3] **and Chuanqi Tang** [1]

[1] School of Computer Science and Engineering, Central South University, Changsha 410083, China; zfliao@csu.edu.cn (Z.L.); gulite@csu.edu.cn (L.G.); 184712010@csu.edu.cn (C.T.)

[2] Information Management Department, Hunan University of Finance and Economics, Changsha 410205, China

[3] Department of Computer, School of Engineering and Built Environment, Glasgow Caledonian University, Glasgow G4 0BA, UK; yan.zhang@gcu.ac.uk

* Correspondence: xpfan@csu.edu.cn; Tel.: +86-0731-88811267

check for updates

Abstract: It has been shown that identifying the structural holes in social networks may help people analyze complex networks, which is crucial in community detection, diffusion control, viral marketing, and academic activities. Structural holes bridge different communities and gain access to multiple sources of information flow. In this paper, we devised a structural hole detection algorithm, known as the Conductance–Degree structural hole detection algorithm (CD-SHA), which computes the conductance and degree score of a vertex to identify the structural hole spanners in social networks. Next, we proposed an improved label propagation algorithm based on conductance (C-LPA) to filter the jamming nodes, which have a high conductance and degree score but are not structural holes. Finally, we evaluated the performance of the algorithm on different real-world networks, and we calculated several metrics for both structural holes and communities. The experimental results show that the algorithm can detect the structural holes and communities accurately and efficiently.

Keywords: structural hole; social networks; conductance; label propagation algorithm; minimal cut

1. Introduction

We are living in an online era, and many people are surfing online social networks to make friends, study, do academic research, or engage in other activities to satisfy their social needs at different levels. Scholarly data can be easily accessed. More powerful data analysis technologies must be developed. The interconnectedness of individuals in different communities has a significant impact on the lifespan and sustainability of the community [1,2]. The structure that acts as a bridge or tie between individuals of different communities tends to allow access to a richer supply of information and determines whether to allow the information from one group to diffuse to another; therefore, it is important to detect structural holes. Burt [3], who studied the social structures of many organizations, first provided the notion of structural holes as a means to bridge diverse groups and lead to benefits and termed the vertices lying on those positions as structural hole spanners. In social networks, users who bridge different communities are known as structural hole spanners. Structural holes are fundamental in many applications, and several models have been developed [4–6]. In viral marketing, structural holes can accelerate new product marketing to different groups [7,8]. Discovering the structural holes from real large-scale networks accurately and efficiently is a challenge that has attracted the attention of researchers.

There are many models that detect structural holes. However, the nodes identified by the existing model in the social network are not necessarily occupants of the structural hole spanner, but may also

be the central node of the network. In Figure 1, the larger blue node is a typical structural hole spanner, while the larger yellow node is not a structural hole spanner but has similar more specific features. It is necessary for us to detect the structural holes more accurately and remove the core nodes from the results.
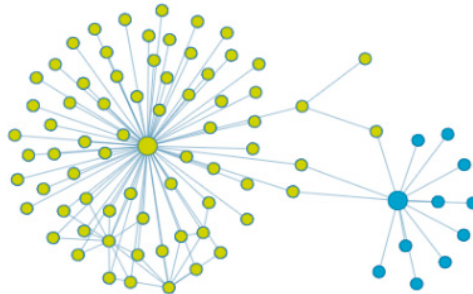

Figure 1. Illustration of structural holes.

In this paper, our contributions are as follows:

(1)   We present a model called the Conductance-Degree structural hole detection algorithm (CD-SHA), which uses conductance and degree to detect structural holes and uses conductance to detect the local minimum communities (LMCs)

(2)   We propose an improved label propagation algorithm based on conductance (C-LPA) to recognize communities in a network and filter the structural hole results.

We use real datasets to evaluate the performance using evaluation indicators, such as constraint, effective size, efficiency, clustering coefficient, and hierarchy. Experimental results show that the structural holes detected by the algorithm act as a bridge between communities in real large-scale social networks. Additionally, the evaluations show that the algorithm performs well regarding accuracy and robustness.

The remaining parts of this paper are arranged as follows. Section 2 discusses related studies and introduces basic notations. Section 3 proposes a solution to the problem. Section 4 introduces the dataset, and then analyzes and evaluates the performance and results of the algorithm. Section 5 presents the study's conclusions.

2. Related Work

2.1. Structural Holes

The concept of structural holes was first proposed as a sociology notion by Burt [3] and was later refined. Goyal et al. [9] proposed a model that is appropriate for star networks. However, social networks do not use a star topology. Those researchers determined that the vertices that lie on a large number of the shortest paths are more likely to be the structural hole spanners, which is similar to betweenness centrality. Kleinberg et al. [10] designed a decreasing function of the number of paths using the length between two neighbors to avoid the star topology, but this model requires careful tuning of the parameters. Because the structural hole spanners are the bridges or ties to connect several groups, there has been a series of studies relying on communities to identify them.

For example, Rezvani et al. [11] devised two fast but scalable linear time algorithms for the problem using both the bounded inverse closeness centrality of the vertices and articulation points of the network. Gong et al. [12] proposed a new solution to identify structural holes based on user profiles and user-generated content through machine learning methods. Wei et al. [13] provided a new improved method to identify structural holes according to the features of a temporal network, while considering nodes as topological, temporal path, and temporal subgraph between the nodes.

2.2. Label Propagation Algorithms

The label propagation algorithm (LPA) was originally proposed by Zhu et al. [14]. This is a semi-supervised learning method based on a graph. The idea of this algorithm is to predict the tag information of other unmarked nodes through the marked node tag information for community detection.

The LPA has been shown to be a highly efficient approach to community detection due to its near-linear time complexity and simplicity. Additionally, the process of label propagation simulates the information dissemination in the network. However, the sequence of nodes for the LPA is important. Different sequences may have different efficiency values and may lead to different results. In this paper, we use conductance to improve the LPA and capture the information about the communities in networks to filter the structural hole results.

Zhu [14] developed the LPA algorithm as a graph-based semi-supervised learning model. The algorithm takes advantage of the information regarding the labels that have been known to predict unknown labels. Barber et al. [15] developed the modularity-specialized label propagation algorithm (LPAm) to avoid allocating all of the nodes into the same community. Those researchers introduced the notions of hop attenuation and node preference to prevent large communities. Kouni et al. [16] simulated a special propagation and filtering process using information deduced from the properties of nodes to detect overlapping communities. Lin et al. [17] proposed an efficient community detection method based on the label propagation algorithm with community kernel (CK-LPA). These researchers discussed the composition of weights, the label updating strategy, the label propagation strategy, and the convergence conditions. Chen [18] proposed a novel label propagation algorithm by iteratively employing a teaching-to-learn and learning-to-teach (TLLT) scheme. Those authors manipulated the propagation sequence to move from the simple to the difficult and determined the feedback-driven curricula. Yang et al. [19] proposed a graph-based label propagation algorithm for community detection. Wang et al. [20] proposed a two-step algorithm with an adjustable parameter based on clustering coefficient and label propagation. The first step is to prioritize the nodes according to their degree and clustering coefficient, and initialize the label according to the ranking result. The second step is based on the first step. In order to avoid randomness, the neighbor nodes are sorted according to their clustering coefficient and degree, and the optimal neighbor node is selected to update the label.

2.3. Definitions and Notations

It is necessary for us to introduce several fundamental notations and background regarding social networks before we formally explain our model. Conductance and degree are often used to develop communities or cluster in social networks. These parameters explain the influence and importance of nodes. The conductance describes the topology structure of nodes in the network.

The expression $G = (V, E)$ represents an undirected connected graph, where V is a set of vertices and E contains the edges representing the relationships between those vertices, given two sets of vertices S, T, with no common part between them. $E(S, T)$ is a set that represents the edges between the two groups and $\text{cut}(S, T)$ represents the cut of the two sets, that is, the number of the edges between S and T. The conductance of a cluster is defined as the probability that a one-step random walk begins in one cluster and finally leaves that cluster. S_bar is the complement of S. The conductance of the set S and S_bar, denoted $\phi(S)$, is as follows:

$$\phi(S) = \frac{\text{cut}(S)}{\min(\text{d\_sum}(S), \text{d\_sum}(S\_bar))} \tag{1}$$

There is $\phi(S) \in [0, 1]$ and $\phi(S) = \phi(S\_bar)$. cut(S) represents the cut of S and S_bar. d_sum(S) represents the sum of degrees of the vertices in S. If given deges(S) is twice the number of edges among vertices in S, we have the following:

$$\text{edges}(S) = \text{d\_sum}(S) - \text{cut}(S) \tag{2}$$

Let us define a single vertex v's neighborhoods as $N(v) = \{w|d(w, v) = 1\}$, where $d(w, v)$ represents the length of the shortest path between w and v. Now, put v and $N(v)$ into a group as a neighbor community of v. If the conductance of the neighbor community of vertex v is smaller than the conductance of the neighbor community of any neighbor vertex w, the neighbor community of v is an LMC. Additionally, to the notation of conductance, the lower the conductance, the fewer the cut(S). That indicates fewer communications with others and more information exchange within the group; it is more likely to be a community, so it is appropriate to consider an LMC as an original community. The LMC can be explained as follows:

$$\forall w_i \in N(v), \phi(N(v)) \leq \phi(N(w_i)) \tag{3}$$

where $N(v)$ represents the neighbor community of v and $N(w_i)$ represents the neighbor community of w.

Conversely, the more the conductance, the more the cut(S) is, which indicates that the neighbors of the node have more communications with others than with the node's other neighbors, as shown in Figure 2. There are few relations between the node in dark color and its neighbors, whereas there are more relations both in the left and right groups. In this paper, we consider both conductance and degree to detect structural holes.
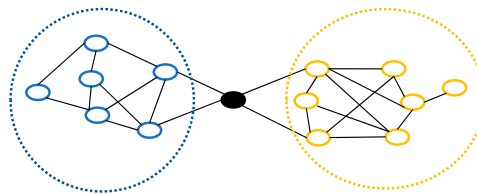

Figure 2. Case of greater conductance. The node in the dark color's neighbor community has greater conductance than its neighbors.

3. Conductance–Degree Structural Hole Detection Algorithm

In this section, we propose a new algorithm to detect structural hole spanners. This algorithm can avoid mistakenly identifying central nodes of social networks as structural hole spanners. Furthermore, through five common evaluation methods, our algorithm is superior to the other four common structural hole detection algorithms. We first computed the conductance and degree of the nodes and calculated the score (CD-score) according to the CD-SHA. The larger the CD-score, the more likely that the node was a structural hole spanner. Next, we identified the LMC structure in a social network to start the C-LPA and detect communities in the network. Next, we considered the position of the nodes in the network and filtered those nodes that did not access communities. Finally, we identified the structural hole spanners according to their CD-scores after filtering. Figure 3 illustrates the process of the algorithm.

3.1. Conductance and Degree Score

The larger the conductance value, the more relations exist among the neighbor community and other groups, and the more nodes have an association with the vertex and the more information per path. In real social networks, if those vertices with large conductance are lying on the edge of the communities, then they are more likely to be structural hole spanners.

It is easy to determine the degree of each node when we load the vertices and edges into memory. The greater the degree, the more importance and influence the node has on the networks. However, not all of the nodes with large degrees are structural hole spanners. Some of them are the core of the communities. In this paper, we computed a CD-score that refers to both the conductance and degree. We denoted α and β as the regulatory factors, and we indicated that $\alpha + \beta = 1$. The bigger the α, the more influence the detected nodes have, and the bigger the β, the more accuracy the detected

nodes have. In our experiments, the α was 0.3 and the β was 0.7. The larger the conductance and the degree, the greater is the CD-score:

$$s = \alpha \cdot \frac{d(v)}{g} + \beta \cdot \phi(v) \tag{4}$$

where s is the CD-score, d(v) is the degree of node v, $\phi(v)$ is the conductance of node v's neighbor community, and α and β are the regulatory factors.
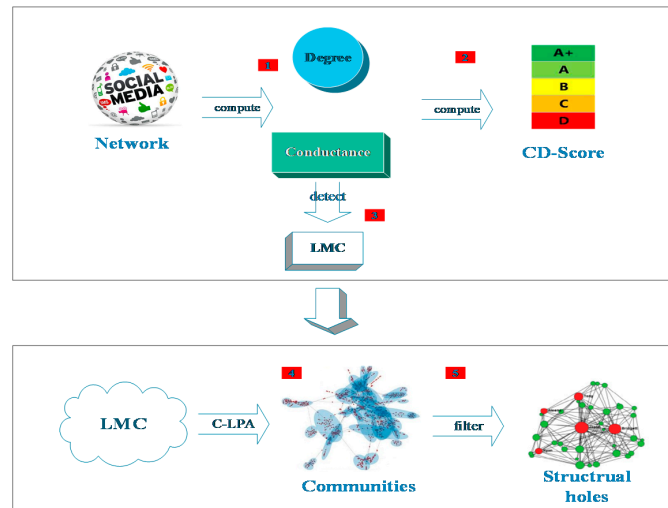

Figure 3. Illustration of the Conductance–Degree structural hole detection algorithm (CD-SHA). CD-SHA includes 5 steps: (1) Compute the conductance and degree; (2) Calculate CD-score; (3) Identify the local minimum community (LMC) structure in the network; (4) Detect communities by the label propagation algorithm based on conductance (C-LPA); (5) Identify the structural hole spanners according to their CD-scores after filtering.

Algorithm 1 provides a method to compute the conductance. It takes $O(n)$ time. Computing the degree of a node to detect structural holes uses different approaches but achieves equally satisfactory results in Goyal's [10] and J. Tang's [21] work. Although they describe the node's message passing ability, the degree is easier to compute, and we improve the method with conductance. However, because the core of the communities has a large conductance and degree, we need more information regarding the relative position of the vertex in the communities.

Algorithm 1

Input: Network (Nodes and edges)
Output: $\phi(S)$
1:　Initialize node List
2:　For each node in List do
3:　　　　if node is in v-neighbor do
4:　　　　　　　d_sum(S) += 2;
5:　　　　　　　edges(S)++;
6:　　　　Else
7:　　　　　　　d_sum(S)++;
8:　　　　　　　cut(S)++;
9:　　　　End if
10: End for
11:　　　$\phi(S) = \frac{cut(S)}{min(d\_sum(S), d\_sum(S\_bar))}$
12: Return

3.2. C-LPA and CD-SHA

The original LPA has many disadvantages, such as the different sequences of vertices resulting in different results of communities. However, we solved this problem by detecting the LMCs as the original communities before we started spreading the labels from those original communities in the C-LPA.

According to the notion of conductance, the lower the conductance, the fewer communications with others and the more information exchange within the group, and the more likely it is to be a community. The detailed process to compute the LMCs is described as Algorithm 2.

Algorithm 2

Input: the original Network (nodes and edges)
Output: $\phi(v)$ and the original Community structure
1: For each node in List do
2:　　　Get the node's neighborhoods
3:　　　Compute $\phi(v)$
4: End for
5: For each node v in List do
6:　　　For each node's neighborhoods w in List do
7:　　　　　if　　$\phi(v) > \phi(w)$
8:　　　　　　　flag = false
9:　　　　　　　continue
10:　　　　　　else
11:　　　　　　　　flag = true
12:　　　End for
13:　　　if flag = true
14:　　　　　add v to result
15:　　　flag = true
16: End For
17: Return

The dominant running time of the algorithm above computes the conductance of the neighbor community of each vertex $v \in V$ and later compares it with its neighbors. We assumed that there were n vertices and each vertex had m neighbors, and it would take $O(mn)$ time. In real social networks, according to the heavy-tailed degree distributions, most nodes have few neighbors, and least nodes have many neighbors, and it is true that $m << n$. We previously computed the conductance for structural holes, so there is little extra cost.

By the end of Algorithm 2, we identified several independent LMCs. Next, we assigned each LMC a unique label and allocated a random label to the other nodes, as is illustrated in the left side of Figure 4. Next, we started the C-LPA with the LMCs. A simplified overview of the process is shown in Figure 4. The right side of the graph shows the situation after the C-LPA is executed. Different colors represent different communities.
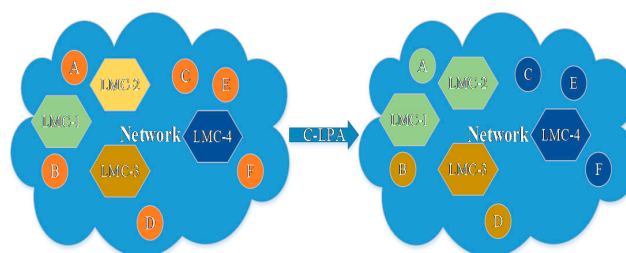

Figure 4. Illustration of the C-LPA. (1) Assign each LMC a unique label and allocate a random label to the other nodes; (2) Start the label propagation and detect the communities.

For the CD-SHA, we defined the structural holes (SHs) to satisfy Equation (5) and across communities; s represents the CD-score.

$$\forall v \in SH, s_v > s_{neighbor} \tag{5}$$

CD-SHA

Input: Network
Output: Community structural and structural holes
1: Allocate unique labels to each LMC
2: Allocate random labels to the other nodes
3: Initialize the List (LMCs are in the front of the List and the other nodes are in the back)
4: While Current Labels! = Last Time Labels do
5:　　　　For each node in List do
6:　　　　　　Compute the Neighbor nodes' community label
7:　　　　　　Update the label
8:　　　　　　if on the edge
9:　　　　　　　compare the CD-score
10:　　　　End for
11: End While
12: Return

While executing the C-LPA, we compared each node's CD-score with its neighbors and found those nodes that did not have a lower CD-score than their neighbors as structural hole candidates. If a candidate crossed at least two communities, we marked it as a structural hole spanner. The CD-score of the vertex told us which nodes exchanged more messages in a social network and the C-LPA informed us about the communities in the network. By the end of the CD-SHA, we identified communities with different labels and structural holes. The algorithm required a linear time similar to that of the LPA.

4. Dataset and Experiment

4.1. Dataset

To evaluate the performance of the proposed algorithms, we prepared several real-world datasets, which are listed in Table 1, namely the dolphin social network and the college football network.

Table 1. Six different real datasets.

Dataset	Edge Info	Node Count	Edge Count
Dolphins	Communication	62	159
Football	Competition	115	616

Dolphin social network. The dataset owns 62 nodes from two dolphin families. Lusseau observed those dolphins for seven years and recorded the relationship between each pair of dolphins. The relationship can be described as 159 edges in the dolphin network.

College football network. The dataset describes the USA college league football match in 2000. There are 115 teams and 616 games in the network. All of the teams were divided into 12 groups according to the geographical situation of the United States. There were many games both within a single group and among groups; therefore, this network is very close to the random network.

4.2. Related Algorithms

We compared the following methods for detecting the structural hole spanners with the CD-SHA.

- Path Count [11]: for each node, the algorithm counted the average number of shortest paths (between each pair of nodes), and then selected those nodes with the highest number as the structural hole spanners.
- Two-step connectivity [22]: for each node, the algorithm counted the number of pairs of neighbors that were not directly connected. Next, those nodes that had high numbers were identified as structural hole spanners.
- PageRank: PageRank can estimate the importance of a webpage. The algorithm used PageRank [22] to compute the importance of every node and then selected those nodes with high PageRank scores as the structural hole spanners.
- CD-SHA: for the network, it computed the conductance and degree score of each node and compared it to its neighbors to identify the larger ones as structural hole candidates. Next, it used the C-LPA to detect communities and filtered the candidates. If the candidates were on the edge of the communities and had an association with at least two groups, the candidates were confirmed as the structural hole spanners.

4.3. Evaluation Indexes

To evaluate the proposed algorithm, we have considered the following performance metrics:

Constraint (CT). The network constraint coefficient uses the degree of dependence of nodes on the other nodes as the evaluation criteria. The greater the value, the stronger the constraint, the stronger the dependence, and the lower the ability to cross the structural hole.

$$C_{ij} = \left(P_{ij} + \sum_q P_{iq} P_{qj} \right)^2 \tag{6}$$

Node q is a common neighbor of node I and node j. P_{ij} is the weight of node j between the neighbors of node i. The constraint of node i is as follows:

$$C_i = \sum_j C_{ij} \tag{7}$$

Effective size (ES). The effective size measures the overall influence of the node. This index measures the importance of the structural hole quantitatively:

$$ES_i = \sum_j \left(1 - \sum_q P_{iq} P_{jq} \right) = n - \frac{1}{n} \sum_j \sum_q P_{jq} \tag{8}$$

where n is the degree of node i, j represents a neighbor node of i, and q is a common neighbor of nodes i and j.

Efficient (EF). Efficient describes the impact of nodes on other nodes in the network. In other words, the efficiency of the nodes in the structural hole is relatively large.

$$EF_i = \frac{ES_i}{n} \tag{9}$$

Clustering coefficient (CC). According to the notation of the structural hole, the greater the clustering coefficient value is, the lower the possibility that the node is a spanner:

$$C(i) = \frac{2E(i)}{k(i)[k(i) - 1]} \tag{10}$$

where $E(i)$ represents the edges of the node i and $k(i)$ is the degree of the node i.

Hierarchy (HI). Hierarchy describes part of the features of the structural hole nodes, and the greater the value, the smaller the possibility the node is a spanner:

$$\text{HI}_j = \frac{\sum_j (C_{ij}/\frac{C}{N})ln(C_{ij}/\frac{C}{N})}{Nln(N)} \tag{11}$$

where C_{ij} is the constraint of the nodes i and j and C is the constraint of the node i.

4.4. Results and Analysis

Figure 5 shows the results of four algorithms on the dolphin network. Different colors in the graph represent different communities, and the green nodes are the structural hole spanners detected by the algorithm. There are two communities and two structural hole spanners in the picture. Regarding the results of the CD-SHA and the Path Count algorithm, the green nodes act as bridges between the groups in the network and each structural hole spanner connects at least two communities. The results of the PageRank and two-step algorithm are in the same community. The number of the structural hole spanners is significantly less than the total number of nodes. That indicates that a few special nodes in the network control much of the information diffusion.
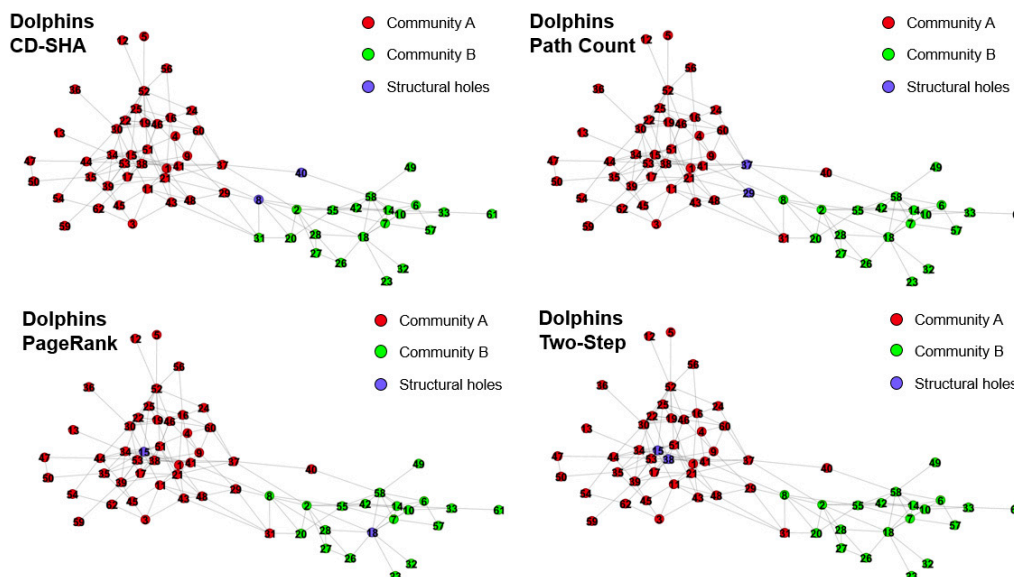

Figure 5. Structural holes in the dolphins' networks detected by the CD-SHA algorithm, the Path Count algorithm, the two-step connectivity algorithm, and the PageRank algorithm.

We computed the constraint, the effective size, the efficient, the clustering coefficient, and the hierarchy of the structural hole spanners detected by the Path Count algorithm, the two-step connectivity algorithm, the PageRank algorithm, and the CD-LPA algorithm. Figure 6 shows the results for the college football network.

Figure 6 shows the performance of different algorithms regarding the constraint coefficient, the efficiency, the clustering coefficient, and the hierarchy in the college football network. Different colors represent different nodes. We chose the top five results from the algorithm results to draw the picture. Regarding the effective size, the four algorithms had similar resultant values; the PageRank algorithm had the largest value and had the best results. Regarding efficiency, the four algorithms had similar resultant values; the CD-SHA had the highest value and was the best of the four. Regarding the constraint, the CD-SHA and the two-step connectivity algorithm had smaller values and were better than the other two algorithms. Regarding the clustering coefficient, the CD-SHA had the smallest value and was the best of the four algorithms. Regarding the hierarchy, the CD-SHA had the smallest value and was the best of the four. In general, the CD-SHA works well regarding the constraint,

the clustering coefficient, the efficiency, and the hierarchy, and has a performance similar to that of the other algorithms regarding the effective size. We then compared the average value of the CT, the ES, the EF, the CC, and the HI values, as shown in Figure 7.
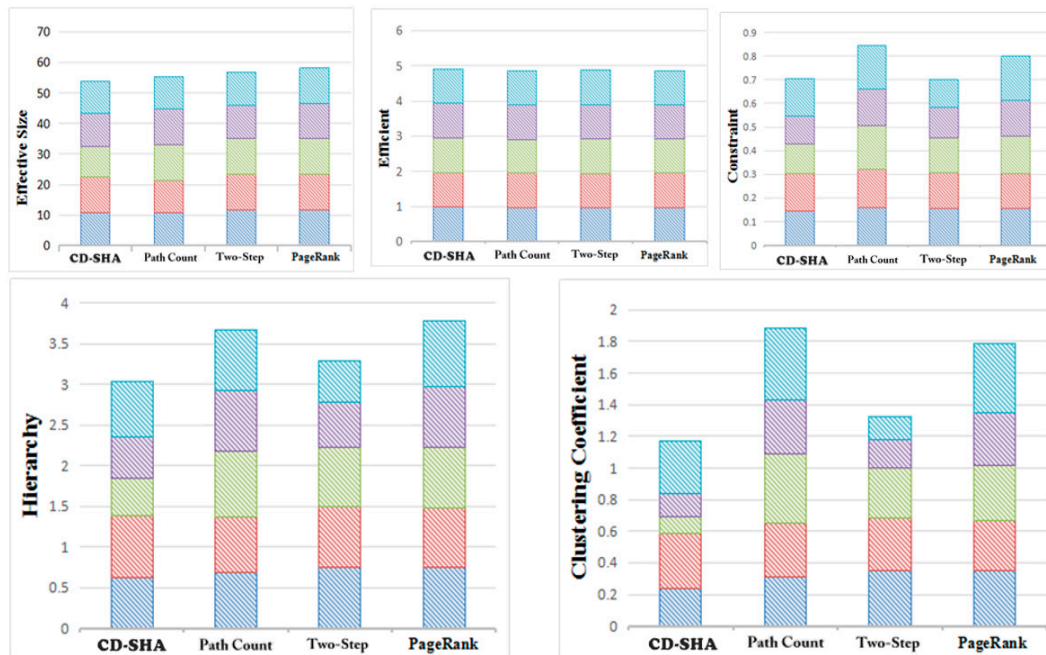

Figure 6. The performance of the CD-SHA algorithm, the Path Count algorithm, the two-step connectivity algorithm and the PageRank algorithm regarding the effective size, the efficiency, the constraint, the hierarchy, and the clustering coefficient in the football network.

(**a**)   (**b**)

Figure 7. Average values. (**a**) Dolphin social network and (**b**) college football network.

The average values of CT, HI, and CC of the CD-SHA in Figure 7, both in the dolphin social network and in the college football network, are lower than those of the other three algorithms. This finding suggests that the structural holes detected by the CD-SHA have better performance regarding CT, HI, and CC. For the EF in Figure 7, the average value of CD-SHA is close to that of the other three algorithms in the college football network and slightly bigger in the dolphin social network. This indicates that the structural holes detected by the CD-SHA are better than the others. Finally, regarding the ES, our algorithm has similar resultant values with the other three algorithms.

5. Conclusions and Limitations

In this paper, we studied how to develop the structural hole spanners in large-scale social networks. We first adapted the idea of conductance and the degree of the node to compute the CD-score in

order to detect the structural hole spanners. Next, we computed the LMC structure in the network as a seed for the C-LPA to filter the result. Next, we filtered the structural holes using the result of the C-LPA. Next, we applied the experiments to real datasets and observed the performance of the proposed algorithm. Finally, we evaluated the algorithm using quantitative indexes and analyzed the result. The results show that the proposed model captures the structural hole spanners efficiently and accurately in social networks. However, at the same time, our experiment has certain limitations. Our experiments are currently conducted on small social networks, and we may consider applying them to larger social networks in the future.

Structural holes play an important role in social networks and relate to a wide range of indicators of social success. For future studies, we need to address weighting networks. So many of the large real social networks are weighting networks, and if we ignore the weight of each edge or node, this results in deviation and mistakes. Furthermore, a visual analytics approach can better represent the location and role of structural holes in the network [23,24]. How structural holes can help social networking applications (such as recommendation, community evolution) warrants further investigation.

Author Contributions: Z.L., L.G., X.F. and Y.Z. contributed to the conception of the study; L.G. and C.T. performed the experiment; Z.L., L.G. and C.T. contributed significantly to analysis and manuscript preparation; L.G. and C.T. performed the data analyses and wrote the manuscript. All authors have read and agreed to the published version of the manuscript.

References

1. Liao, Z.; Zhao, B.; Liu, S.; Jin, H.; He, D.; Yang, L.; Zhang, Y.; Wu, J. A prediction model of the project life-span in open source software ecosystem. *Mob. Netw. Appl.* **2019**, *24*, 1382–1391. [CrossRef]

2. Liao, Z.; Deng, L.; Fan, X.; Zhang, Y.; Liu, H.; Qi, X.; Zhou, Y. Empirical research on the evaluation model and method of sustainability of the open source ecosystem. *Symmetry* **2018**, *10*, 747. [CrossRef]

3. Burt, R.S. *Structural Holes: The Social Structure of Competition*; Harvard University Press: Cambridge, MA, USA, 1992.

4. Wenbin, Z.; Tongrang, F.; Zhixian, Y.; Zijian, F.; Feng, W. An evaluation method of scientific research team influence based on heterogeneity and node similarity of content and structure. *J. Ambient Intell. Humaniz. Comput.* **2019**, 1–10. [CrossRef]

5. Yang, J.; Zhang, Y.; Liu, L. Identifying Opinion Leaders in Virtual Travel Community Based on Social Network Analysis. In Proceedings of the International Conference on Human-Computer Interaction, Cham, Switzerland, 26–30 June 2019; pp. 276–294.

6. Du, M.; Gao, H.; Zhang, J. Toward a guanxi-bases view of structural holes in sales gatekeeping: A qualitative study of sales practices in China. *Ind. Market. Manag.* **2019**, *76*, 109–122. [CrossRef]

7. Qin, Y.; Ma, J.; Gao, S. Efficient influence maximization under TSCM: A suitable diffusion model in online social networks. *Soft Comput.* **2017**, *21*, 827–838. [CrossRef]

8. Vaswani, S.; Lakshmanan, L.V.S. Adaptive Influence Maximization in Social Networks: Why Commit when You can Adapt? *arXiv* **2016**, arXiv:1604.08171. Available online: https://arxiv.org/abs/1604.08171 (accessed on 31 May 2020).

9. Goyal, S.; Vega-Redondo, F. Structural Holes in Social Networks. *J. Econ. Theory* **2007**, *137*, 460–492. [CrossRef]

10. Kleinberg, J.; Suri, S.; Tardos, E.; Wexler, T. Strategic Network Formation with Structural Holes. In Proceedings of the 9th ACM Conference on Electronic Commerce, New York, NY, USA, 8–12 July 2008; pp. 284–293.

11. Rezvani, M.; Liang, W.; Xu, W.; Liu, C. Identifying Top-k Structural Hole Spanners in Large-Scale Social Networks. In Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, New York, NY, USA, 19–23 October 2015; pp. 263–272.

12. Gong, Q.; Zhang, J.; Wang, X.; Chen, Y. Identifying Structural Hole Spanners in Online Social Networks Using Machine Learning. In Proceedings of the ACM SIGCOMM 2019 Conference Posters and Demos, New York, NY, USA, 19–23 August 2019; pp. 93–95.

13. Wei, W.; Wang, J.; Wang, H. Method for Detecting Nodes Influence Who Occupy Structural Holes in Temporal Network. In Proceedings of the 2019 5th International Conference on Information Management (ICIM), Cambridge, UK, 24–27 March 2019; pp. 113–117.

14. Zhu, X.; Ghanramani, Z. *Learning from Labeled and Unlabeled Data with Label Propagation*; Carnegie Mellon University: Pittsburghers, PA, USA, 2002.

15. Barber, M.J.; Clark, J.W. Detecting network communities by propagation labels under constraints. *Phys. Rev. E* **2009**, *80*, 26129. [CrossRef] [PubMed]

16. El Kouni, I.B.; Karoui, W.; Romdhane, L.B. Node Importance based Label Propagation Algorithm for overlapping community detection in networks. *Expert Syst. Appl.* **2019**, 113020. [CrossRef]

17. Lin, Z.; Zheng, X.; Xin, N.; Chen, D. CK-LPA: Efficient community detection algorithm based on label propagation with community kernel. *Phys. A Statist. Mech. Appl.* **2014**, *416*, 386–399. [CrossRef]

18. Gong, C.; Tao, D.; Liu, W.; Liu, L.; Yang, J. Label Propagation via Teaching-to-Learn and Learning-to-Teach. *IEEE Trans. Neural Netw. Learn. Syst.* **2017**, *28*, 1452. [CrossRef] [PubMed]

19. Yang, G.; Zheng, W.; Che, C.; Wang, W. Graph-based label propagation algorithm for community detection. *Int. J. Mach. Learn. Cybern.* **2019**, *11*, 1319–1329. [CrossRef]

20. Wang, M.; Xu, Y. Research on Label Propagation Algorithms Based on Clustering Coefficient. In Proceedings of the 2019 IEEE 4th International Conference on Cloud Computing and Big Data Analysis (ICCCBDA), Chengdu, China, 12–15 April 2019; pp. 348–352.

21. Tang, J.; Lou, T.; Kleinberg, J. Inferring Social Ties across Heterogenous Networks. In Proceedings of the fifth ACM International Conference on Web Search and Data Mining, Seattle, WA, USA, 8–12 February 2012.

22. Page, L.; Brin, S.; Motwani, R.; Winograd, T. *The Pagerank Citation Ranking: Bringing Order to the Web*; Stanford InfoLab: Stanford, CA, USA, 1999.

23. Liao, Z.; Kong, L.; Wang, X.; Zhao, Y.; Zhou, F.; Liao, Z.; Fan, X. A visual analytics approach for detecting and understanding anomalous resident behaviors in smart healthcare. *Appl. Sci.* **2017**, *7*, 254. [CrossRef]

24. Liao, Z.; He, D.; Chen, Z.; Fan, X.; Zhang, Y.; Liu, S. Exploring the characteristics of issue-related behaviors in github using visualization techniques. *IEEE Access* **2018**, *6*, 24003–24015. [CrossRef]