Weight-discounted Symmetrization in Clustering Directed Graphs

Bin He[1,2], Hui Liu[1], Xianghui Zhao[1], Zefeng Li[1,2]
[1]China Information Technology Security Evaluation Center, Beijing, China
[2]School of Information, Renmin University of China, Beijing, China
{binhe, cblizefeng111}@ruc.edu.cn, liuh@itsec.gov.cn, xianghuizhao@hotmail.com

***Abstract*—An increasing attention has been recently devoted to uncovering community structure in directed graphs which widely exist in real-world complex networks such as social networks, citation networks, World Wide Web, email networks, etc. A two-stage framework for detecting clusters is an effective way for clustering directed graphs while the first stage is to symmetrize the directed graph using some similarity measures. Any state-of-the-art clustering algorithms for undirected graphs can be leveraged in the second stage. Hence, both stages are important to the effectiveness of the clustering result. However, existing symmetrization methods only consider about the direction of edges but ignore the weights of nodes. In this paper, we first attempt to connect link analysis in directed graph clustering. This connection not only takes into consideration the directionality of edges but also uses node ranking scores such as authority and hub score to explicitly capture in-link and out-link similarity. We also demonstrate the generality of our proposed method by showing that existing state-of-the-art symmetrization methods can be derived from our method. Empirical validation shows that our method can find communities effectively in real world networks.**

Keywords-clustering; directed graph; graph transformations; community detection

I. INTRODUCTION

Many complex networks display community structure which reveals natural or underlying relationships between objects or nodes. Community structure is considered to be an important factor in understanding the dynamics and functions of complex networks [1]. Hence, finding community detection approaches or algorithms becomes an important topic which has attracted considerable attention from various fields like computer science, physics, etc.

Finding community structure in networks or community discovery can be converted to the problem of graph clustering aims at grouping sets of "related" vertices of the graph into clusters, communities or modules by taking into consideration the edge structure of the graph [5]. Most existing works about finding communities in networks mainly focused on undirected networks where edges have no specific direction as noted in [11]. However, many real-world complex networks and applications have implicit or explicit direction information between the objects which reveals asymmetric influence or information flow that we can't ignore, such as the citation networks, social networks, technological networks like the World Wide Web, etc. Hence, in some extent the effectiveness of any clustering algorithms in directed networks depends crucially on how to handle the direction of edges properly.

Ignoring the directionality of edges is the simplest way for clustering directed graphs. Obviously, this method isn't an appropriate solution in many real world networks. For example, if a person i follows person j but person j doesn't follow person i in social networks like Facebook, while simply ignore the direction of links means the relationships of person i and j are reciprocal, in fact it means person i may be interested in person j but it is not correct vice versa. Therefore, this method isn't accurate and may loss of important direction information.

In order to effectively capture and represent the direction of edges in directed networks, a common approach focused on finding new objective function. Based on this intuition, traditional objective function for clustering undirected graphs such as modularity or normalized cut, have widely extended to directed graphs [1, 3, 4, 13, 14]. Arenas [12] generalize the modularity function proposed by Newman and Girvan [8], Kim [1] proposed a generalized form of modularity in directed networks by introducing a new quantity LinkRank which is considered as the PageRank of links. Another common benefit function for optimizing community quality is normalized cut proposed by Shi and Malik [9]. Zhou [13] and Huang [14] have extended it to directed networks using random walks. Beside, Meila [3] introduced WCut which is a general class of weighted cut measures on graphs.

It is important to notice that modularity or normalized cut based approaches for clustering directed networks implicitly share a similar definition of community structure in which nodes are densely connected within community compared to the rest of networks. However, we can find meaningful clusters in directed networks even though nodes within the same community don't densely connect. As noted in [2, 15, 22], nodes have similar incoming and outgoing neighbors may have great similarity that should be assigned to the same community. Based on this intuition, Satuluri [2] adopted a two-stage framework for clustering directed graphs. First, transform the directed graphs into undirected graphs using some symmetrization methods, and then, the symmetrized graph can be clustered using existing state-of-the-art graph clustering algorithms. The advantage of the two-stage framework for detecting clusters in homogeneous directed graphs is that prior work on undirected graph clustering can be reused. Obviously, both symmetrization methods and existing undirected clustering methods may have an

Published by Atlantis Press, Paris, France.

important impact on the effectiveness of a two-stage framework for clustering directed graphs.

From the prospective of sysmmetrization, we find the degree-discounted symmetirzation which is the state-of-the-art symmetrization method has its weakness. First, the degree-discounted symmetrization approach assumed implicitly that the in-degree and out-degree information of nodes stand for the in-link weight and out-link weight of nodes respectively, however, the degree is only one basic metric of graph to discount the weight of hub nodes which generate many spurious connections in some domains and applications. Second, nodes have weights in some complex networks. Therefore, we think the weights of nodes are important for clustering directed graphs while the degree-discounted symmetrization didn't take into account.

In this paper, we propose a novel symmetrization method called Weight-discounted which take into account the weights of nodes. Our method is similar to degree-discounted symmetrization method and has proven to be more effective in some real networks. We are the first to combine link analysis with directed graph clustering and propose a novel concept on clustering directed weighted graphs by taking into consideration nodes weights. Furthermore, our method is a generalization of degree-discounted symmetrization method when the weight of nodes is the degree of nodes.

II. RELATED WORK

Graph clustering focuses on grouping sets of "related" objects of the graph into clusters. However, how to measure the "related" relationships between objects based on structural context? At the fundamental level approaches based on normalized cut assume that a best community will have more links within communities and fewer links to the rest of communities. From the view of random walk, some recent studies [1] revealed that a community is a group of nodes which a random walker is more likely to be trapped in. This is consistent with the assumption of normalized cut based methods. As we stated earlier, more meaningful clusters don't obey this rule. Generally objects or nodes within the same cluster are similar and dissimilar to objects from other clusters. However, "similar" or "dissimilar" are ambiguous. Different definition of similarity will lead to different definition of community in the domains of community detection and will have different community structure. If a domain displays community structure, which similarity measure method is more suitable for real-world complex networks? This is a question not only essential to the effectiveness of graph clustering but also important to the fundamental understanding of the complex networks.

The similarity measure becomes more complicated when we attempt to consider about the directionality of edges. Generally, clustering directed graph can be roughly put into two categories. The first category is directly build a model for the directed graph, and then optimize the model. For example, the metric-based approaches such as normalized cut or modularity belong to this category. Besides, [6] suggest a probabilistic model which can model both in-link and out-link for directed network community detection. The second category is to transform the directed graph to undirected graph and then use the undirected graph clustering methods. In the second category, any state-of-the-art clustering methods for undirected graph can be leveraged. In the essence, the symmetrization process of a two-stage framework for clustering directed graphs is equivalent to measure similarities among objects based on link structure. Therefore, it is possible and reasonable to transform directed graphs to undirected graphs using some symmetrization methods. Obviously, the effectiveness of clustering directed graphs depends crucially on the transformed undirected graphs while the quality of transformed undirected graph depends mainly on the symmetrization methods. Hence, a reasonable cluster generated by symmetrizing directed graphs depends not only on state-of-the-art clustering algorithms but also a reasonable symmetrization method.

We investigate various symmetrization methods for clustering directed graph, some methods will be discussed below. Let A be the adjacency matrix of original directed graph, and U be the adjacency matrix of the resulting symmetrized undirected graphs.

A. $A+A^T$

Simply ignoring the directionality of edges is the simplest way to obtain a symmetrized graph, while using the transformation $U = A + A^T$ to derive an undirected graph is similar to it except the weight of bidirectional edges would be the sum of the weights from both direction. For its simplicity, this method is implicitly used in other related works [3, 13, 24] when dealing with directed graph.

However, this method has the problem of losing important direction information in the process of symmetrizing directed graph. Besides, nodes don't have direct edges but have common in-coming or out-going neighbors won't cluster together in the symmetrized graph.

B. Random walk symmetrization

The symmetrized matrix U obtained from the prospective of random walk is as follows:

$$U = \frac{\Pi P + P^T \Pi}{2} \quad (1)$$

Here P is the transition matrix of the random walk that can be obtained by normalizing the rows of A. Denote π as its associated stationary distribution of P. We can obtain the diagonal matrix Π with π on the diagonal. This method is similar to $A+A^T$ since it has the same non-zero structure as $A+A^T$. Hence, the drawbacks of $A+A^T$ will also become the weakness of this method even though the actual weights on the edges may be different from $A+A^T$ symmetrization.

C. Bibliographic coupling and co-citation symmetrization

In the field of bibliometrics, co-citation and bibliographic coupling are two noteworthy methods for analyzing and understanding the patterns or relationships between scientific papers from their cross-citations. Kessler [16] introduced the bibliographic coupling method where the similarity between two papers p and q is computed by the number of papers cited by both p and q. The bibliographic coupling matrix B is

given by $B = AA^T$, where B(p, q) means the number of papers that paper p and q both point to in the directed bibliography network.

The co-citation schema which was proposed by Small [17] provides a new way to discover the patterns between documents. Similarity measure between two documents p and q is based on the number of documents that cite both p and q, the co-citation matrix is obtained from $C = A^T A$.

The two bibliometric methods are useful if we only want to analyze the in-link or out-link relationship of directed graph. On the other hand, any symmetrization method that needs to take into account both directions of edges wouldn't be effective if only adopt one of them.

D. Degree-discounted symmetrization

Since bibliometric coupling matrix AA^T capture the out-link similarity and the co-citation matrix $A^T A$ take into account in-link similarity, Satuluri [2] proposed bibliometric symmetrization via transformation $U = AA^T + A^T A$ which naturally takes the sum of both matrix to account for both. Generally, this transformation is more effective than bibliographic coupling or co-citation symmetrizations. As is well known, the hub nodes would generate many spurious connections in large scale power-low graphs, therefore, the bibliometric method works poorly facing with this situation.

Considering about this, Satuluri introduced degree-discounted symmetrization approach which incorporates the in-degrees and out-degrees of each node in the symmetrization process. The idea is that when two nodes i and j commonly point to a third node k, the similarity of i and j should be inversely related to in-degree of k. Similarly, when node h pointed by nodes i and j, the out-degrees of node h should be inversely to the similarity of node i and node j. The degree-discounted symmetrization matrix is defined as:

$$U = D_o^{-\alpha} A D_i^{-\beta} A^T D_o^{-\alpha} + D_i^{-\beta} A^T D_o^{-\alpha} A D_i^{-\beta} \quad (2)$$

Here, D_o and D_i are the diagonal matrix of out-degrees and in-degrees. α and β are the discounting parameters. Empirically $\alpha = \beta = 0.5$ work the best.

The degree-discounted symmetrization approach has proven to be effective in some real world complex networks such as citation networks, Wikipedia networks, etc. As we discussed in section 1, this approach has its advantages and disadvantages. In case of its weakness, on one hand, nodes often carry weight information in some real networks while the degree-discounted symmetrization method didn't take into account. On the other hand, the degree metric of nodes is not the only solution to measure and analyze networks or graphs. Motivated by this, we propose a generalization form of symmetrization method that can be suitable for clustering homogeneous directed graphs where nodes may carry weight information.

The generalization form of our proposed symmetrization method called Weight-discounted symmetrization which has proven to be more effective than degree-discounted method in some real world complex networks. For the sake of simplicity and easy to computation, the weight information of each node in our Weight-discounted symmetriztion method can be replaced by in-degree and out-degree of each node. In this situation, the Weighted-discounted method is relaxed to degree-discounted approach. Our method is a universal method for symmetrizing directed graph which the state-of-the-art clustering algorithms like spectral clustering or Markov clustering can be leveraged. Another notion motivated us is that the weights of nodes can be put into directed graph clustering.

III. THE PROPOSED TECHNIQUE

Instead of using the degrees of nodes to penalize hub nodes in degree-discounted symmtrization process, our proposed weight-discounted symmetrization attempt to adopt in-link and out-link weights of each node to discount the hub nodes. Hence, in this section, we first discuss the problem of the in-link and out-link weight of the nodes which is popular in link analysis; and then we present the weight-discounted method we use to symmetrize real world directed graphs.

A. Link analysis

Link analysis is another topic in directed network mining which has attracted a lot of attention. Generally, given the link relationships of nodes, the objective is to compute the node ranking score based on link structure. PageRank [20] which was developed by Brin and Page is one of the most popular node ranking algorithms. In PageRank, the total out-going weights of each node is the same and every link from node i is weighted by $1/D_o(i)$, where $D_o(i)$ is the out-degree of node i. PageRank is considered as a random surfer model which models two types of random jumps. A random surfer often follows the out-going links with probability p, and sometimes the surfer would jump to other nodes which not pointed by current node with probability 1-p.

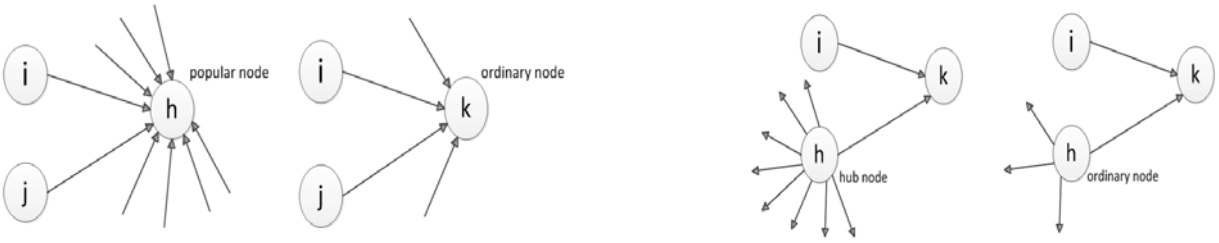
Another popular node ranking algorithm is HITS proposed by Keinberg [21]. Different from PageRank, every node in networks has two scores: hub score and authority score. The intuition behind HITS is that a good hub should point to many good authorities and a good authority is pointed by many good hubs. The iterative algorithm for computing hub score and authority score can be represented as the following operations,

$$\mathbf{a} = A^T \mathbf{h}, \mathbf{h} = A\mathbf{a} \quad (3)$$

where vector $\mathbf{a} = (a_{v1}, \cdots, a_{vn})^T$ and $\mathbf{h} = (h_{v1}, \cdots, h_{vn})^T$ contain the authority score and hub score of each node respectively. The final authority and hub scores of every node after the iterative processes are

$$\mathbf{a} = A^T A\mathbf{h}, \mathbf{h} = AA^T\mathbf{a} \quad (4)$$

Here, the authority vector $\mathbf{a}$ is the principal eigenvector of the authority matrix $A^T A$ and the hub vector $\mathbf{h}$ is the principal eigenvector of the hub matrix AA^T. Obviously the hub matrix AA^T and authority matrix $A^T A$ are corresponding to the bibliographic coupling matrix and co-citation matrix respectively. This is consistent with the symmetrization process in clustering directed graph. For simplicity, we use HITS algorithm to compute the weights of nodes where authority scores can be considered as



(a) Similarity between i and j (**left**) < (**right**) (b) Similarity between i and j (**left**) < (**right**)

Figure 1. Scenarios motivated weight-discounted

the in-link weight of nodes, and hub scores can be considered as the out-link weight of nodes. Note that HITS algorithm is not the only choice to compute the weights of nodes. Any other methods that can effectively compute the in-link and out-link weights of nodes can also be leveraged.

B. Weight-discounted symmetrization

Most existing directed graphs clustering algorithms ignore the weights of nodes or have an implicit assumption that nodes have the same weights. However, nodes are not equal in directed networks generally. The interactive relationships between two nodes may be reciprocal, unidirectional, or sometimes there are no direct links between them but own some common features, such as common neighbors, equal authority scores, etc. The implicit similarity computation of degree-discounted approach considering about the common in-going and out-going neighbors in directed graphs, this is reasonable and consistent with other studies [22] about link-based similarity measure in directed networks. We want our proposed symmetrization method incorporates the weights of each node in symmetrizing the directed graph. Hence, is it possible for us to present a novel symmetrization method for directed graphs that could incorporate the weights of nodes? The answer turns out to be yes.

When we obtain the in-link and out-link weights of nodes from any node ranking algorithms like HITS. The first step is to construct weight matrix. For a directed network, we denote the diagonal matrix of in-link weights by W_a with associated adjacency matrix A. Similarly, the diagonal matrix of out-link weights is denoted as W_h. In this paper, we obtain out-link weights from computing hub score and in-link weights from corresponding authority score. It is obvious that W_a and W_h are similar to the diagonal matrix of in-degrees D_i and out-degrees D_o respectively except the value in the diagonal.

The similarity between two nodes should be higher in the same cluster and lower in different clusters. Hence, a good symmetrization for clustering directed graph would place high weight on edges between nodes of the same community and set low weight on edges that in different communities. It is possible that two nodes have similar score or weight may have high similarity. For example, if we search "Albert Einstein" in search engine like Google or Bing, a good search engine would return pages relevant to the query and the resulting adjacent web pages have similar PageRank scores. Generally, the adjacent web pages are very similar. In order to introduce how the weights of nodes should be put into the symmetrization process, first we consider the following two scenarios (see Fig.1 (a)):

1) Nodes i and j both point to nodes h, which has a high authority score, that means h is a popular node in the original directed network.

2) Node k pointed by nodes i and j while node k has a low authority score in the original directed network.

Case 1 above is often seen in China's Twitter-like microblog like Weibo, a famous people often owns a lot of followers but that doesn't mean all followers are similar. On the other hand, if two people i and j both follow an ordinary people k that has a low authority score, intuition suggest that i and j may be more similar than case 1. Similarly, if nodes i, j, and h both point to node k, while node h has a higher hub score than node j, in other words, node h points to a lot of nodes besides node k. As seen in Fig.1 (b), node i will receive more contribution from node j than node h. Based on this intuition, we define the out-link weights based similarity between nodes i and j as follows:

$$W_{out}(i,j) = \frac{1}{W_h(i)^{\alpha} W_h(j)^{\alpha}} \sum_k \frac{A(i,k)A(j,k)}{W_a(k)^{\beta}} = \frac{1}{W_h(i)^{\alpha} W_h(j)^{\alpha}} \sum_k \frac{A(i,k)A(k,j)}{W_a(k)^{\beta}} \quad (5)$$

Here $W_h(i)$ is short for $W_h(i,i)$ which is the hub score of node i, and $W_a(j)$ is short for $W_a(j,j)$ which is the authority score of node j. $W_{out}(i,j)$ is the out-link similarity between nodes i and j based on weights of both nodes. α and β are the discounting parameters. We have made a small modification on W_{out} matrix and the final matrix format can be expressed as:

$$W_{out} = W_h^{\alpha} A W_a^{\beta} A^T W_h^{\alpha} \quad (6)$$

Similarly, the in-link weight matrix can be represented as $W_{in} = W_a^{\beta} A^T W_h^{\alpha} A W_a^{\beta}$. We naturally take the sum of in-link and out-link weights matrix, the final symmetrized matrix is defined as follows:

$$W = W_h^{\alpha} A W_a^{\beta} A^T W_h^{\alpha} + W_a^{\beta} A^T W_h^{\alpha} A W_a^{\beta} \quad (7)$$

Note that the above expression is similar to the degree-discounted symmetrization matrix. However, instead of using the in-degree and out-degree of nodes, we effectively take into consideration the weights of nodes. This is the biggest difference of our proposed method to the degree-discounted approach and we will discuss later that the degree-discounted symmetrization method can be derived from our weight-discounted symmetrization approach.

As we discussed in section 3.1, the weight matrix W_a and W_h can be obtained by some node ranking algorithms like HITS. Someone may doubt that every iterative process in HITS algorithm follows by a normalization process. Hence, the hub and authority score of each node obtained from HITS algorithm will be very small and almost close to zero if the directed graph is large scale. This may lead to the similarity score between two nodes turn to be very small. Hence, we use the logarithmic function defined as follows to maximize the similarity:

$$\mathrm{a}(i)=\begin{cases}1-\log_e \mathrm{a}(i), \mathrm{a}(i)\neq 0\\ 1, \mathrm{a}(i)=0\end{cases} \tag{8}$$

$$\mathrm{h}(i)=\begin{cases}1-\log_e \mathrm{h}(i), \mathrm{h}(i)\neq 0\\ 1, \mathrm{h}(i)=0\end{cases} \tag{9}$$

After above transformation, the weights will be greater than 1 and we have retained the edges in the original directed graph. We find the discounting parameters $\alpha=\beta=1$ would be more useful in some small real world network. While in large scale directed networks, the hub score and authority score would be small, and the absolute value of logarithmic function will be inversely very large so the final value of hub and authority. Hence, the value of α and β should smaller than 1 which can penalize the weights of node. Generally $\alpha=\beta=0.5$ work the best. This is consistent with degree-discounted symmetrization.

C. Relationship with existing symmetrization

Next we will describe the relationships between our proposed weight-discounted method and existing state-of-the-art symmetrization methods in clustering homogeneous directed graph. It turns out that biblometric and degree-discounted symmetrization methods can be considered as the special case of our method. Such connection demonstrates that our weight-discounted method provides a general framework to unify existing symmetrization methods.

Let I_h and I_a be the diagonal matrix of hub score and authority score of nodes respectively where all scores equal to 1. The weight-discounted symmetrization can simplify to the form as follows:

$$\begin{aligned}\mathrm{W}&=\mathrm{I}_h\mathrm{A}\mathrm{I}_a\mathrm{A}^T\mathrm{I}_h+\mathrm{I}_a\mathrm{A}^T\mathrm{I}_h\mathrm{A}\mathrm{I}_a\\ &=\mathrm{A}\mathrm{A}^T+\mathrm{A}^T\mathrm{A}\end{aligned} \tag{10}$$

The bibliometric symmetrization $\mathrm{U}=\mathrm{A}\mathrm{A}^T+\mathrm{A}^T\mathrm{A}$ as we describe in section 2 can't effectively deal with the influence of hub nodes. Because there exist an assumption in this symmetrized matrix that all nodes in directed networks have equal hub score and authority score. This is not true in real-world networks since power-law distribution of degrees exists in large networks.

The degree-discounted symmetrization method can also be derived from our method. If the hub score and authority score replaced by the inverse of out-link degrees in-link degrees respectively, we denote D_h and D_a as the diagonal matrix which the value is the inverse of out-degrees and in-degrees respectively. Note that set $\mathrm{A}=\mathrm{A}+\mathrm{I}$ prior to the symmetrization ensures the degrees are above zero. Therefore, the transformed symmetrized matrix will be the degree-discounted matrix as follows:

$$\begin{aligned}\mathrm{W}&=\mathrm{D}_h{}^{\alpha}\mathrm{A}\mathrm{D}_a{}^{\beta}\mathrm{A}^T\mathrm{D}_h{}^{\alpha}+\mathrm{D}_a{}^{\beta}\mathrm{A}^T\mathrm{D}_h{}^{\alpha}\mathrm{M}\mathrm{D}_a{}^{\beta}\\ &=\mathrm{D}_o{}^{-\alpha}\mathrm{A}\mathrm{D}_i{}^{-\beta}\mathrm{A}^T\mathrm{D}_o{}^{-\alpha}+\mathrm{D}_i{}^{-\beta}\mathrm{A}^T\mathrm{D}_o{}^{-\alpha}\mathrm{A}\mathrm{D}_i{}^{-\beta}\end{aligned} \tag{11}$$

As we can see from the above equation, our proposed weight-discounted method provides a general form of symmetrization methods for clustering directed graphs. Our innovation is that we put into node weights in the process of symmetrization while other symmetrization approaches don't take into account.

D. Pruning the symmetrized graph

For small-scale graphs, the time complexity and space complexity is not very high, so in order to retain complete information of links and weights, we needn't to do extra work. For large-scale real world networks, the resulting symmetrized matrix will have many non-zero elements so that it's costly for graph clustering algorithms to deal with. We suggest a threshold for our proposed method if the directed graph is very large. Generally, the symmetrized matrix will be densely connected if we set a low threshold. However, it's impossible and impractical for most clustering algorithms to cluster a dense matrix. On the other hand, it's hard for low performance computer to symmetrize large directed graphs without threshold or with low threshold.

In terms of large graphs, it's reasonable to pick a threshold to retain elements above the threshold. We find the weight-discounted symmetrized matrix is as easy as degree-discounted method to choose a suitable threshold.

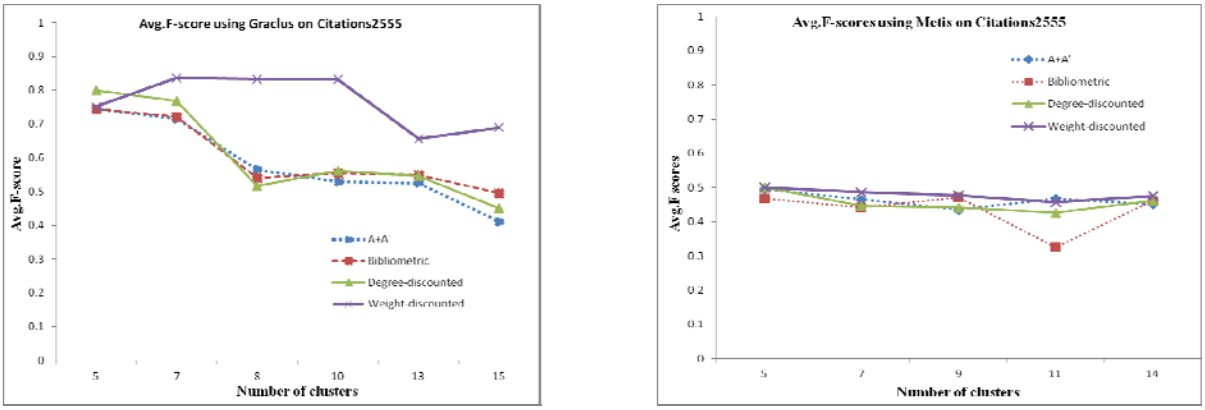
IV. Experimental Evaluation

In this section, we will introduce the evaluation method and the datasets we use. Finally, we will give a simple discussion involving the implementation.

A. Evaluation method

Considering about the ground truth, we use one commonly used metric called average F-measure for evaluating the performance of graph clustering output. We first give some description about precision and recall.

Denote T_k and S_k as the set of nodes in the k-th cluster come from the ground true category and the graph clustering output respectively. Given the true category $\mathrm{T}=\{\mathrm{T}_1,\mathrm{T}_2\cdots,\mathrm{T}_n\}$ and the graph clustering category $\mathrm{S}=\{\mathrm{S}_1,\mathrm{S}_2\cdots,\mathrm{S}_n\}$, then the precision and recall of one output cluster is defined as:



(a) (b)

Figure 2. Effectiveness of symmetrizations on Citations2555 using (a) Graclus and (b) Metis,as the clustering algorithms

$$\text{precision}(S_i, T_j) = \frac{|S_i \bigcap T_j|}{|S_i|} \quad (12)$$

$$\text{recall}(S_i, T_j) = \frac{|S_i \bigcap T_j|}{|T_j|} \quad (13)$$

Where $|\cdot|$ indicates the cardinality of a set. Then the F-score is the defined as:

$$\text{F-score} = \frac{2 * \text{precision} * \text{recall}}{\text{precision+recall}} \quad (14)$$

Generally each graph output cluster S_i will obtain a highest F-score $F(S_i) = \max_j \{F(S_i, T_j)\}$. We record the highest F-measure of each output cluster and then use it to compute the average F-score:

$$\text{Avg.F-score} = \frac{\sum_i |S_i| * F(S_i)}{\sum_i |S_i|} \quad (15)$$

B. *Datasets*

We use a real world dataset to evaluate our method. This dataset is a paper citation network obtained from [23]. This directed graph consists of 2555 vertices and 6101 edges. In what follows, we call this dataset Citations2555. Besides the graph of paper citations, the papers have already classified into 10 research topics of CS (such as Data Mining, Information Retrieval), which have discovered by an author-conference-topic model and available at arnetminer.org.

C. *Implementation*

We ran experiment comparing our proposed approach with several other symmetrization methods applicable to symmetrize directed graphs. All clustering processes consist of two steps: first the asymmetric directed graph matrix is transformed to symmetric matrix, then the symmetric matrix would be clustered using some existing undirected clustering algorithms such as Graclus [24], Metis [25]. We obtained the latest version of those graph clustering software from authors' respective webpages. Different symmetrization methods we use to compare ours were written by JAVA, using sparse matrix representations. Note that we perform our experiments on a dual core machine of 3.1GHz processor speed and 4GB of main memory.

V. RESULTS ANALYSIS

The effectiveness of different symmetrizations using Graclus and Metis is shown in Fig.2.

As seen in Fig.2 (a), the Avg. F scores of our proposed weight-discounted symmetrization have a distinct advantage over other symmetrization methods using Graclus. Other symmetrizations have similar Avg. F scores which are difficult to judge good from bad. In Fig.2 (b), we can see that our proposed symmetrization have a slight superiority than other methods even though the Avg. F scores are very low using Metis as the clustering algorithm. It is obvious that Graclus is better than Metis in graph clustering.

We also examine the effect of parameters α and β in Table I. For ease of comparison, we fixed the number of clusters at 10. When $\alpha = \beta = 1$ the average F-score is about 0.831, and average F-score is also good when $\alpha = \beta = 0.5$.

TABLE I. EFFECT OF VARYING α, β (USING GRACLUS). THE BEST RESULTS ARE INDICATED IN BOLD

α	β	Avg.F-score using Graclus
0	0	0.557113468
0.25	0.25	0.554377408
0.25	0.5	0.574694395
0.25	0.75	0.554325802
0.5	0.25	0.549671017
0.5	0.5	0.603897803
0.5	0.75	0.544767741
0.75	0.25	0.544889421
0.75	0.5	0.607495442
0.75	0.75	0.521429686
1.0	1.0	**0.831569118**

VI. CONCLUSION AND FUTURE WORK

A two-stage framework of clustering directed graph depends crucially on the effectiveness of symmetrizing the directed graph and the state-of-the-art graph clustering algorithm. In this paper, we present a novel symmetrization method called weight-discounted which incorporates the weights of nodes into the process of symmetrization. On one hand, our method is complete in symmetrizing the directed graph, which not only captures the in-link and out-link similarity between nodes but also take into consideration the weights of nodes; on the other hand, our approach is a general form of bibliometric and degree-discounted symmetrization methods. For future work, we would like to extend our method from homogeneous directed graph to heterogeneous directed networks. Nodes often own metadata such as gender, interest, profession, etc. in directed networks like Facebook or Twitter. Hence, the metadata should be put into the computation of nodes weights and that would have a great impact on the similarity among different nodes.

REFERENCES

[1] Y. Kim, S.-W. Son, and H. Jeong, "LinkRank: finding communities in directed networks" Phys. Rev. E, Vol. 81, 016103, 2010.

[2] V. Satuluri and S. Parthasarathy, "Symmetrizations for clustering directed graphs," In Proceedings of EDBT, 2011, pp. 343-354.

[3] M. Meila and W. Pentney, "Clustering by weighted cuts in directed graphs," Proc. of the 7th SIAM International Conference on Data Mining, 2007, pp. 135–144.

[4] E. A. Leicht and M. E. J. Newman, "Community structure in directed networks," Physical Review Letters, Vol. 100, 118703, 2008.

[5] S.E. Schaeffer, "Graph clustering," Computer Science Review, Vol. 1, 2007, pp. 27–64.

[6] T. Yang, Y. Chi, S. Zhu, Y. Gong, R. Jin, "Directed network community detection:A popularity and productivity link model," SIAM International Conference on Data Mining, 2010, pp. 742–753.

[7] U. von Luxburg, "A tutorial on spectral clustering," Statistics and Computing, Vol. 17, pp.395–416,2006.

[8] M. E. J. Newman, M. Girvan, "Finding and evaluating community structure in networks," Physical Review E, Vol. 70, 066111, 2004.

[9] J. Shi and J. Malik, "Normalized cuts and image segmentation," IEEE Conference on Computer Vision and Pattern Recognition, 1997, pp. 731–737.

[10] S. Gregory, "An algorithm to find overlapping community structure in networks". In *PKDD*, 2007, pp. 91-102.

[11] S. Fortunato, "Community detection in graphs" Physics Reports, Vol. 486, pp. 75–174, 2010.

[12] A. Arenas, J. Duch, A. Fernandez, and S. Gomez, "Size reduction of complex networks preserving modularity," New Journal of Physics, Vol. 9, No 6, pp176–180, 2007.

[13] D. Zhou, J. Huang, and B. Scholkopf, "Learning from labeled and unlabeled data on a directed graph," ICML'05, pp. 1036–1043, 2005.

[14] J. Huang, T. Zhu, and D. Schuurmans, "Web communities identification from random walks," Lecture Notes in Computer Science, vol. 4213, p. 187, 2006.

[15] M. E. J. Newman and E. A. Leicht, "Mixture models and exploratory analysis in networks," Proceedings of the National Academy of Sciences of the United States of America, Vol. 104, pp. 9564–9569, 2007.

[16] M. Kessler, "Bibliographic coupling between scientific papers," American Documentation, Vol. 14, pp. 10–25, 1963.

[17] H. Small, "Co-citation in the scientific literature: A new measure of the relationship between documents," Journal of the American Society for Information Science, Vol. 24, pp. 265–269, 1973.

[18] D. Chakrabarti and C. Faloutsos, "Graph mining:Laws, generators, and algorithms," ACM Comput. Surv., Vol. 38(1), 2006.

[19] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On power-law relationships of the internet topology," ACM SIGCOMM, pp.251-262, Sep 1-3, Cambridge MA, 1999.

[20] S. Brin and L. Page, "The anatomy of a large-scale hypertextual Web search engine," Computer Networks and ISDN Systems, Vol. 30, pp. 1-7, 1998.

[21] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," Journal of the ACM, Vol. 46, pp. 604–632,1999.

[22] G. Jeh and J. Widom, "SimRank: A measure of structural-context similarity," In ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, New York , 2002, pp. 538–543.

[23] J. Tang, J. Sun, C. Wang, and Z. Yang, "Social Influence Analysis in Large-scale Networks," In Proceedings of the Fifteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2009, pp. 807–816.

[24] I. S. Dhillon, Y. Guan, and B. Kulis, "Weighted Graph Cuts without Eigenvectors: A Multilevel Approach," IEEE Trans. Pattern Anal. Mach. Intell., Vol. 29, pp.1944–1957, 2007.

[25] G. Karypis and V. Kumar, "A fast and high quality multilevel scheme for partitioning irregular graphs," SIAM Journal on Scientific Computing, Vol.20, pp. 359–392, 1998.