Department of Computer Science
George Mason University
Technical Report Series

4400 University Drive MS#4A5
Fairfax, VA 22030-4444 USA
http://cs.gmu.edu/
703-993-1530

Detection of Communities and Bridges in Weighted Networks

Tanwistha Saha tsaha@gmu.edu
Carlotta Domeniconi carlotta@cs.gmu.edu
Huzefa Rangwala rangwala@cs.gmu.edu

Technical Report GMU-CS-TR-2010-15

Abstract

Traditional graph-based clustering methods group vertices into discrete non-intersecting clusters under the assumption that each vertex can belong to only a single cluster. On the other hand, recent research on graph-based clustering methods, applied to real world networks (e.g., protein-protein interaction networks and social networks), shows overlapping patterns among the underlying clusters. For example, in social networks, an individual is expected to belong to multiple clusters (or communities), rather than strictly confining himself/herself to just one. As such, overlapping clusters enable better models of real-life phenomena. Soft clustering (e.g., fuzzy c-means) has been used with success for non-graph data, when the objects are allowed to belong to multiple clusters with a certain degree of membership. In this paper, we propose a fuzzy clustering based approach for community detection in a weighted graphical representation of social and biological networks, for which the ground truth associated to the nodes is available. We compare our results with a baseline method for both multi-labeled and single labeled datasets.

1 Introduction

Many real-world data, e.g. social networks [12, 28, 31], biological networks [23] and collaboration networks [19], can be represented as graphs which can be further analysed to explore the properties of those networks. For many years, physicists and mathematicians have been actively studying the statistical properties that many networks have in common. One such property is the presence of structural sub-units (a.k.a. *communities*) which are highly interconnected and which can be identified by graph-based clustering methods (the terms *community* and *cluster* are synonymous in case of networks). Mining the *community structure* of a network has been a popular field of research for the past few years. For example, knowing the groups or communities within a social network can be used to infer about the trends of collaboration between individuals in academia as well as in industry. In biological sciences, uncovering the nature of interactions between group of proteins in a Protein-Protein Interaction (PPI) network will lead to understanding the function of key biological processes. Hence, the challenge in the community mining area is to explore a wide range of well-structured heterogeneous networks, where the community structure is not clearly evident, and is difficult to predict using traditional clustering methods.

In many applications, a given vertex in a graph, representing an individual connected with other people in the network, can belong to multiple clusters with a certain *degree of membership*. The concept of *fuzziness* arises while computing these membership values. As per the convention of fuzzy logic, the sum of all these membership values for a particular vertex must be one. In our approach, we aim to find these membership values for every vertex (node) in the graph (network). A weight can be associated to the edge connecting a pair of vertices, representing the association between the two corresponding entities (encoded by the vertices).

Of particular interest are the vertices which have high degree of membership for more than one cluster in the network. These nodes are called *bridges* [17, 18] between the communities. Identifying bridges in a network can help in a number of applications, as in a) finding proteins with a certain critical function in protein-protein interaction networks in biology; b) finding individuals who participate in different communities in social networks; and c) identifying malicious organizations who act as ne-

gotiators between terrorist networks. Communities in social networks represent grouping between individuals in social gatherings [12, 28, 31]. Communities in collaboration networks represent the collaborations between a group of researchers [19]. In this type of networks, associations between entities or individuals can be expressed as edges between the nodes. The graph itself can be considered as a collection of connected communities where there are certain individuals who act as negotiators between multiple groups, and can be considered as bridges between different communities. Being able to detect the structure of networks would help us exploit these properties more effectively. To this end, we formulate an optimization problem which leverages the weights associated with the edges. The solution to the problem aims to find the true clusters of the nodes in the network.

The rest of this paper is organized as follows. In Section 2 we discuss background and related work. Section 3 explains relevant mathematical notations and provides details of our proposed approach. Section 4 describes the experimental setup, and the results are reported in Section 5. Section 6 discusses ideas for future work and concludes paper.

2 Background and Motivation

Techniques for identifying groups or communities within a network can be classified into two different categories: (i) *graph partitioning* based approaches [5, 14, 15], and (ii) *modularity scoring* based approaches [1, 3, 4, 6, 21, 29, 33]. Graph partitioning based methods generally partition different nodes into groups that share common features or topologies. However, when used for community identification, these approaches produce a hard partitioning of the networks, and thereby, not allowing overlap between communities. Graclus [5] is an efficient multilevel graph-partitioning approach for weighted graphs but produces hard clusters, and is not validated with multi-labeled datasets.

On the other hand, modularity-based clustering algorithms propose a cluster or group quality score derived from the topological structure of the network, or features extracted from properties of the nodes and edges. The modularity score is then optimized to produce high quality clusters or communities.

Community structure in networks were explored using *Edge betweenness* [7] which was computed as a function of edges between the nodes within a community. Recently, researchers have proposed another betweenness centrality measure known as *split betweenness* of vertices [8], which show good performance in community detection. However, all these methods do not consider weighted networks. Weights represent the degree of association between the corresponding pair of nodes, and can provide useful information for community detection. The work of Newman [20] explores weighted networks by reducing integer weights on edges to a multigraph (where each edge of weight n is replaced by n parallel edges), and applying the modularity measures developed for unweighted networks. Ensemble based approaches use a combination of different modularity scores to create independent cluster association matrices which can be combined using a graph-partition based approach [1, 6, 26]. Other approaches include spectral clustering [25, 30], symmetric non-negative matrix factorization [16] and density based algorithm [22] for community detection.

None of the methods discussed so far, take into account the concept of *bridgeness*, which is important for identifying the information flow between communities within a network. To detect the multiple membership of nodes within different communities, several approaches [13, 17, 18, 24, 33] use fuzzy sets [2, 11, 32] to explore the underlying structure of these networks. Within the context of protein-protein interaction (PPI) networks, a modularity measure was developed by determining hub-induced subgraphs [29]. In a tangible approach, a method was developed to distinguish between dense and sparse subgraphs in weighted networks to identify community structure [9].

In contrast with the previously developed approaches, we propose a method to perform fuzzy clustering of weighted graphs (denoted as FCWG). At the same time, we aim to identify those nodes which are potential bridges in the network. This proposed approach is close to a fuzzy clustering based algorithm developed for unweighted graphs [17, 18] (denoted as FCUWG). For the FCUWG algorithm [17], each node in the network is assumed to belong to multiple clusters with a *membership value* associated with each cluster. This can be mathematically expressed in the form of a matrix which has been referred as *fuzzy cluster profile* (or just *cluster profile*) in [17]. As a result, each node has a *cluster profile*, which is a vector of values expressing the probability of belonging to the various clusters. Our FCWG approach differs in the optimization function, allows unweighted as well as weighted networks, and is evaluated on network datasets which have multi-labeled information.

3 Methodology

All the aforementioned methods, either disregard the concept of bridgeness in the network, or provide efficiency of their approaches in single-labeled and unweighted social networking datasets only. We try to bridge the gap between these methods by proposing a

new technique which leverages the weights on the edges, and also tries to validate the concept of bridgeness by using multi-labeled data (for which we had the ground truth) to identify the overlapping structure of the network. We provide the technical details of our approach in this section.

3.1 Fuzzy Clustering of Weighted Graphs

Given a weighted graph in input, we aim to find a fuzzy clustering of the vertices of the graph. We assume we are given as input the number of clusters k to be found. Let N be the number of vertices in the graph. We also assume that any given pair of vertices is connected by at most one edge, and there are no self-loops in the graph.

We consider a representation of the vertices in the space of clusters (*cluster profile*):

$$\mathbf{c}_i = (c_{i1}, \ldots, c_{ik}) \tag{1}$$

where the component c_{ij} represents the probability that vertex v_i belongs to cluster j, and $\sum_{j=1}^{k} c_{ij} = 1$. The objective is to estimate the vectors $\mathbf{c}_i$, for $i = 1, \ldots, N$. The cluster profile matrix $\mathbf{C}$ is defined as follows

$$\mathbf{C} = [c_{ij}] \tag{2}$$

where $\mathbf{C}$ has N rows and k columns, N is the number of vertices in the graph, and k is the number of clusters. We formulate the problem as an optimization problem where the cost function is expressed in terms of the vectors $\mathbf{c}_i$ for $i = 1, \ldots, N$, i.e., the rows of the cluster profile matrix $\mathbf{C}$.

Two vertices are considered similar if their cluster profiles are similar. Thus, we measure the similarity between two vertices by computing the inner product between their cluster profiles:

$$s_{ij} = \mathbf{c}_i \cdot \mathbf{c}_j \tag{3}$$

which gives the probability that v_i and v_j belong to the same cluster. Conceptually, two similar vertices are likely to have a strong association between them, which corresponds to a high value of the weight on the edge connecting them. Thus, we leverage the weights on the edges to represent the association between the nodes.

We want to formulate the problem so that the cluster profiles associated to the vertices capture the similarity measure embedded in the weights of the graph edges. This is achieved by defining the following objective function:

$$f(\mathbf{C}) = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} (w_{ij} - \mathbf{c}_i \cdot \mathbf{c}_j)^2 \tag{4}$$

where $w_{ij} \in (0,1)$ is the weight of the edge connecting vertices v_i and v_j, $\sum_{j=1}^{k} c_{ij} = 1 \; \forall i$, and $c_{ij} \geq 0 \; \forall \; i, j$. The function $f(\mathbf{C})$ corresponds to the mean square error of the predicted cluster profile of the vertices.

To encourage solutions that assign vertices to a small number of clusters, we add a regularization term representing the entropy of the distribution of the cluster profile vector components for each vertex:

$$E(\mathbf{C}) = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} (w_{ij} - \mathbf{c}_i \cdot \mathbf{c}_j)^2 - h \sum_{i=1}^{N} \sum_{l=1}^{k} c_{il} \log c_{il} \tag{5}$$

subject to the same constraints $\sum_{j=1}^{k} c_{ij} = 1 \; \forall i$. The coefficient $h \geq 0$ is a parameter of the procedure. The minimization of the cost function $E(\mathbf{C})$ in equation (5) will aim to minimize the combination of the mean square error and of the entropy term. In particular, the minimization of the entropy term will tend to disregard those solutions of cluster profiles with equal probabilities for a vertex to belong to multiple clusters. We discuss further about the optimization process in the following sections.

3.2 Fuzzy Clustering of Unweighted Graphs

We compare our approach discussed in the previous subsection with the baseline method proposed in [17] for fuzzy clustering of unweighted graphs (FCUWG). In this section we briefly discuss the cost function introduced in [17], and our implementation strategy for the same.

Given an adjacency matrix $\mathbf{A} = [a_{ij}]$ of a graph G, the cost function f, expressed in terms of the fuzzy cluster profile matrix $\mathbf{C}$, can be defined as follows:

$$f(\mathbf{C}) = \prod_{i=1}^{N} \prod_{j=1}^{N} \begin{cases} 1 - \mathbf{c}_i^T \mathbf{c}_j & \text{if } a_{ij} = 0 \\ \mathbf{c}_i^T \mathbf{c}_j & \text{otherwise} \end{cases} \tag{6}$$

$f(\mathbf{C})$ is maximized under the constraints $\sum_{j=1}^{k} c_{ij} = 1$ $\forall i$, and $c_{ij} \geq 0 \; \forall \; i, j$, where v_i and v_j are any pair of vertices in the graph. To avoid the possibility of obtaining too small values (due to the presence of products in the cost function), in our implementation we maximize the $\log(f(\mathbf{C}))$.

3.3 Bridgeness Measure

Once the optimal fuzzy cluster profile matrix $\mathbf{C}$ for the given graph has been computed, we can analyze the resulting vectors $\mathbf{c}_i$ to quantify the degree to which a given vertex is shared among different clusters. This measure is called *bridgeness* of the vertex. As mentioned in [18], the bridgeness b_i of vertex v_i is defined as:

$$b_i = 1 - \sqrt{\frac{k}{k-1}\sum_{j=1}^{k}(c_{ij} - \frac{1}{k})^2} \qquad (7)$$

If a vertex belongs to all the clusters in the graph with equal probabilities, then the term inside the summation evaluates to zero, which in turn gives a bridgeness score of 1. This implies that ideal bridges in the network will belong to multiple communities with equal probabilities. We observe that vertices with low degree and high bridgeness usually correspond to *outliers* - for example, in the case of social networks, these are individuals who do not really belong to any community. To distinguish between the true bridges in the network and the outliers, we use another measure; δ-*corrected bridgeness* (defined in [17]), which is the product of the degree of a vertex and the bridgeness obtained previously from equation (7). Vertices having high δ-corrected bridgeness scores are the estimated true bridges in the network.

Table 1: Description of Datasets.

Dataset	#Nodes	#Edges	#Clusters	Average degree	Average classes
PPI-1	256	1583	5	12.3672	1.4023
PPI-2	116	501	8	8.6379	2.3017
Zachary	34	78	2	4.58	1

4 Experimental Setup

4.1 Datasets

Table 1 provides the description of the datasets we have used in our experiments to test the effectiveness of our method. "Average degree" denotes the average number of edges connected to each node in the entire dataset. "Average classes" indicates the average number of classes each node belongs to. We have selected two weighted sub-networks (denoted as PPI-1 and PPI-2) of randomly selected proteins from the Protein-Protein interaction (PPI) network derived from the BioGRID database [27]. These interaction networks have some proteins which belong to multiple classes (labels denote function of proteins); the column *Average Classes* gives an idea of the count. The PPI-1 network has 83 nodes which belong to more than one class. The PPI-2 network has 81 nodes which belong to more than one class. The PPI-1 network has 1 node which belongs to all the 5 clusters. The PPI-2 network has 3 nodes which belong to all the 8 clusters.

Zachary's karate club dataset [31] is a popular social network extensively used by researchers. Due to a dispute between the two instructors of the club, the original network of 34 individuals split into two groups. The Zachary's network is unweighted. We derive weights for the edges using equation (8) provided in the following sub-section. Unlike the PPI networks, all nodes in the Zachary's karate club network belong to one class only. We specifically selected these networks because of the availability of the ground truth regarding the cluster assignments.

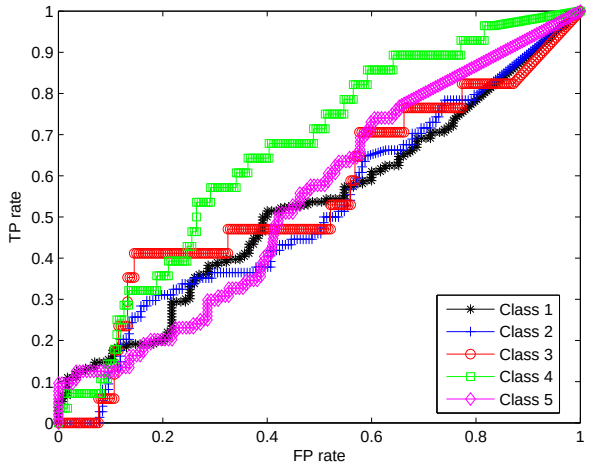

Figure 1: ROC curves for 5 classes in PPI-1 network having 256 nodes.

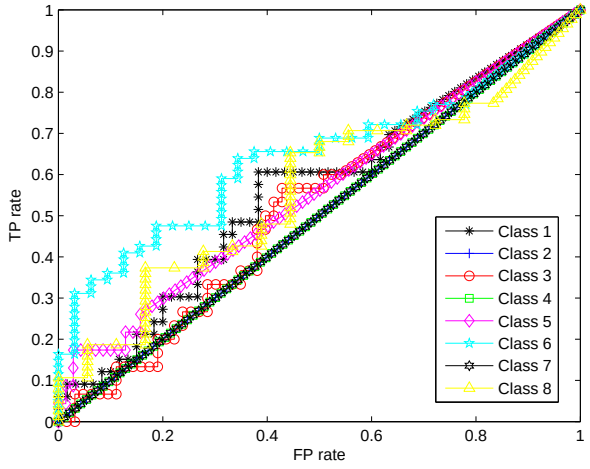

Figure 2: ROC curves for 8 classes in PPI-2 network having 116 nodes.

4.2 Edge weights

For unweighted networks, the adjacency matrix **A** provides a representation of the graph. In this case, we use the expression given in equation (8) to calculate the

weights of the edges in the graph. We aim to account for the effect of common neighbors (e.g., node k) to evaluate the strength of the association between nodes i and j. The more neighbors nodes i and j share, the stronger their association is. The effect of a common neighbor k is weighted by the inverse of its degree:

$$w_{ij} = \mathbf{A}_{ij} + \sum_{k \in N} \left(\frac{\mathbf{A}_{ik}}{D_i - \mathbf{A}_{ij}} \times \frac{\mathbf{A}_{kj}}{D_k} \right) \quad (8)$$

where N is the number of nodes in the network, $\mathbf{A}$ is the adjacency matrix for the graph, and D_i is the degree of the vertex i. This approach allows us to assign weights to the edges of any unweighted undirected graph, and to leverage those weights through our approach. In our experiments, we have used equation (8) to derive the weights for all the edges in the Zachary's karate club network. The PPI networks have weights associated with each edge, reflecting the number of wet-lab experiments where the interaction was observed between the corresponding pairs of proteins. In other words, it signifies the reliability of a particular protein-protein interaction. In all cases, the weight matrix was normalized before performing any further computation.

4.3 Label correspondence

Since clustering is an unsupervised problem, we first solve a label correspondence between the cluster labels found by our method, and the true cluster labels of the ground truth. To this end, we perform *defuzzification* to assign a particular cluster label to a node, i.e., we assign to a node the label of the cluster with the largest probability value in the corresponding row of $\mathbf{C}$. We then cluster the nodes accordingly, and sort the resulting clusters in non increasing order of their sizes. We do the same for the groups labeled according to the ground truth. Clusters with the same position in the two sorted lists give the label correspondences.

4.4 Evaluation Metrics

Some nodes in the PPI networks are multi-labeled. We analyzed the fuzzy cluster profiles obtained for all the nodes. To compute the fuzzy clustering accuracy of multi-labeled data, as well as single labeled data [31], we defined the following four different metrics:

- Top-1 accuracy

  For a particular node, we check its cluster profile: the cluster with the highest probability value is considered as the node's final cluster assignment. We then consider the ground truth labels, and check whether the set of true cluster labels of the node under consideration contains the identified label. If yes, we consider the assignment as correct. This is the most *restricted* evaluation metric.

- Top-m accuracy

  This is determined by finding the top m most probable cluster assignments for a particular node in the cluster profile matrix. We assign the node to the identified m clusters. We then consider the ground truth and check if these predicted m labels belong to the set of true labels for the node. The value of m is driven by the number of clusters present in the network, e.g., in our experiments, we have set $m = 2$ for PPI-1, and $m = 3$ for PPI-2.

- Any-1 accuracy

  For a particular node, we consider the clusters in the corresponding profile vector as predicted clusters, if their corresponding probability values are greater than $\frac{1}{k}$, where k is the number of clusters in the data. This is the most *relaxed* metric for accuracy measure.

- AUC score

  The AUC score is the normalized area under a curve (Receiver Operating Characteristic or ROC curve) that plots true positives against false positives for different possible thresholds for classification. For each class in a dataset, we plot ROC curve and compute the AUC score. We report the average AUC scores across all classes in a network.

4.5 Software

To solve the optimization problem we have used AMPL modeling language[1] and Quadratic Programming solver SNOPT[2] available online at NEOS server[3]. AUC scores were computed using PERF [4]. Analysis of output and plots were done in MATLAB[5].

[1] http://www.ampl.com/
[2] http://www.sbsi-sol-optimize.com/
[3] http://neos.mcs.anl.gov/neos/solvers/index.html
[4] http://kodiak.cs.cornell.edu/kddcup/software.html
[5] http://www.mathworks.com/

Table 2: Community identification accuracy across the datasets

Zachary's karate club network				
Method	Top-1	Top-m	Any-1	Mean AUC score
FCWG(avg h)	0.98 ± 0.0369	-	-	-
FCWG($h = 0.8$)	**1**	-	-	**1**
FCUWG	0.9706	-	-	0.9982
PPI-1 network				
FCWG(avg h)	0.2813 ± 0.0954	0.5685 ± 0.0834	$\mathbf{0.6877} \pm 0.0866$	-
FCWG($h = .85$)	**0.3625**	**0.6932**	**0.8087**	$\mathbf{0.5616} \pm 0.0578$
FCUWG	0.3068	0.5737	0.6693	0.4855 ± 0.1491
PPI-2 network				
FCWG(avg h)	0.4061 ± 0.0808	$\mathbf{0.7921} \pm 0.0656$	$\mathbf{0.8194} \pm 0.0477$	-
FCWG($h = 1$)	**0.5054**	**0.8387**	**0.9140**	$\mathbf{0.5442} \pm 0.0490$
FCUWG	0.4624	0.7527	0.7097	0.4958 ± 0.1678

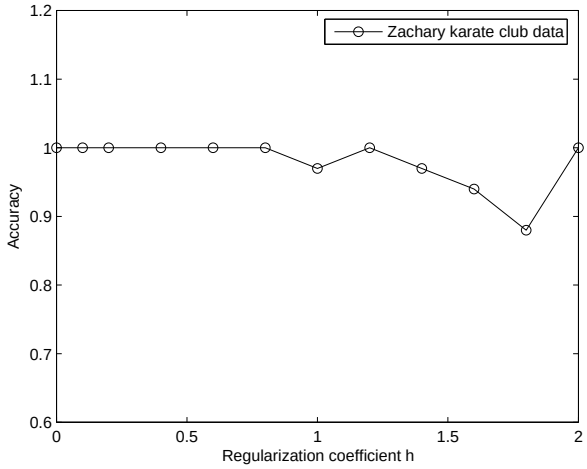

Figure 3: Sensitivity of accuracy w.r.t. h for the Zachary's karate club network.

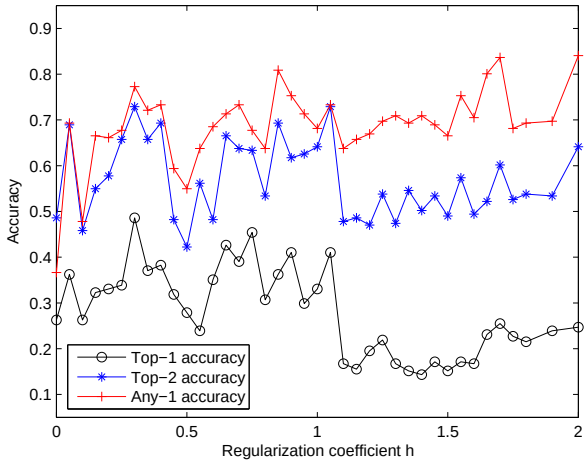

Figure 4: Sensitivity of accuracy w.r.t. h for the PPI-1 network.

5 Results

5.1 Accuracy

Table 2 shows the accuracy obtained for all the three datasets based on Top-1, Top-m, Any-1 metric and AUC score. All the values are averaged across different values of the regularization coefficient h varying from 0 to 2 in steps of 0.05. Since each node in the Zachary's karate club network is single-labeled, we have computed only the Top-1 accuracy for this dataset. For the PPI networks, we have reported the average accuracies based on the four different metrics, along with the standard deviations. The maximum accuracy achieved for each metric, and the corresponding value of h, are also reported. For analyzing bridges and plotting ROC curves, we fix h to the value that provided the maximum accuracy. In particular, $h = 0.8$ for the Zachary's dataset, $h = 0.85$ for PPI-1, and $h = 1.0$ for PPI-2 show the best Top-1 accuracy scores. The *Mean AUC score* gives the average area under the ROC curves for all classes in PPI networks, using our method.

Figures $1-2$ show the ROC curves for the PPI networks obtained for our approach. We also report the corresponding accuracy and Mean AUC scores computed using FCUWG [17]. The accuracies reported are higher for our method (FCWG) compared to the baseline FCUWG, provided that h is chosen appropriately. Comparing FCWG with FCUWG for specific values of h, we achieved 3% improvement in Top-1 accuracy for the Zachary's karate club network, 20% improvement in Top-m accuracy for the PPI-1 network and 11% improvement in Top-m accuracy for the PPI-2 network. Cases in which our method achieved the best accuracy are high-

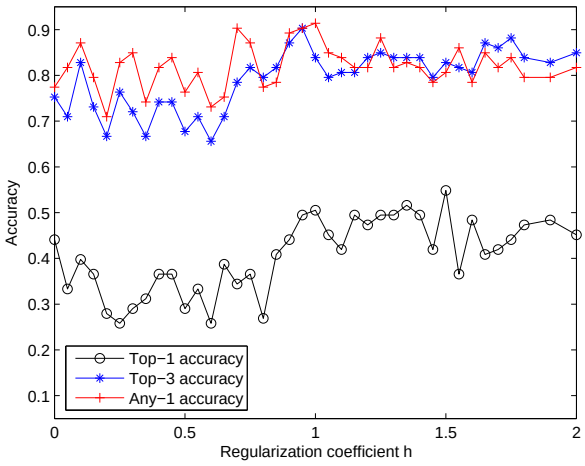

Figure 5: Sensitivity of accuracy w.r.t. h for the PPI-2 network.

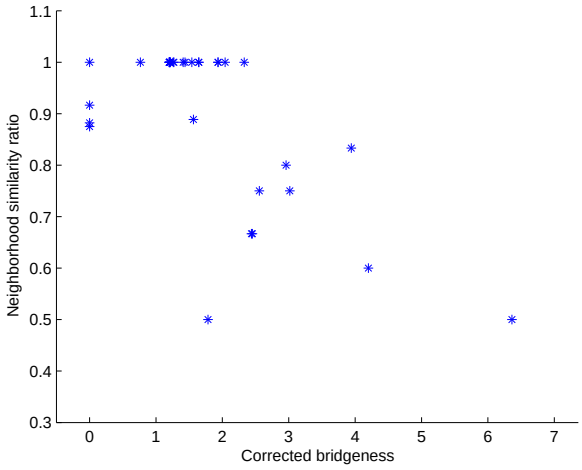

Figure 6: Zachary's karate club network: probable bridge between the two groups have high δ-corrected bridgeness score.

lighted in bold in Table 2.

5.2 Sensitivity Analysis

The purpose of the regularization coefficient h in equation (5) is to penalize solutions in which the vertices belong to all the clusters with equal probabilities. Hence, if we vary h, we are bound to obtain different solutions, and therefore different accuracy values for all the metrics. Table 2 reports the highest accuracy we obtained for all the three networks, with specific h values. Figures $3-5$ show the sensitivity of the accuracy with respect to h for all three networks.

While the optimal value of h depends in general on the data, for all the three datasets considered here a value close to 1 provided the optimal solution, i.e., achieved the optimal balance between the two terms in equation (5). For the Zachary's network, the accuracy is stable across different values of h. For each PPI network, the three accuracy measures reveal a similar trend across the h values. For the PPI-1 network, h values in the range $(0,1)$ provide on average higher accuracy. For the PPI-2 network, h values in the range $(1,2)$ give better results on average. Despite these differences, for both networks, a value of h close to 1 gives optimal results.

5.3 Bridgeness Analysis

To analyze the δ-corrected bridgeness score obtained from equation (7), we define a new metric called *Neighborhood Similarity Ratio (NSR)*, defined for each node in the network. Let v_i be a node of the network, L_i be the set of labels of v_i according to the ground truth, and n_i be the set of neighboring nodes of v_i. We define the function

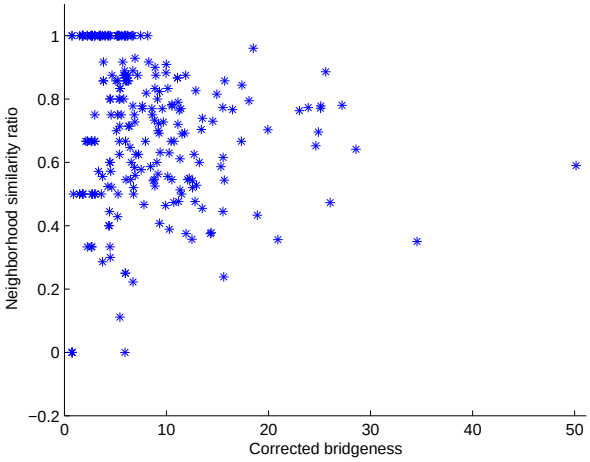

Figure 7: PPI-1 network: probable bridges between the groups have high δ-corrected bridgeness score.

$\mathbb{I}(.)$ for v_i as follows:

$$\mathbb{I}(L_i \cap L_j) = \begin{cases} 1 & \text{if } L_i \cap L_j \neq \phi \text{ and } v_j \in n_i \\ 0 & \text{otherwise} \end{cases} \quad (9)$$

The function $\mathbb{I}(.)$ counts the number of neighbors $v_j \in n_i$ that share at least one label with v_i. The Neighborhood Similarity Ratio (NSR) measure is defined as follows:

$$NSR(v_i) = \frac{\sum_{j=1}^{|n_i|} \mathbb{I}(L_i \cap L_j)}{\sum_{j=1}^{|n_i|} min(|L_i|, |L_j|)} \quad (10)$$

The denominator of (10) measures the maximum number of labels node v_i can possibly share with each of

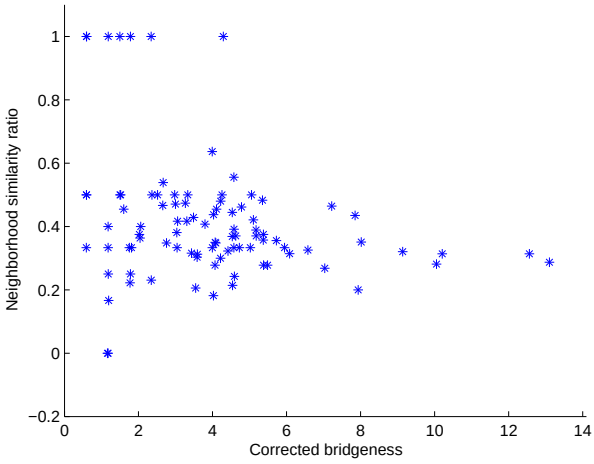

Figure 8: PPI-2 network: probable bridges between the groups have high δ-corrected bridgeness score.

it's neighbors. Figure $6-8$ show scatter plots of the NSR versus the δ-corrected bridgeness score for all the three networks. Ideally, most of the nodes in any network should have low degree compared to the degree of the bridge node, and hence their δ-corrected bridgeness score should be low. In the plots, these nodes tend to clutter towards the left, accounting for low bridgeness. These nodes do have a large NSR value because the NSR measure increases with the association of nodes in the same group. A community is strong if the nodes within that community are likely to have the same set of labels. Nodes having NSR score 1 are the centers of the communities because they have the majority of the labels in common with their neighbors. Nodes which have high bridgeness and high degree clutter towards the right end and are small in number. These are the true *bridges* of the network. For a node to act like a bridge, it should be connected to other nodes across multiple clusters. In the plots, we see that these nodes are the ones which get high δ-corrected bridgeness score and an approximate NSR score of 0.5.

Figure 9 shows the Zachary's Karate club network with a black node denoting the probable bridge in the network. The white nodes have lower scores, and gray nodes denote intermediate scores respectively for the δ-corrected bridgeness. As per our results, node 3 has maximum value of δ-corrected bridgeness, which means that this node is probably the bridge between the two groups after the club split. It is also the node which has NSR score approximately equal to 0.5 in the scatter plot of Figure 6. Compared to the ground truth in [31], this node was misclassified by the method proposed in [7] but our method could classify it correctly and also identify it as a bridge node.

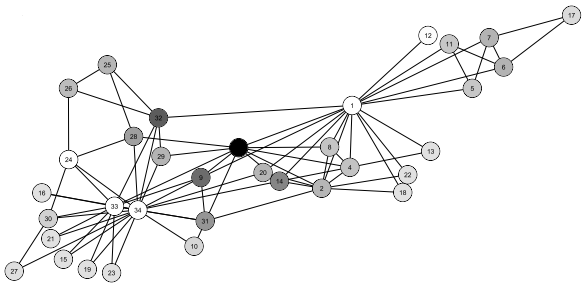

Figure 9: Zachary's karate club network: the black node is the probable bridge between two groups.

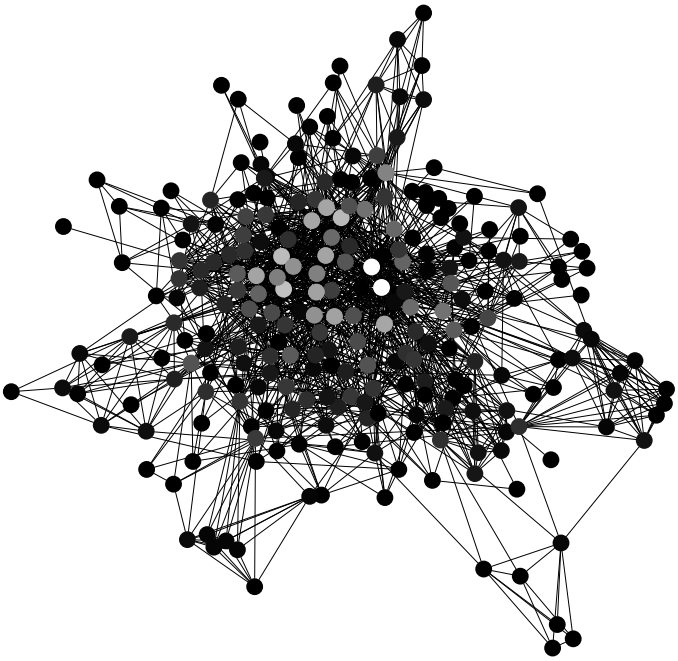

Figure 10: PPI-1 network: white colored nodes in the center have the maximum δ-corrected bridgeness score.

Figures $10-11$ shows the plots of the PPI networks with nodes colored according to the δ-corrected bridgeness values, but here the coloring convention is different. The white nodes denote high δ-corrected bridgeness scores and vice versa (in order to visualize the bridge nodes more prominently).

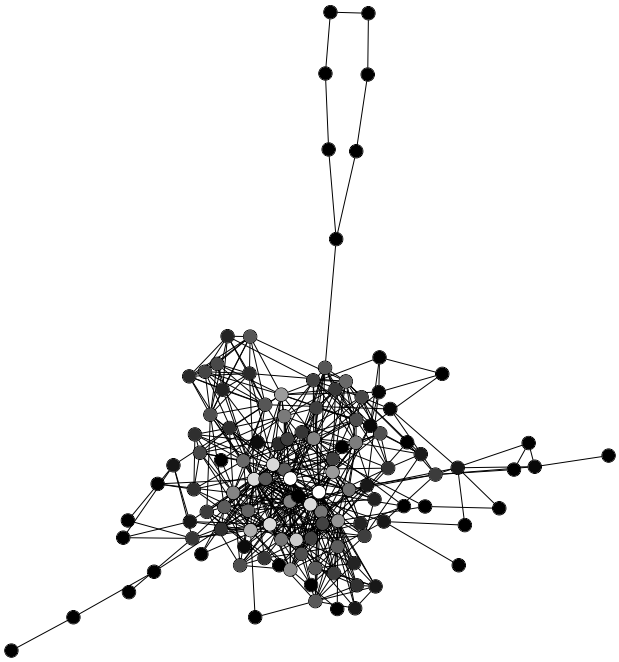

Figure 11: PPI-2 network: white colored nodes in the center have the maximum δ-corrected bridgeness score.

6 Conclusion and Future work

We have proposed a new approach for the identification of community structures in weighted networks of moderate size, and, at the same time, have compared our results with a baseline method discussed in [17]. The identification of bridges in biological network will serve as an important aspect in analyzing quite a few outstanding problems on protein-protein interactions in biological sciences.

The networks we have explored so far are limited in size, compared to social networks in the real world. For example, Facebook social networking site has millions of individuals connected in groups. One way towards scaling to large networks would be trying a hierarchical approach to reduce the size of the graph and then apply fuzzy clustering as mentioned in this paper. Since, we consider the degree of membership of the nodes to different clusters in the network, another interesting study would be to explore the statistical models for partial membership as discussed in [10], and compare our results for multi-labeled data to those.

Currently, for the method discussed in this paper, we need to know the number of communities present in the network beforehand, which is a problem for any unsupervised clustering technique. Automatic detection of the number of communities present in the network, and thereafter, analyzing the underlying information flow between the groups, is another interesting direction to pursue for future research.

7 Acknowledgment

Authors of this paper would like to thank Gaurav Pandey from University of Minnesota for sharing the weighted network obtained from BioGRID. This research is supported by NSF grant III−0905117 and by NSF CAREER IIS−0447814.

References

[1] S. Asur, D. Ucar, and S. Parthasarathy. An ensemble framework for clustering protein–protein interaction networks. *Bioinformatics*, 23(13):i29, 2007.

[2] J. Bezdek. Fuzzy mathematics in pattern classification. *Unpublished Ph. D. dissertation, Cornell University, Ithaca, NY*, 1973.

[3] J. Chen, O. Zaiane, and R. Goebel. Detecting communities in social networks using max-min modularity. *SDM 2009*, pages 978–989, 2009.

[4] A. Clauset, M. Newman, and C. Moore. Finding community structure in very large networks. *Physical Review E*, 70(6):66111, 2004.

[5] I. Dhillon, Y. Guan, and B. Kulis. Weighted graph cuts without eigenvectors a multilevel approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1944–1957, 2007.

[6] G. Duggal, S. Navlakha, M. Girvan, and C. Kingsford. Uncovering Many Views of Biological Networks Using Ensembles of Near-Optimal Partitions. *Proceedings of MultiClust: 1st International Workshop on Discovering, Summarizing and Using Multiple Clusterings, KDD*, 2010.

[7] M. Girvan and M. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences of the United States of America*, 99(12):7821, 2002.

[8] S. Gregory. An algorithm to find overlapping community structure in networks. *Knowledge Discovery in Databases: PKDD 2007*, pages 91–102, 2007.

[9] S. Gunnemann and T. Seidl. Subgraph Mining on Directed and Weighted Graphs. *Advances in Knowledge Discovery and Data Mining*, pages 133–146, 2010.

[10] K. Heller, S. Williamson, and Z. Ghahramani. Statistical models for partial membership. In *Proceedings of the 25th International Conference on Machine learning*, pages 392–399. ACM, 2008.

[11] F. Hoeppner. *Fuzzy cluster analysis: methods for classification, data analysis, and image recognition*. Wiley, 1999.

[12] P. Hoff. Random effects models for network data. In *Dynamic social network modeling and analysis: Workshop summary and papers*, pages 303–312, 2003.

[13] T. Hong, K. Lin, and S. Wang. Fuzzy data mining for interesting generalized association rules* 1. *Fuzzy sets and systems*, 138(2):255–269, 2003.

[14] G. Karypis and V. Kumar. Parallel multilevel k-way partitioning scheme for irregular graphs. In *Proceedings of the 1996 ACM/IEEE Conference on Supercomputing, 1996*, pages 35–35, 1996.

[15] B. Long, X. Wu, Z. Zhang, and P. Yu. Unsupervised learning on k-partite graphs. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 317–326. ACM, 2006.

[16] X. Ma, L. Gao, X. Yong, and L. Fu. Semi-supervised clustering algorithm for community structure detection in complex networks. *Physica A: Statistical Mechanics and its Applications*, 389(1):187–197, 2010.

[17] T. Nepusz, A. Petróczi, and F. Bazsó. Fuzzy Clustering and the Concept of Bridgedness in Social Networks. *Proceedings of the International Workshop and Conference on Network Science, NetSci*, 2007.

[18] T. Nepusz, A. Petróczi, L. Négyessy, and F. Bazsó. Fuzzy communities and the concept of bridgeness in complex networks. *Physical Review E*, 77(1):16107, 2008.

[19] M. Newman. Scientific collaboration networks. I. Network construction and fundamental results. *Physical Review E*, 64(1):16131, 2001.

[20] M. Newman. Analysis of weighted networks. *Physical Review E*, 70(5):56131, 2004.

[21] M. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical review E*, 69(2):26113, 2004.

[22] G. Palla, I. Derényi, I. Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814–818, 2005.

[23] E. Ravasz, A. Somera, D. Mongru, Z. Oltvai, and A. Barabási. Hierarchical organization of modularity in metabolic networks. *Science*, 297(5586):1551, 2002.

[24] J. Reichardt and S. Bornholdt. Detecting fuzzy community structures in complex networks with a Potts model. *Physical Review Letters*, 93(21):218701, 2004.

[25] J. Ruan and W. Zhang. An efficient spectral algorithm for network community discovery and its applications to biological and social networks. In *Seventh IEEE International Conference on Data Mining, ICDM 2007.*, pages 643–648. IEEE, 2008.

[26] E. Sawardecker, M. Sales-Pardo, and L. Amaral. Detection of node group membership in networks with group overlap. *The European Physical Journal B*, 67(3):277–284, 2008.

[27] C. Stark, B. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, and M. Tyers. BioGRID: a general repository for interaction datasets. *Nucleic acids research*, 34(suppl 1):D535, 2006.

[28] B. Thurman. In the office: Networks and coalitions* 1. *Social Networks*, 2(1):47–63, 1980.

[29] D. Ucar, S. Asur, U. Catalyurek, and S. Parthasarathy. Improving functional modularity in protein-protein interactions graphs using hub-induced subgraphs. *Knowledge Discovery in Databases: PKDD 2006*, pages 371–382, 2006.

[30] S. White and P. Smyth. A spectral clustering approach to finding communities in graphs. In *Proceedings of the fifth SIAM international conference on data mining*, page 274. Society for Industrial Mathematics, 2005.

[31] W. Zachary. An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33(4):452–473, 1977.

[32] L. Zadeh. Fuzzy sets*. *Information and control*, 8(3):338–353, 1965.

[33] S. Zhang, R. Wang, and X. Zhang. Identification of overlapping community structure in complex networks using fuzzy c-means clustering. *Physica A: Statistical Mechanics and its Applications*, 374(1):483–490, 2007.