Defining and Discovering Communities in Social Networks

Stephen Kelley, Mark Goldberg, Malik Magdon-Ismail, Konstantin Mertsalov, and Al Wallace

1 Introduction

The categorization of vertices in a network is a common task across a multitude of domains. Specifically, structural divisions into internally well connected sets have been shown to be useful in computer science, social science, and biology. In each of these areas, grouping vertices using structural boundaries helps one to understand the underlying processes of a network. Identifying such groupings is a non-trivial task, and a subject of intense research in recent years.

In general, identifying groups of vertices in a network based on structural properties alone is known as *community detection*. Methods to identify such groups take a wide variety of approaches, mirroring the diversity in domains where an accurate view of structural communities is useful. Depending on the definition of a community used, one could discover groups which maximize a global quality function, contain a specific set of substructures, or satisfy a set of local criteria. Each of these definitions has resulted in a number of methods which aim to produce the "best" set of communities relative to the definition chosen.

Rather than focusing on a number of features which differentiate these definitions and methods from each other, this text will focus on perhaps the most fundamental question in the field of community detection; should groups be disjoint or should they be allowed to overlap?

In the past, the field of community detection has primarily focused on identifying a set of groups such that each vertex in the network is assigned to a single group. Such a requirement results in a set of disjoint groups covering the entire network. However, with the explosion of social network and on-line communication

Stephen Kelley
Oak Ridge National Laboratory

Mark Goldberg · Malik Magdon-Ismail · Konstantin Mertsalov · Al Wallace
Rensselear Polytechnic Institute

data available, research has expanded towards methods which consider overlapping groups.

In the remainder of this text, we will first include a brief discussion on the intuition behind disjoint and overlapping communities as well as provide the reader with a basic understanding of a small sample of commonly used methods for community detection. Further into the text, we will present the difficulties involved when detecting overlapping communities and introduce a method for discovering overlapping communities which avoids these common pitfalls. This algorithm will be presented with results on real and synthetic benchmark networks. Finally, we will show that in real data, communities which do are natural and necessary to capture many of the associations between vertices in a network.

2 Methods for Detecting Community Structure

The most fundamental division between community definitions is whether or not vertices can belong to a single community or any number of communities. Justifications exist for each approach, and ultimately, the selection of which definition to use is likely domain and application dependent. For instance, when analyzing biological protein interaction networks, if an analyst wishes to generate a taxonomy of proteins, a hierarchical disjoint method is desired. When analyzing social networks, due to the variety of affiliations and interests that an individual may have, an overlapping method may be more appropriate.

We begin with a brief examination of some of the previous work in the area of community detection to give the reader a sense of current methods. This examination is far from complete; it is intended to serve only as a brief introduction. For a more comprehensive survey covering a variety of methods in depth, please see [8].

2.1 Disjoint Community Detection

The majority of current methods work treat the problem of locating communities as a hierarchical partitioning problem. According to this approach, the community structure of a network is assumed to be hierarchical; individuals form disjoint groups which become subgroups of larger groups until one group, comprising the whole society, is formed. Such methods for a tree of subgroup relations called a dendrogram. A dendrogram allows the community structure of a network to be at various resolutions. An example of this structure, which is commonly used as a visual tool for hierarchical clustering methods, is given in Figure 1.

Originally, the method for detecting a hierarchical grouping in networks was to repetitively identify edges which do not belong to the same dense subgraph [9, 20]. If we consider a group containing all individuals, and for each edge, compute the centrality according to one of a number of definitions (information, shortest path,

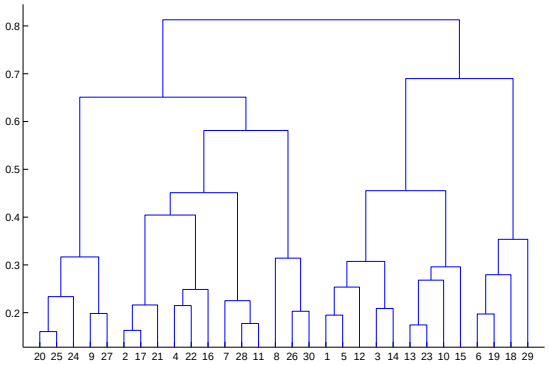


Fig. 1 Dendrogram visualization detailing the merging or splitting communities until the entire society is contained in a single group or until each community consists of a single individual.

circuit, betweenness, etc). Edges with higher centrality scores will be ones which link, rather than compose, dense areas of the network. Such edges are repetitively removed. Those edges removed first will be edges which form a significant connection between two dense areas of a network. This process of calculation and removal is performed until the graph becomes disconnected. Upon disconnection, a single group splits into two groups containing each component. This process is continued until each vertex is contained in a group by itself. As a result, a hierarchy of splits is produced, showing the relationship between small groups and larger ones.

This analysis can be quite useful for networks where visual inspection of the dendrogram provides an accurate picture of However, this method lacks the ability to point out precisely at what level of the hierarchy the "best" groups have been discovered. For large networks where visual inspection is impossible or for networks in which there exists no intuition to suggest the best set of groups, this fact is problematic. In order to determine the best split in an automated manner, the notion of *modularity* [16] has been proposed. This measure can be expressed as

$$Q = \frac{1}{2m} \sum_{i,j \in V} [A_{i,j} - \frac{k_i k_j}{2m}] \delta(c_i, c_j) \tag{1}$$

where m is the number of edges in a network, $A_{i,j}$ is the edge weight connecting vertex i and j, k_i is the degree of vertex i, and $\delta(c_i, c_j$ is a function returning 1 if the community assignments of vertex i and vertex j are the same and 0 otherwise. Intuitively, the measure expresses the difference between the number of edges inside communities and the number which are expected to be within a community, given a community's degree. Given this measure, one can compare the modularity of all

levels of the hierarchy and identify the most defined set of groups compared to the null model.

The introduction of modularity as an evaluation measure of group quality has resulted in a number of methods which attempt to optimize this value. The most well known of these methods is a greedy agglomerative method originally proposed by Clauset, Newman, and Newman [5]. This algorithm begins by placing vertices in unique communities and merging those which produce the largest increase in modularity. Additional methods have been proposed based on simulated annealing [11], extremal optimization [7], methods from statistical mechanics [22], and other heuristic optimizations [3]. Recent work has also identified a variety of non-hierarchical utilizing label propagation [21] and minimizing the amount of information needed to express random walks in a network [23].

2.2 Overlapping Communities

While hierarchical grouping is valid for some types of networks, *e.g.*, organizational networks or taxonomies, intuition and experience suggest that social networks contain pairs of communities that overlap while not containing each other as a sub-community. Consider an individual in a social network representing "friendship." He or she may have friendship relations across many different social circles, such as those formed in the workplace, by a family unit, by a religious group, or by social clubs. In this case, assuming social structure of the network to be hierarchical might lead to missing important information about members' attachment to the numerous social circles with which they concurrently interact.

However, the shift from disjoint community assignments to non-disjoint assignments is not a simple one. Various interpretations exist for how vertices can be assigned to groups. Specifically, there is some debate as to whether the goal is to identify a weighted assignment from an individual to all groups or a set of binary assignments indicating an individual's membership. The former has been used in identifying fuzzy groups via probabilistic assignment [6, 26] and maximizing an overlapping version of modularity [17]. Additional work has been done on finding the best set of communities such that each individual can only associate with k sets. An interesting algorithm based on label propagation can be found in [**?**]. This text however, will examine only the problem of deriving a set of binary individual to group mappings without such constraints. Such a mapping allows communities to be discovered at a local level, where a vertex's association with a group is determined independently of any association with other groups.

Methods which identify these non-fuzzy overlapping communities tend to be one of two types; either the algorithm attempts to identify instances of a specific structure in the network or a modularity value is calculated relative to a small subset of the network. It is important to notice that, unlike the global measure of modularity, each of these tasks is local in nature.

2.2.1 Clique Percolation

An example algorithm which attempts to identify a defined, local substructure which is indicative of a community is the Clique Percolation Method (CPM) originally proposed in [19]. In a nutshell, the algorithm first finds all cliques of size k, called k-cliques, and defines a k-clique graph whose nodes are the k-cliques. Two nodes are adjacent in the k-clique graph if the corresponding cliques share $k-1$ nodes. The nodes in the union of the k-cliques corresponding to each connected component are declared to be a community. For $k = 2$, clique percolation defines the communities as the connected components in the network.

CPM attempts to discover communities by identifying complete subgraphs of size k. One can claim that, for reasonably sized values of k, such substructure is clearly an instance of community structure. However, this definition sets a very rigid definition for a community. If one edge of a otherwise complete subgraph is missing or if two k-cliques overlap by only $k-2$ nodes, it is not considered a community. Clique percolation would not, for example, be able to find the group illustrated in the toy community in Figure 2. The main problem with such a definition is that it is too rigid and is uniform over the whole network, requiring all communities to share the same structural composition. Additionally, identifying k-cliques of arbitrary sizes can be very expensive computationally.

2.2.2 Local Optimization

In an effort to identify communities of various composition, new methods have been proposed based on the notion of local optimality. Generally, these methods begin with some set of seed groups which are then optimized relative to a local density function. The seed groups are considered communities when a single vertex addition or removal does not increase the group's quality relative to a density function.

Despite a large number of proposed methods for detecting communities via local optimization [2, 4, 15], there has been a general agreement in the form of the density function used to optimize seed groups. Intuitively, the search for community structure can be viewed as a search for sets of individuals which are intensely connected relative to their isolation with the rest of the network. Specifically, this can be expressed in a manner representative of the functions in previous literature as the ratio of edges internal to the set over all edges connected to the set. This can be given as

$$d(S) = \frac{w_{in}}{w_{in} + w_{out}} \tag{2}$$

where w_{in} is the number of edges internal to the set S and w_{out} is the number of edges connecting the set S to the rest of the network. This and similar density functions are essentially local modularity measures which attempt to maximize internal while minimizing external edges.

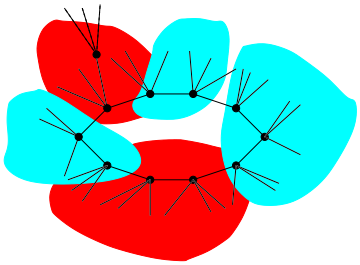


Fig. 2 A demonstration of local optimality

Methods based on local optimization add and remove a vertex relative to a set's density when the vertex is evaluated. The implications of this will be discussed at length later in the chapter. However, for now it is only important to realize that locally optimal sets are constructed relative to only their neighborhood. This allows a wide range of communities, with both high and low densities, to be discovered.

To motivate why this is important, consider the stylized example in Figure 2. This figure depicts some form of organized/coordinated ring-group which would intuitively pass as a community (for example, a committee of NSF-reviewers). Since we allow overlapping groups, a node could belong to multiple communities, as illustrated by the shaded areas. A node belongs simultaneously to this ring-community as well as to other communities. By virtue of belonging to those other communities, the node communicates extensively outside the ring-group (especially if the node belongs to many other communities). This means that the node displays *more extra-group* similarity than intra-group similarity with respect to the ring-group. There is no flaw with the intuition that a community should display intra-group similarity; the reason the extra-group similarity can be larger is because the communities can overlap. Note that the ring itself in our example, though it is *connected* and appears structured, is not particularly dense; in fact, if each member connects to δ external nodes, then $d(S) = 1/(\delta+1)$, which can be sufficiently small. Other communities may not have as low a density as this.

We can go further in claiming that this subset should be considered a community independent of the nature of the other communities in the network. Accepting the *locality* property of the communities suggests that the methods which define a global objective function (for example, modularity [16]) and optimize it to identify all the communities might fail to discover the ring-community. Such methods have found success in partitioning a network, but when overlap is allowed and essential, it is not even clear how to properly define global objective functions.

In the toy ring group shown in Figure 2, the density of our ring-community is $d(S) = 1/(\delta+1)$. One can easily verify that if we remove a node u from the group, its density drops to

$$d(S-u) = \frac{1}{\delta+1+\delta/(|S|-2)}. \tag{3}$$

Alternatively, suppose we try to add one of the neighboring nodes z to S. To illustrate, assume that this node has one connection into S and β connections to other nodes. In this case, adding z changes the density to

$$d(S+z) = \frac{1+1/|S|}{\delta+1+\beta/|S|}, \tag{4}$$

which is smaller than $d(S)$, when z has more connections to the outside world than the average for nodes already in S. This means that S is *locally optimal* with respect to single node moves. Thus, the requirement of local optimality can capture S as a community.

The main benefits of defining communities as locally optimal sets are that sets with vastly different structural properties can be locally optimal, with varying densities and that locally optimal communities can overlap. Not being able to improve a community (as measured by the density d) is intuitive; this does *not* require a high density or a specific structure of the community. The unified idea of the discussion is that a community is a *locally* defined object. A community in one part of the network should not rely on what is going on in another part of the network. Further, community structure can vary over the network – communication in some communities can be more intense than in others; their structures can be different.

3 Local Optimality Examined

The benefits of local optimality as a mechanism to discover overlapping communities have not been lost on researchers. However, despite general agreement that locally optimal sets of vertices form reasonable communities, there is a lack of consensus as to the specifics of the notion of local optimality. Further, additional issues which present themselves when identifying local communities are largely ignored. In this section, we begin by examining the notions of local optimality and density functions. Consolidating this discussion, the section is concluded with a set of axioms which we suggest to be the simplest, smallest set of criteria which any local, overlapping groups should satisfy.

3.1 Vertex Removals and Connectivity

As previously stated, various methods have been proposed which attempt to optimize local density functions to identify potentially overlapping communities. However, methods define optimality with respect to different processes. In the process of optimization, some methods allow vertices to be added and removed while others allow only additions. This results in two different notions of local optimality.

An additional problem, which exists with any algorithm allowing vertices to be removed during the optimization, involves the connectivity of communities. As shown by the adding condition, whether a vertex is added to a group or not is determined by the distribution of the the vertex's degree as well as the community's density at the time of consideration. This may cause a cut vertex, which was previously inserted into the set based on an earlier, lower density to be removed, thereby disconnecting the set. Producing a disconnected set of vertices in a grouping algorithm is clearly a problem and affects those local optimization algorithms provided by Baumes[2] and Lancichinetti [15].Clauset's algorithm in [4] successfully avoids this problem by only adding to the group during the optimization, and [24] only merges candidate groups, ensuring the connectivity of the resulting set.

Examining Figure 3, a graph is shown which demonstrates this problem. Consider a candidate group being optimized containing only vertex 1. Initially, the set's density is 0, as there are no internal edges. Upon iterating through all vertices in order of increasing degree, vertex 2 is added to the cluster. This results in an increase in density due to the addition of an internal edge. Proceeding to Figure 3(c), the group expands to contain the chains and triangles connected to vertex 2. At this point however, the density has increased such that the removal condition given above in (7) is now true. This will result in the removal of vertex 2 and the disconnection of the set. Vertex 1 will also be removed producing a locally optimal, disconnected set.

3.2 Tuning Parameters

Examining the previously defined density function in (2), we wish to determine the conditions by which a vertex is added or removed from the set. Consider the situation detailed in Figure 4. Here, some vertex i is being considered for addition into the set C. The vertex's degree k_i is split into α and β such that $\alpha = \sum_{j \in C} w_{i,j}$, $\beta = \sum_{j \notin C} w_{i,j}$, and $k_i = \alpha + \beta$. For the vertex i to be added to the set, the density of $C \cup \{i\}$ must be greater than the density of C alone. Therefore, we have

$$\frac{w_{in}}{w_{in} + w_{out}} < \frac{w_{in} + \alpha}{w_{in} + w_{out} + \beta}. \tag{5}$$

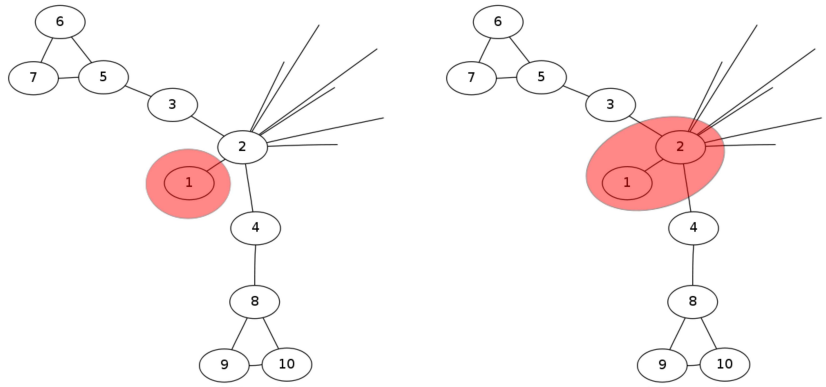
Which can be simplified to

$$\frac{\alpha}{\beta} > \frac{w_{in}}{w_{in} + w_{out}}. \tag{6}$$

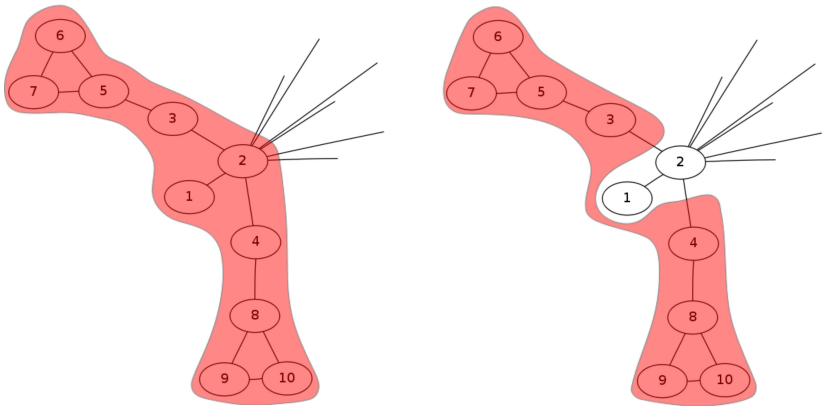
Performing a similar procedure for removals, we arrive at

$$\frac{\alpha}{\beta} < \frac{w_{in}}{w_{in} + w_{out}}. \tag{7}$$

It is clear to see from these two relations, that additions and removals occur relative to the density of the set at the time of consideration. It is worth examining



(a) $\mathrm{d}(C) = \frac{0}{1}$ (b) $\mathrm{d}(C) = \frac{1}{9}$



(c) $\mathrm{d}(C) = \frac{11}{17}$ (d) $\mathrm{d}(C) = \frac{8}{10}$

Fig. 3 A sample graph demonstrating the generation of a locally optimal, disconnected group. The density function being used for this examination is (1).

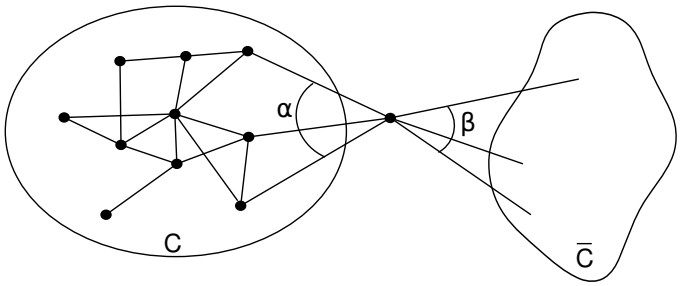


Fig. 4 The breakdown of α and β for the addition of a vertex to community C.

how this metric behaves when sparse areas of the graph are encountered. Consider a vertex with degree 2, adjacent to the set being optimized, where $\alpha = \beta = 1$. Since there is at least one edge cut by the community's boundary (implying a density < 1), vertices matching this description will always be added to the group. In practice, this results in groups with a large amount of edges forming a "core" and expanses of sparse vertices. This is a problem primarily when dealing with low degree graphs, or social networks whose degree distribution is scale free. This effect is shown in Figure 5. The d values show how density increases until the entire chain is contained within the set. For many applications, such a grouping would be inaccurate, since vertices on the left and right of the chain are very distant and can be presumed to be dissimilar.

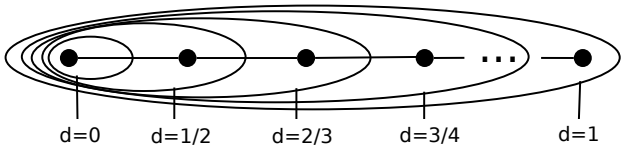


Fig. 5 A sample graph demonstrating the performance of local optimization on a chain of vertices.

It is unintuitive how a community detection algorithm should handle sparse chains of vertices. At one end of the spectrum, one could imagine each pair of vertices composing the most salient communities. However, there could also be an argument made that the entire chain should compose a group. This can be controlled by adding a parameter to the density function, introducing a penalty for additions which significantly reduce the edge probability of the community. The following density function is proposed

$$d(C) = \frac{w_{in}}{w_{in} + w_{out}} + \lambda e_p. \tag{8}$$

where e_p is the edge probability within the group C

$$e_p = \frac{\sum_{i,j \in C} e_{i,j}}{|C| \times (|C| - 1)} \tag{9}$$

and λ is a parameter allowing the results to be fine-tuned. Setting $\lambda = 0$ will produce the same results as (2), while larger values will increase the amount of significance the internal edge probability of the set has. This also has the advantage of producing smaller groups for larger values of λ which allows groups to be produced across a wide variety of resolutions. As suggested by Lancichinetti et al. in [15], this parameter could also be used to determine the significance of groups. Groups which are structurally significant could be likely to exist across numerous values of λ.

3.3 Local, Overlapping Axioms

Based on the above observations, as well as previous literature, a set of axioms can be described which any local, community detection method should aim to satisfy. We now state the minimum requirements of a community.

Connectedness. A community should induce a connected subgraph in the network. If the only way to get from one node to another in the community is via some external node, it suggests that the community is incomplete.
Local Optimality. According to an appropriate density metric $d()$, predefined on all subsets of nodes, the density of a community cannot be improved with the removal or addition of a single node.

Note, that the local optimality requirement, but not the connectivity requirement, was first introduced in [1, 2]. Examples can be easily developed of locally optimal sets that induce disconnected subgraphs. Our community axioms posit, in particular, that communities are identified "locally," within one-hop distance from the set. Specifically, we require local optimality with respect to the addition or removal of a single vertex. Previously proposed methods have suggested identifying locally optimal sets with respect to addition only. However, it can be argued that if a community can be improved relative to some density function via removal, it is less meaningful than one constructed via addition and removal. Additionally, one could suggest further notions of local optimality which are relative to a larger number of removals or additions. These other notions of optimality are left for future work. As we will see, these two axioms alone are sufficient for discovering communities which overlap and satisfy the intuitive properties we expect of a community.

It is important to note that this definition is quite different from many previous notions such as those of a "strong" or "weak" community suggested by Raddichi in [20] as well as the definition of modularity which was previously discussed. Rather, this definition focuses on a localized approach that eschews globally formulated null models and strict edge-based requirements.

Algorithmically, it is not easy to identify all communities satisfying these properties, and so we resort to a simple heuristic which we discuss next. Our goal is to show that the communities discovered using this heuristic identify salient communities in both common benchmark data as well as real, observed on-line associations.

4 Connected Iterative Scan

In [2], the authors describe a community detection algorithm, termed Iterative Scan. Here we describe a modification of IS to discover communities the previously identified axioms of optimality and connectedness.

Iterative Scan, IS, consists of repeated "scans" each starting with an initial set developed by the previous scan (a "seed-set" for the first iteration). It examines each node of the network once, adding or removing it if such an action increases

the current density of the set. The scans are repeated until the set is locally optimal with respect to a defined density metric. The choice of the seed-sets is not predetermined; they can be the nodes, or the edges of the network. A procedure for seeding, called LinkAggregate, is presented in [1]. LinkAggregate efficiently produces seed-sets that form a cover (with some overlap) of the entire vertex set. The nodes are evaluated by IS in the order of increasing node-degree, from low to high degree. Iterative scan in this form had been used for a variety of interesting applications such as modeling dynamic networks [10]. A similar method, implementing the idea of the greedy local optimization (as a replacement of a scan in IS) was later given in [15]. For every iteration, the algorithm examines all vertices in order to find the one which causes the maximum increase of the density. That vertex is used to update the current set and any density improving removals are then performed.

The density metric itself can be defined in a number of ways; our analysis uses a modification of the standard density function in Equation 2. Rather than using w_{in}, recent literature [15] has proposed using using the internal and external degree of all vertices in the group rather than the number of edges. This is a slight modification, resulting in the the use of $2 * w_{in}$ in place of w_{in}. For the sake of comparison to previous work, we will optimize using this density function. Our experiments show that in many social networks, there is a very large set of potential communities, *i.e.*, sets that satisfy the two axioms above. Thus, filtering of candidate sets is often necessary and should be done as dictated by the specifics of the application in which community structure is useful. One possibility is to order the candidates by $d(S)$, and consider as most "interesting" those communities which had more internal than external communication ($d(S) > \frac{1}{3}$). This filter is consistent with the notion of a "weak" community as defined by Raddicchi *et al* in [20] and is done in this work to restrict the scope of the analysis for computational reasons. Note that when overlap is allowed, this additional requirement might not be satisfied by all communities. The other possibility of filtering is to look at the communities for which $d(S) < \frac{1}{3}$, as these communities are still connected and locally optimal, even though their members communicate outside of the community a significant fraction of time, which results in sparse internal communication.

To ensure the connectivity of the identified communities, we modify IS and term the resulting algorithm Connected Iterative Scan, CIS. Psuedocode for this algorithm is presented in Algorithm 1. As is the case with IS, CIS consists of a number of scans that are repeated for each current set until no change of the set occurs. The set is then declared to be a community. Every scan proceeds through the nodes in the order of increasing node degree. Once a scan is finished, the set's connectivity is examined. If the set consists of multiple connected components, it is replaced by the connected component with the highest density, after which the next scan starts. Note that selecting only the highest density component effectively sidesteps the issue of repeatedly optimizing to the same, disconnected cluster. The specific selection of this rule for identifying connected, locally optimal sets was motivated by the desire to generate as many groups as possible. The running time of the algorithm however, suffers from repetitive connectivity evaluations. For applications where running time is important, one can simply discard those sets which are not

Algorithm 1 Connected Iterative Scan

Require: $G = (V, E), S \neq \emptyset$
Ensure: density$(S) \geq$density$(S \cup \{v\})$ & density$(S) \geq$density$(S \backslash \{v\}), \forall v \in V$
  improved $\leftarrow$ **true**
  while improved == **true do**
    improved $\leftarrow$ **false**
    for all $v \in V$ **do**
      if $v \in S$ **then**
        if density$(S \backslash \{v\}) >$density(S) **then**
          $S \leftarrow S \backslash \{v\}$
          improved $\leftarrow$ **true**
        end if
      else
        if density$(S \cup \{v\}) >$density(S) **then**
          $S \leftarrow S \cup \{v\}$
          improved $\leftarrow$ **true**
        end if
      end if
    end for
    $S \leftarrow$maxComponent(S)
  end while

connected as a additional post-processing step. Finally, the seeding is this text is done using placing each vertex in its own initial seed community.

The disadvantage of CIS is the same as that of IS; both methods may produce a large number of highly overlapping communities. However, this problem can be managed by effective post-processing of results and merging of highly similar communities. Sample results of CIS for a community analysis of Zachary's Karate Club data set [25] are given in Figure 6. This network represents a set of friendships with in a collegiate martial arts club. Performing analysis on the data, which was collected while the group was undergoing a fissure, provides interesting insight into the set of individuals for whom selecting which splinter group to join was not a trivial choice. Using CIS, these individuals exist in the overlap between the two larger groups in the network. These groups are clearly salient and similar results are found across a variety of literature in community detection.

The complexity of CIS is difficult to analyze due to its dependence on the number and quality of the seeds being optimized as well as the underlying graph structure. However, similar optimization techniques have previously [15, 1] been empirically shown to have a running time on the order of $O(n^2)$. For many graphs, running time can likely be reduced by introducing higher quality seeds, utilizing a simpler density function, or simply throwing out locally optimal, disconnected sets rather than checking for connectivity at each iteration. Additionally, since the optimization process is independent for each seed, the algorithm is highly parallelizable.

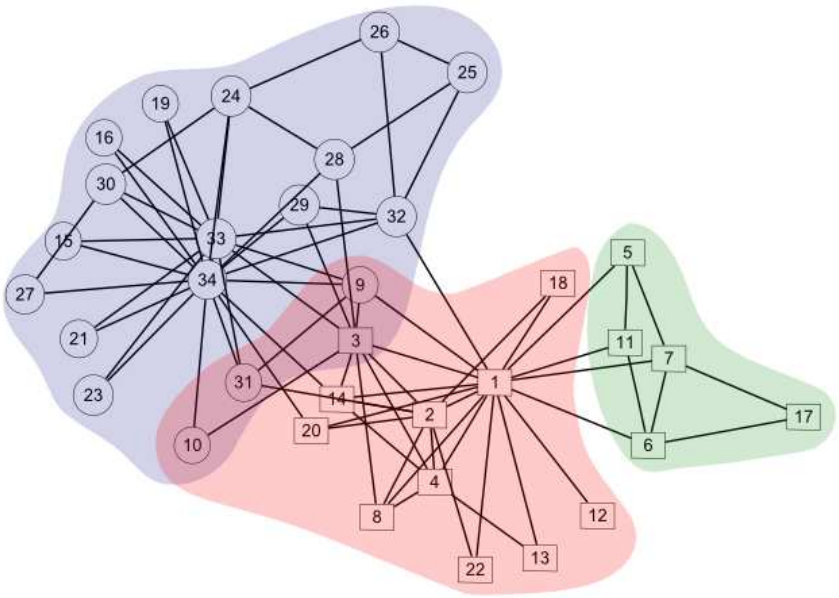


Fig. 6 Overlapping groups found in Zachary's Karate Club dataset. Different shapes identify the eventual group division. Groups were ordered to correspond to the number of distinct seeds which produced them. Groups were then selected until the graph was covered. Additional examination of groups which are produced by fewer seeds offers insight into potentially overlapping subgroups of the primary groups presented here.

4.1 Benchmark Performance

Quantifying the performance of the algorithm is difficult due to the approach. Namely, few other methods aim to produce a large set of locally optimal groups. Rather, they tend to focus on finding partitionings or covers which best express the data. In addition, methods which allow for overlap tend to be insufficient due to the unsatisfied community axioms. In this section, numerous benchmarks will be examined. First, a method to compare two sets of overlapping groups will be presented. Next, a small, toy graph with uniform degree proposed by Girvan and Newman will be considered. Then, random scale free networks with embedded community structure will be explored for the non-overlapping case. Each of these experiments will be evaluated via the Normalized Mutual Information measure originally proposed in [15].

4.1.1 GN Benchmark

One of the first benchmarks proposed for community detection algorithms was proposed by Girvan and Newman in [9]. This benchmark dataset, consists of 128 vertices divided into four groups of 32. Each vertex has a degree of 16. The strength of the community associations are given by a mixing parameter which indicates the probability that an edge is placed between two communities rather than internal to a single community. Specifically, this mixing parameter is given by

$$\mu_k = \frac{k_o}{k_i + k_o} \tag{10}$$

where k_o is the number of edges connecting a vertex to a vertex in another community and k_i are the number of edges connecting a vertex to other vertices within a community. It should be explicitly noted that this benchmark assigns each vertex to exactly one community during network generation. Despite this, it is important that methods which identify non-disjoint communities be capable of producing accurate communities even when the underlying structures are disjoint.

For Connected Iterative Scan, the results are given in Figure 7. Each point represents the average normalized mutual information over 25 graphs with a given mixing parameter. Seeds are generated by placing each vertex in a candidate cluster. The results shown are a reflection of what is considered to be the "base" settings of the algorithm. This configuration is the density function previously described in the text, vertices ordered by increasing degree, and seeding done by placing each vertex into a seed group by itself.

The two curves in Figure 7 show the result of taking all locally optimal sets discovered by the algorithm as well as using some domain knowledge to filter out the four most frequently discovered sets. It should be noted in the results that the curve is similar to those produced via other methods, though with slightly less accuracy for networks with well defined group structure.

4.1.2 LFR Disjoint Benchmark

A more realistic set of benchmark graphs can be found using the LFR benchmark. Here, a scale free graph is generated with communities of varying sizes. This benchmark was first used in [14] to compare various methods of community detection on a more complex network than the GN benchmark. For the experiments contained within this text, graphs are generated matching a power-law degree distribution with $\alpha_d = 2$ and a power-law community size distribution with $\alpha_c = 1$. For all networks, the average degree of each vertex is 20 and the max degree 50. Community sizes are limited to 10-50 for runs marked "S" and 20-100 for runs marked "B". The output of CIS is processed for evaluation by removing duplicate communities and removing those communities which contain the entire graph. Each data-point represents the average of 25 trials.

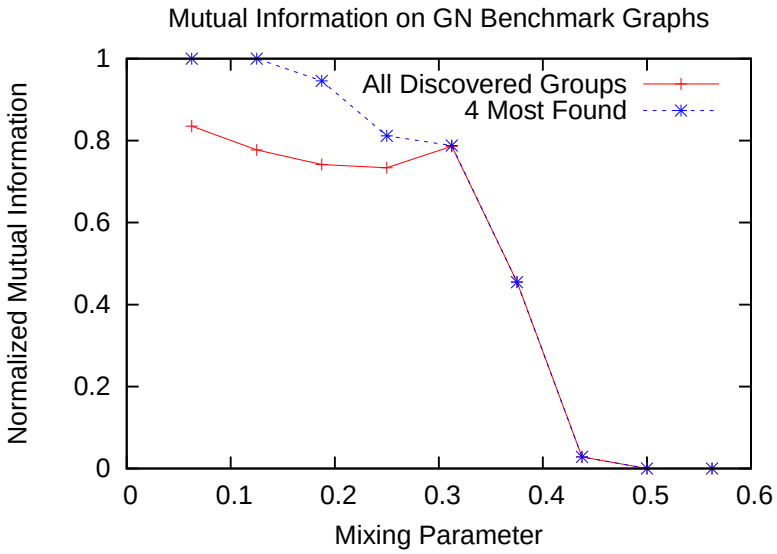
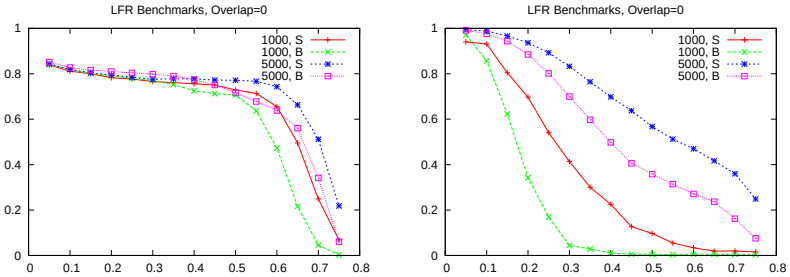


Fig. 7 Normalized mutual information for Connected Iterative Scan on GN benchmark graphs

The results of this analysis using CIS and CPM are given in Figure 8. Figure 8(a) clearly shows the limitations of identifying a specific structure when compared to Figures 8(b)-8(d). Identifying overlapping cliques is much less accurate as group size increases. While CPM produces better results for networks with well defined, small communities, Connected Iterative Scan produces better results in networks with larger community sizes as well as those networks with less well defined communities. The quality of the communities produced via CIS are comparatively stable in the face of changing community and graph properties.

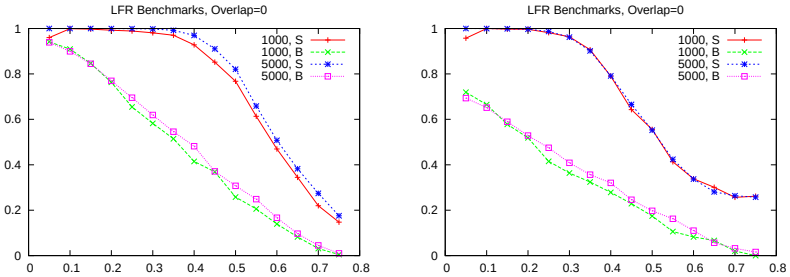
4.1.3 LFR Overlapping Benchmark

The LFR benchmark software also allows groups to be embedded such that a given portion of individuals exist in a specified number of groups. This allows algorithms to be compared on networks with known community overlap. Taking the same degree and community size distributions as the previous set of experiments, Connected Iterative Scan and CPM can be compared at varying levels of overlap. Figures 9 and 10 detail the results of this comparison for 10% and 30% of the vertices existing in 2 communities. Again, the same general trend exists; identifying communities by looking for a set of rigid structural traits fails to identify larger embedded communities, while those produced by CIS are discovered with the same accuracy regardless of community composition.



(a) Normalized mutual information for Connected Iterative Scan on LFR benchmark graphs

(b) Normalized mutual information for CFinder on LFR benchmark graphs with k=3



(c) Normalized mutual information for CFinder on LFR benchmark graphs with k=4

(d) Normalized mutual information for CFinder on LFR benchmark graphs with k=5

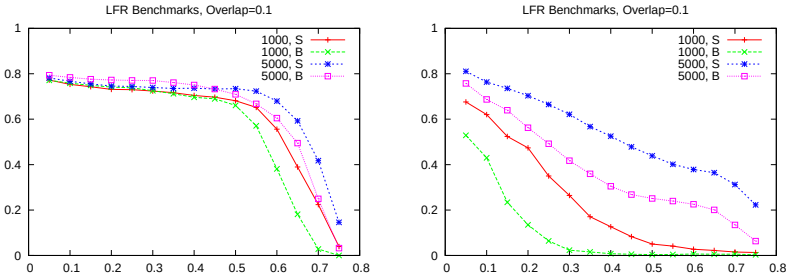
Fig. 8 Connected Iterative Scan vs CFinder for LFR benchmark graphs with disjoint embedded communities

4.2 λ value

Intuitively, inclusion of the internal edge probability in the density function for Connected Iterative Scan allows the algorithm to be tuned to discover different types of communities. It introduces a criteria for addition different from what was initially proposed during the development of Iterative Scan. When $\lambda > 0$, the vertex being considered for addition must strike a balance between the change in the original density value and the change in edge probability.

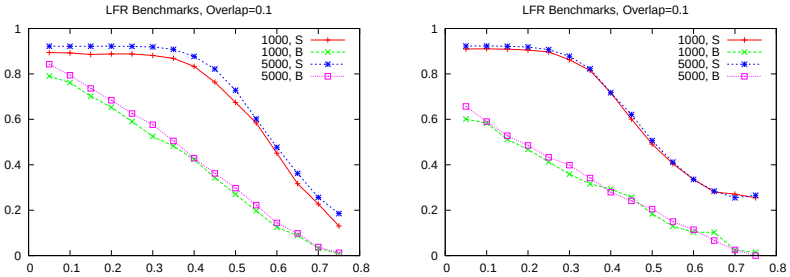
This effect can be seen in real networks as well. In this analysis we consider a network in which vertices represent football teams affiliated with universities within the United States. Typically, teams are members of conferences, within which they play a significant portion of their games. Edges in the network indicate that two teams played each other. Groupings produced by Connected Iterative Scan can be compared to the natural divisions created by conferences.

Groupings were performed using a number of different values of λ and filtering the communities by taking only the most discovered groups. The normalized mutual information between the true grouping and the discovered grouping are plotted



(a) Normalized mutual information for Connected Iterative Scan on LFR benchmark graphs

(b) Normalized mutual information for CFinder on LFR benchmark graphs with k=3



(c) Normalized mutual information for CFinder on LFR benchmark graphs with k=4

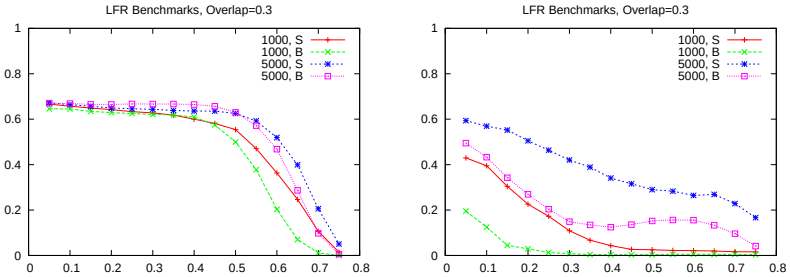
(d) Normalized mutual information for CFinder on LFR benchmark graphs with k=5

Fig. 9 Connected Iterative Scan vs CFinder for LFR benchmark graphs with overlapping embedded communities where 10% of the vertices associate with 2 communities

in Figure 11. The peak at $\lambda = 0.125$ indicates the grouping which most closely matches the underlying conference structure of the network. Qualitatively, the difference between $\lambda = 0$ and $\lambda = 0.125$ is an increased focus on small, tight-knit cores.

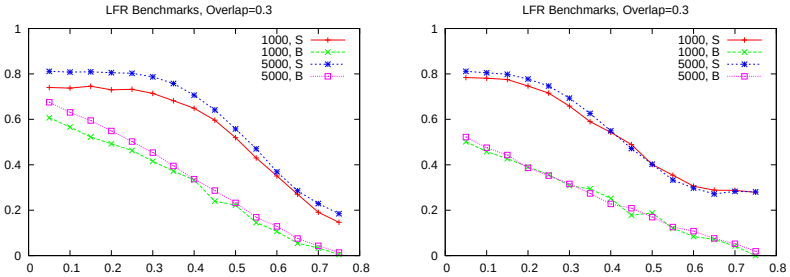
5 Significance of Overlap

In order to demonstrate that group overlap is a significant feature of some social networks, it is important first to consider the features which pairs of groups should have to indicate that the overlap between them is significant. Consider the overlapping groups presented in Figure 12. Here group *A* consists of white and grey vertices, and group *B* consists of the the black and grey vertices. By this definition, individuals represented by vertices colored grey are members of both group *A* and *B*.



(a) Normalized mutual information for Connected Iterative Scan on LFR benchmark graphs

(b) Normalized mutual information for CFinder on LFR benchmark graphs with k=3



(c) Normalized mutual information for CFinder on LFR benchmark graphs with k=4

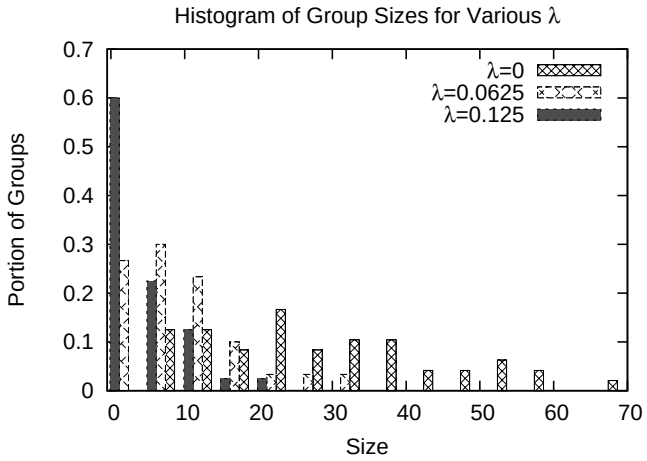
(d) Normalized mutual information for CFinder on LFR benchmark graphs with k=5

Fig. 10 Connected Iterative Scan vs CFinder for LFR benchmark graphs where 30% of the vertices associate with 2 communities

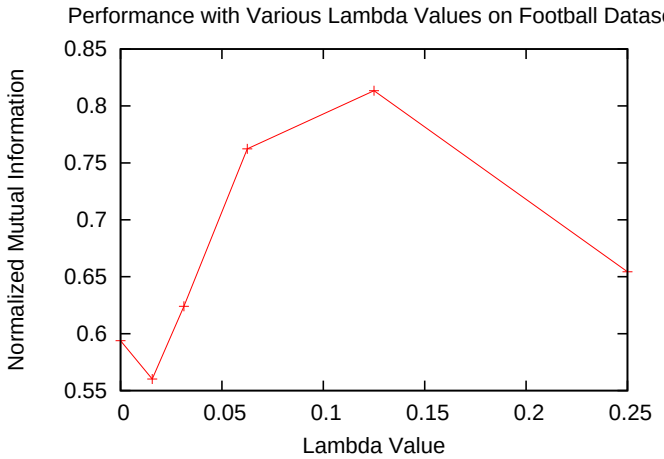
For a pair of overlapping groups to have *significant* overlap, and thus be considered a *non-separable pair*, the groups and their overlap must fit certain criteria. In general, each criterion serves to identify quality of overlapping groups that cannot be expressed via a single group (the union), or two, or three partitions. These criteria can be described conceptually as follows.

5.1 Structural Significance

The existence of overlap between a pair of groups should enhance the "quality" of each of the groups individually. For example, if the quality of each group is measured by the ratio of edges internal to the group to those which are cut by the boundary of the group, removing $A \cap B$ from A and B in the groups expressed in Figure 12 would result in a decrease in the quality of each group. The two vertices in the intersection $A \cap B$ have the same degree within each group as they have external to each group. Thus, relative to the previous quality metric, the vertices should be a



(a) Size distribution for groups produced with various values of λ on the college football dataset.



(b) Plot showing the peak in NMI between the discovered groups and the ground truth and λ 0.125

Fig. 11 Performance of various λ values on the college football dataset

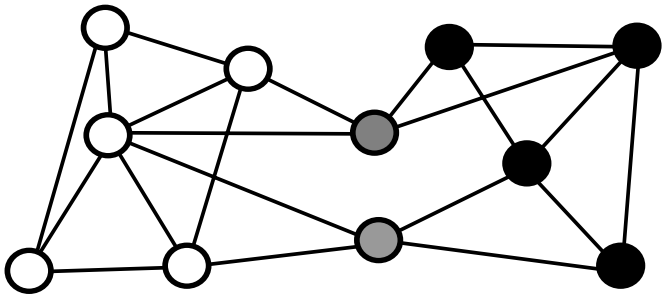


Fig. 12 An example of a pair of groups that overlap. The overlap is identified by the grey vertices while individuals in only one group are colored black or white depending on the group of which they are a member.

part of each group since they increase the numerator while holding the denominator constant. Therefore, the overlap is the key to the structural significance of both groups in Figure 12.

5.2 Group Validity

It is also important that each group be somehow verifiable using a reasonable method relative to the input data. Ideally, using some underlying traits of the individuals in the network being analyzed, groups should have higher trait similarity between members than one would expect if membership in groups were determined at random. Examples of this type of validation have been used in various previous literature, using age and location as traits of the individuals [18]. Group validity is essential in filtering out groups that are products of random structures in the underlying communication graph, and serves to ensure that the group detection is accurate.

5.3 Overlap Validity

Using the same notion of trait similarity, the individuals within the overlap must have some similarity with the remainder of each group of which they are a member. In Figure 12, the graph is divided into three groups, $A-B$, $B-A$, and $A \cap B$ (white, black, and grey respectively). For overlap to be important, $A-B$ and $A \cap B$ must be similar, $B-A$ and $A \cap B$ must be similar, and $A-B$ and $B-A$ must be dissimilar, relative to certain significant traits in the data, that is individuals in the overlap need to be clearly similar to the remainder of either group. However, it is necessary that the remaining individuals in each group be dissimilar to those in the other group. If this dissimilarity does not exist, the overlapping pair can be captured in a single partition and overlap is not necessary to explain the relationships in the data.

Pairs of groups that satisfy each of these criteria are fundamentally sound communities due to their structural significance and their group validity. Conceptually, the existence of overlap validity restricts how the individuals can be placed in a partitioning. If all users of the three groups are placed in a single partition, dissimilar vertices in $A-B$ and $B-A$ are associated. If the vertices are placed in three partitions according to color, a strong association between $A \cap B$ and both $A-B$ and $B-A$ is missed. The vertices may be placed in a pair of disjoint groups only if the similarity between $A \cap B$ and both $A-B$ and $B-A$ is highly unbalanced. If the two similarities are comparable, however, one does not have justification to place the users in one group or the other. A detailed description of each of these cases is given further in the text. Significant numbers of non-separable pairs indicate that overlap is an essential component of communities within the network.

5.4 Measures

It becomes necessary to formulate a set of methodologies to indicate whether the notions of group validity and overlap validity are satisfied for a given community or pair of communities. We begin by identifying the set of data used in the analysis.

Due to the implementation of the Friend Feed provided by LiveJournal, friendship declarations can serve as an indicator of interest. By declaring a friendship, the declaring user is notified whenever his or her friend makes a post. It can be assumed that individuals which attract a large number of these friend declarations are highly important to the discourse on some set of topics. Thus, friendship declarations serve as a proxy for some set of declared interests from each user. In this analysis, an individual is defined as influential if he or she has a friendship in-degree of 300 or more. This criteria marks approximately 4,800 bloggers as influential.

The selection of a subset of the friendship relations was done for purely computational reasons, cutting the set of possible friend relations from 500,000 to 5,000. Additionally, interest declarations could be used as validation data. However, within LiveJournal, this data is entered via comma separated values, resulting in a much larger set of possible declarations. Additionally, the popular declared interests, such as ”books”, ”movies”, or ”music”, are much more universal than the most popular friendships. Further, words typed with spelling errors, abbreviations, slang, and the use of synonyms can all be indicative of the same set of topics. The friendship relationship is used in this situation because of its concreteness.

Now, given that each vertex i has a set of declared friendships F_i, we can describe our validation measures. The group validity requirement claims that there should be more similarity within the group than one would find at random. To measure this, we define the notion of *internal pairwise similarity* (denoted *IPS*). For a given community C, the internal pairwise similarity can be computed as

$$IPS(C) = \frac{\sum_{i \in C} \sum_{j \in C, j \neq i} J(F_i, F_j)}{|C|^2 - |C|} \tag{11}$$

where $J(F_i, F_j)$ is the Jaccard index [12] between the two sets. This value can be expressed as

$$J(F_i, F_j) = \frac{|F_i \cap F_j|}{|F_i \cup F_j|} \tag{12}$$

The value $J(F_i, F_j)$ will be maximized ($J(F_i, F_j) = 1$) if the sets F_i and F_j are identical and will be minimized ($J(F_i, F_j) = 0$) if the two sets are disjoint. Intermediate values of $J(F_i, F_j)$ indicate shared friendships and is normalized by the number of possible shared friendships between the two individuals. Thus, the *IPS* value measures the average similarity between the friendship declarations of pairs within the community. This value is utilized in place of Normalized Mutual Information discussed earlier due to the fact that the "ground truth" in this situation is unknown.

Revisiting the notion of overlap validity, it becomes apparent that a method comparing sets of friendship declarations are needed. Given a pair of overlapping communities A and B, three friendship declaration vectors can be computed. These vectors, denoted L_{A-B}, L_{B-A}, and $L_{A \cap B}$, give the probability that a vertex within each set indicated by the subscript will declare a given individual in the popular friend set as a friend. Formally, $L^i_{A \cap B}$ can be defined for each of the elements of $L_{A \cap B}$ as

$$L^i_{A \cap B} = \frac{|\{x | x \in A \cap B, i \in F_x\}|}{|A \cap B|} \tag{13}$$

where F_x is the set of friends declared for vertex x. Similar vectors can be defined for L_{A-B} and L_{B-A}.

Once these vectors are constructed, the similarity between each of them can be calculated via the *cosine* similarity. Formally, this can be given, relative to two equal dimension vectors X and Y, as

$$\cos(\theta_{X,Y}) = \frac{X \cdot Y}{\|X\| \|Y\|} \tag{14}$$

A low value of $cos(\theta_{X,Y})$ indicates that the vectors X and Y are close to orthogonal. High values indicate that the vectors have similar values across many dimensions.

Given the three friendship declaration vectors described previously, the *cosine* similarity between them can give an indication as to whether or not the overlapping group satisfies the overlap validity requirement. Namely, that the inter-group similarity $cos(\theta_{L_{A-B}, L_{B-A}})$ be less than the intra-group similarities $cos(\theta_{L_{A-B}, L_{A \cap B}})$ and $cos(\theta_{L_{B-A}, L_{A \cap B}})$.

In order to simplify this notion, the intra-group and inter-group similarities can be combined into a single statistic representing the relative similarity between the three sets. For the sake of notation, let the inter-group similarity $cos(\theta_{L_{A-B}, L_{B-A}})$ be given by the variable *inter* and let each of the intra-group similarities $cos(\theta_{LA-B, A \cap B})$

and $cos(\theta_{L_{B-A,A\cap B}})$ be given by $intra_A$ and $intra_B$ respectively. These values can be combined into a measure of overlap validity as

$$\mathrm{OV}(A,B) = \frac{intra_A + intra_B}{2} - inter \tag{15}$$

for values of $\mathrm{OV}(A,B) > 0$, the intersection is more similar to each group than the remainder of each group is with each other, indicating that the overlap is split in its association with each set.

5.5 Results on LiveJournal

We applied the Connected Iterative Scan algorithm, CIS, to the LiveJournal dataset to produce a set of communities which satisfy the axioms. We also partitioned this graph using the algorithm CNM designed by Clauset, Newman, and Moore ([5]) to give the reader a point of reference and to demonstrate the difference in community sets produced by the two methods. Statistics demonstrating the number of groups, average size, average density, modularity (Q, only applicable for the partitioning), and the number of vertices which are placed in at least one community are given in Table 1.

Statistics of Groups Found via CNM and CIS

	Groups	AvSize	AvDens	Q	Cov
CNM	264	1190	0.745	0.485	100%
CIS	14903	168.8	0.455	–	47.5%

Table 1 Statistics of groups from CNM and CIS. Q shows the modularity value of the grouping generated by CNM and "Cov" indicates the portion of vertices which are in at least one group.

The partitioning produces a small number of sets across a wide variety of sizes, while the overlapping group detection produces a much larger number of smaller groups which do not cover the entire graph. Coverage is not a requirement; it is not necessary for every node to belong to a cluster. Rather, we are interested in finding those groups which naturally overlap and studying the significance of this overlap.

If the overlapping groups detected fit the requirement of having structural significance, removal of a pair's overlap will produce a decrease in group quality, as measured by the density d. Overlapping groups are more compelling when the overlap is structurally necessary for each group. After filtering out subset inclusion (a trivial form of overlap), the remaining overlapping groups display a high degree of structural significance for the overlap. Specifically, for 80.8% of the overlapping pairs, both groups in the pair experience a decrease in density if the intersection is removed. Figure 13 shows more details of the exact distribution of changes in density when the overlap is removed. Even though we observed that some groups

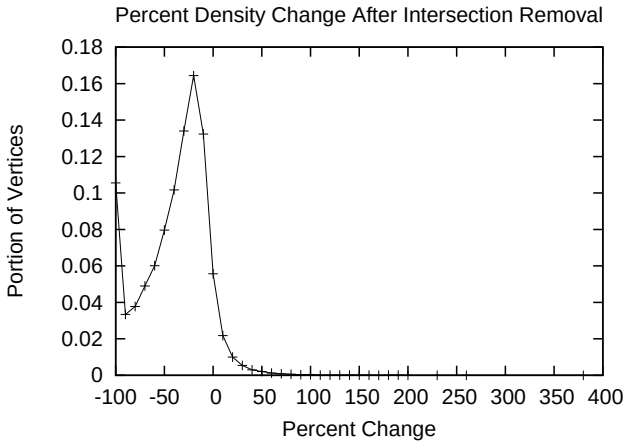


Fig. 13 Portion of clusters that experience a given percentage change in density when the intersection of an overlapping pair is removed. Portions are collected in bins of size 10%. This plot contains 50 data points.

are improved by the removal of intersection, the overwhelming majority of groups experience a significant decrease in density. We conclude that the overlap is structurally significant.

We now investigate the validity of the groups found, with respect to user traits. Figure 14(a) shows the average internal pairwise similarity between users within a community as well as the average similarity between users in connected random groups as a function of size. The figure shows that groups produced by CIS have much larger amounts of similarity between users than the random case for sizes greater than 10. This value appears lower than random for sizes less than 10 due to the number of groups which have undefined friendship declarations. The portion of these groups discovered by CIS and at random are given in Figure 14(b). Figure 14(c) shows the same information as Figure 14(a) but with these undefined friendships removed.

Figure 15 shows the overlap validity measure over pairs of groups with a given overlap. This value is compared with the overlap validity measure for randomly selected groups with the same size and overlap. The x-axis denotes the overlap of the pair, where overlap is defined as the Jaccard index of the two sets. Clearly, there is a larger difference in similarity between the groups identified via CIS and those generated at random.

For the 14,903 unique groups that were discovered, 6,373(30%) of them overlap with at least one other group such that the pair can be considered justified by the three conditions previously described. These pairs are composed of 125740 unique users, a very significant portion of the graph.

Further, a significant portion of the non-separable groups have comparable intra-group similarity between the intersection $A \cap B$ and both of the sets $B - A$ and $A - B$.

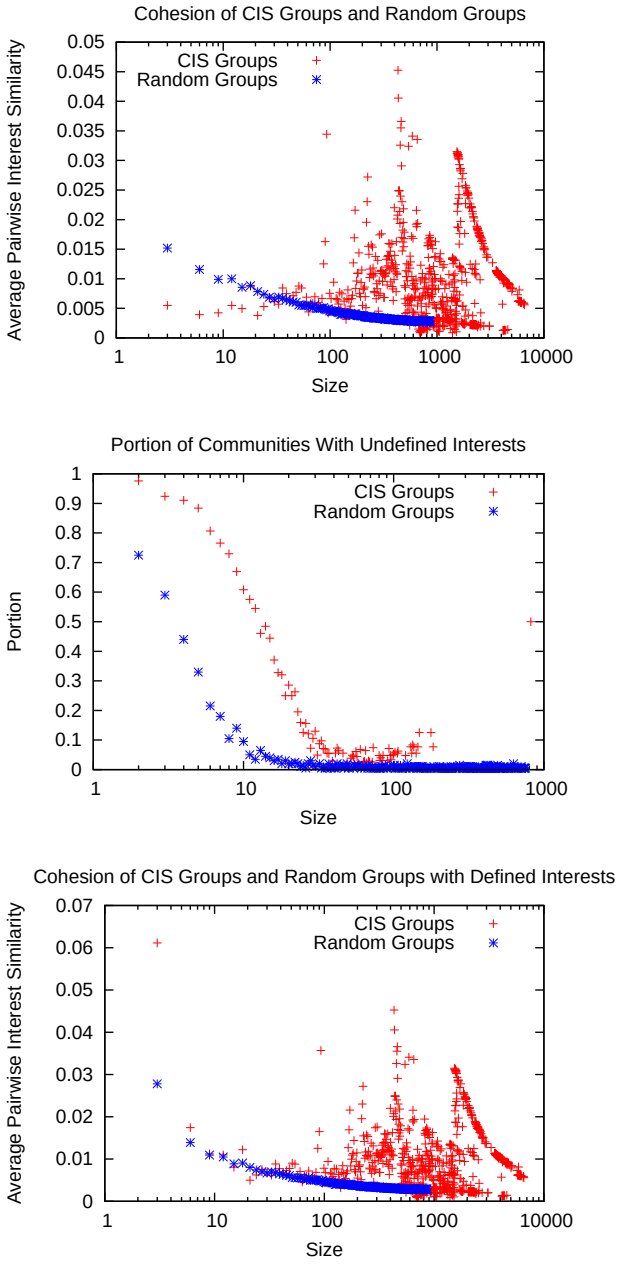


Fig. 14 Plot showing the average pairwise Jaccard Index of vertex friendships for all pairs within discovered communities of the same size and values found in randomly generated connected groups of the same size. The plot indicates that there is more similarity in a majority of the discovered groups than one would expect at random.

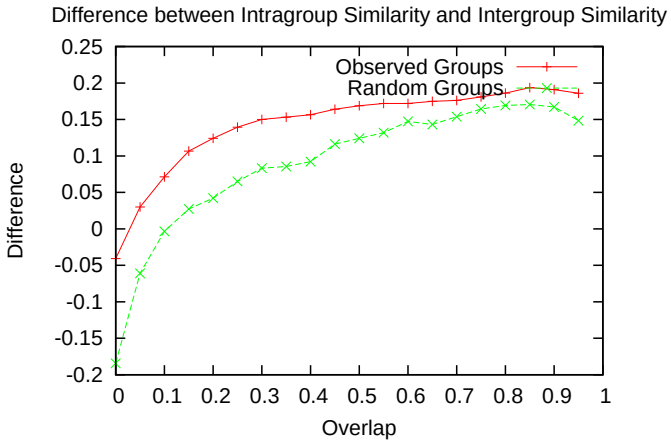


Fig. 15 Curves showing the average overlap validity measure $\mathrm{OV}(A,B)$ for identified, non-subset overlapping pairs and random groups of the same size and overlap.

If the similarities are considered comparable when they are within 5% of each other, 3,544 of the non-separable pairs have an overlap that is associated equally with the remainder of each group. These groups consist of 100,000 unique users. The existence of these groups is particularly significant in justifying overlap between communities. They clearly show that many sets of users are equally associated with distinct groups. Using a partition-based method for the detection of communities would either merge the entire pair into one group, failing to recognize the dissimilarity between the vertices in sets $A-B$ and $B-A$, or place the intersection with $A-B$ or $B-A$, missing the connection between the intersection and the other set.

6 Summary

Detecting communities in networks is a highly useful, highly non-trivial task. In certain domains, it is reasonable to expect that community structure overlaps. This necessitates defining the fundamental notions of what overlapping communities should look like. The axioms laid out in this chapter attempt to fulfill that need, while at the same time being as minimal as possible to allow for methodological and application specific variations.

Additionally, this chapter has shown that having a loosely defined definition of community structure is often a better choice compared to more restrictive methods which attempt to discover very specific structural formations in networks. The ability of a method and definition to produce quality communities across a wide array of network types is quite important. The axioms laid out in this text provide

a framework for such methods to be proposed within. We have also shown that in some networks, the best set of communities will only be found via some additional parameter tuning, particularly those parameters that relate to the size of the groups discovered.

Previous attempts at developing algorithms for the detection of overlapping communities have been primarily intuitive, and were developed without first examining to what degree overlap occurs in naturally occurring networks. A large amount of justified overlap indicates that the added complexity of new methods is essential to capturing all relationships expressed in the data. As a test network, we examined a social network composed of communications in a popular blogging service.

The overlap between groups must satisfy certain criteria to be considered significant. First, the inclusion of the common region in either group should enhance the quality of the groups by some metric. In addition, the groups themselves should be verifiable as significant through the use of a set of relevant user traits. Finally, the similarity between components of both groups involved in the overlap must be such that the intersection is more similar with the remainder of each group than the remainder of the groups are with each other. If each of these criteria is satisfied, placing the members of the group in some partitioning will not capture the subtle associations present in the data.

7 Future Directions

The use of overlapping community structure has significant potential to aid in the comprehension of underlying processes in an increasingly interconnected world. Intuition and the empirical observations contained in this chapter suggest that the associations contained within such communities capture essential and meaningful relationships which are implicit in the data. The field is far from mature, and various questions have arisen throughout research which remain open problems.

Community detection algorithms have tended to focus on static networks. However, real world data has the potential to be quite dynamic. As a result, new methods will need to be proposed to handle network ties with a temporal component. One simple extension to the work described in this text would be a sociologically grounded edge weight function. Such a function would take the age of a network association into account and decrease edge weight accordingly. The introduction of edge decay creates a potentially interesting area of study involving repetative reoptimization of sets over time.

An additional open area is the identification of additional methods of validating and quantifying the correctness of community detection methods. Recent work has introduced new methods to compare sets of overlapping sets [15], however, more fundamental analysis techniques should be used for comparison. Additional validation techniques such as computing feature similarity of identified groups require data sets with additional, frequently self-reported, information. The problems which exist with self-reported information can clearly be seen in the lack of networks with

a well defined, overlapping "ground truth". Often, overlapping communities tend to be more subtle than their disjoint counterparts. As such, it is difficult for individuals to list each of the groups with which they associate, as such groups may be ill defined in the minds of their members.

Another open problem is identifying a method or measure to determine the significance of a community among the set of those which have been discovered. As previously stated, using the minimal axioms described above, there are a vast number of sets which can be considered groups. In order for this type of analysis to be useful as a feature to some other mechanism, it is likely that the "best" groups with regard to application specific metrics will prove to be more useful than others. Significance measures have previously been explored somewhat with regards to disjoint community detection [13], but with the exception of a brief comment in [15], this discussion has largely been absent when examining the detection of overlapping communities.

References

1. J. Baumes, M. Goldberg, and M. Magdon-ismail. Efficient identification of overlapping communities. In *In IEEE International Conference on Intelligence and Security Informatics (ISI*, pages 27–36, 2005.
2. J. Baumes, M. K. Goldberg, M. S. Krishnamoorthy, M. Magdon-Ismail, and N. Preston. Finding communities by clustering a graph into overlapping subgraphs, 2005.
3. V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, 2008.
4. A. Clauset. Finding local community structure in networks. *Physical Review E (Statistical, Nonlinear, and Soft Matter Physics)*, 72(2):026132, 2005.
5. A. Clauset, C. Moore, and M. E. J. Newman. Finding community structure in very large networks. *Physical Review E*, 70(6):066111, 2004.
6. G. B. Davis and K. M. Carley. Clearing the fog: Fuzzy, overlapping groups for social networks. *Social Networks*, 30(3):201 – 212, 2008.
7. J. Duch and A. Arenas. Community detection in complex networks using extremal optimization. *Physical Review E*, 72:027104, 2005.
8. S. Fortunato. Community detection in graphs. *Physics Reports*, 486(3-5):75 – 174, 2010.
9. M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proc Natl Acad Sci USA*, 99(12):7821–6, 2002.
10. M. Goldberg, S. Kelley, M. Magdon-Ismail, K. Mertsalov, and W. A. Wallace. Communication dynamics of blog networks. In *The 2nd SNA-KDD Workshop '08 (SNA-KDD'08)*, August 2008.
11. R. Guimerà, M. Sales-Pardo, and L. A. N. Amaral. Modularity from fluctuations in random graphs and complex networks. *Phys. Rev. E*, 70(2):025101, 2004.
12. P. Jaccard. Étude comparative de la distribution florale dans une portion des alpes et des jura. *Bulletin del la Société Vaudoise des Sciences Naturelles*, 37:547–579, 1901.
13. B. Karrer, E. Levina, and M. E. J. Newman. Robustness of community structure in networks. *Physical Review E*, 77(4):046119+, Sep 2007.
14. A. Lancichinetti and S. Fortunato. Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *Phys. Rev. E*, 80(1):016118, Jul 2009.

15. A. Lancichinetti, S. Fortunato, and J. Kertesz. Detecting the overlapping and hierarchical community structure of complex networks. *New Journal of Physics*, 11, 2009.
16. M. E. Newman. Modularity and community structure in networks. *Proc. Natl. Acad. Sci. USA*, 103(23):8577–8582, 2006.
17. V. Nicosia, G. Mangioni, V. Carchiolo, and M. Malgeri. Extending the definition of modularity to directed graphs with overlapping communities. *J.STAT.MECH.*, page P03024, 2009.
18. G. Palla, A.-L. Barabasi, and T. Vicsek. Quantifying social group evolution. *Nature*, 446(7136):664–667, 2007.
19. G. Palla, I. Derenyi, I. Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435:814, 2005.
20. F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi. Defining and identifying communities in networks. *Proceedings of the National Academy of Sciences of the United States of America*, 101(9):2658–2663, 2004.
21. U. N. Raghavan, R. Albert, and S. Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Phys. Rev. E*, 76(3):036106, Sep 2007.
22. J. Reichardt and S. Bornholdt. Statistical mechanics of community detection. *Phys. Rev. E*, 74(1):016110, Jul 2006.
23. M. Rosvall and C. T. Bergstrom. Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences*, 105(4):1118–1123, 2008.
24. H. Shen, X. Cheng, K. Cai, and M.-B. Hu. Detect overlapping and hierarchical community structure in networks. *Physica A: Statistical Mechanics and its Applications*, 388(8):1706 – 1712, 2009.
25. W. Zachary. An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33:452–473, 1977.
26. S. Zhang, R.-S. Wang, and X.-S. 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.