2019

# Analyzing overlapping communities in networks using link communities

Olivia Hurd

# Analyzing overlapping communities in networks using link communities

By: Olivia Hurd

Research mentor: Professor James Bagrow

## Abstract

One way to analyze the structure of a network is to identify its communities, groups of related nodes that are more likely to connect to one another than to nodes outside the community. Commonly used algorithms for obtaining a network's communities rely on clustering of the network's nodes into a community structure that maximizes an appropriate objective function. However, defining communities as a partition of a network's nodes, and thus stipulating that each node belongs to exactly one community, precludes the detection of overlapping communities that may exist in the network. Here we show that by defining communities as partition of a network's links, and thus allowing individual nodes to appear in multiple communities, we can quantify the extent to which each pair of communities in a network overlaps. We define two measures of community overlap and apply them to the community structure of five networks from different disciplines. In every case, we note that there are many pairs of communities that share a significant number of nodes. This highlights a major advantage of using link partitioning, as opposed to node partitioning, when seeking to understand the community structure of a network. We also observe significant differences between overlap statistics in real-world networks as compared with randomly-generated null models. By virtue of their contexts, we expect many naturally-occurring networks to have very densely overlapping communities. Therefore, it is necessary to develop an understanding of how to use community overlap calculations to draw conclusions about the underlying structure of a network.

## Introduction and Background

A structural feature of many networks is the organization of networks' nodes into communities. Heuristically, the number of edges connecting nodes within a given community to each other outweighs the number of edges connecting member nodes to nodes outside the community[1,2,3,4]. For example, consider the example of a social network, where nodes represent people and edges represent interactions between them. It would be reasonable for this network's communities to elucidate subgroups of individuals who belong to common workplaces, families, or social groups. By extending this idea to other contexts, we can assume that there is considerable insight to be gained from defining the communities underlying internet networks, metabolic networks, and communication and distribution networks[5]. A network's community structure is one approach from which to begin developing an understanding of the intricacies of the network as a whole. However, defining a network's communities is a challenging task for a few reasons: (1) relevant research lacks consensus on a singular, specific definition of communities, and thus (2) there is an absence of criteria to distinguish between a community and a non-community; finally, (3) there have been a myriad of proposed community detection algorithms, deemed by many to be "intractable"[4,6]. Additionally, many community detection methods operate under the assumption that every node belongs to exactly one community, which precludes the study of overlapping structure.

Nonetheless, it is illustrative to study community structure because it has been shown to be indicative of the properties guiding the underlying systems behind many networks[6]. Communities may provide insight into both the structural properties of a network as well as functional roles of subgroups of a network's nodes[3]. A network's optimal community structure

may differ depending on which of these two properties (structural or functional) researchers desire to illuminate, which leads to intrinsic difficulty in interpreting results[7].

By studying networks whose ground-truth communities (an a priori expectation of the community structure) are explicitly stated, researchers have evaluated the performance of structural community detection methods in identifying these pre-defined functional subunits of the network[3,6]. In 2012, researchers from the Department of Computer Science at Cornell University analyzed 10 community detection algorithms under this framework[6]. These methods included: breadth first search, two variations of a random walk algorithm, ($\alpha$,$\beta$), link communities[1], infomap, Louvain, Newman-Clauset-Moore, Markov clustering, and metsis. Because we seek a unified, context-independent way to study communities, the established collection of community detection methods is based on mathematical optimization, and different methods tend to produce significantly different community structures. However, the random-walk-based algorithms generally produce the community structure that most closely resembles the ground-truth communities, when known[6]. The plethora of proposed community detection methods can be grouped into broader categories[8], including traditional clustering methods, divisive algorithms[4], methods based on modularity[5], dynamic algorithms (including random walk[4,6]), and methods based on statistical inference.

One of the most common techniques for community detection seeks to maximize modularity, which is a measure of the quality of the identified communities[9]. However, a significant drawback of using this method is that it is subject to a resolution limit, whereby communities below a certain size (dependent on the size of the network and the interconnectedness of the communities) may not be detected[10]. In order to begin studying

overlapping communities, we may relax the assumption that each node belongs to one community. However, this makes modularity optimization even more difficult to use as a criteria for identifying a meaningful community structure.

Although the details of their implementations differ, node-grouping methods dominate the existing literature on network community detection[1,2,9,12]. The need to develop an alternative method can be validated by a simple example[2,12]: Consider a network of social ties between individuals as described previously, where nodes represent people and edges represent interactions between them. As we have stated, a node-partitioning community detection approach requires that the clusters create a partition of the network's nodes. Thus, each individual may belong to only one of the identified communities. However in reality, it is expected that in many cases an individual will belong to multiple social groups. The link-partitioning method for community detection holds that each edge of the network belongs to exactly one community[1,2]. However, nodes induced by each edge in a community may show up in multiple communities, effectively elucidating the overlapping communities that may exist in a network. A major shortcoming of the node community method can be resolved by using an analogous procedure to instead partition a network into disjoint and exhaustive sets of its edges. Further, this new method identifies the optimal community structure as the one that maximizes partition density (defined in Methods section) and eliminates the need to rely on modularity. We will explain the node community detection method and show how it gives rise to the link community detection method used in our analyses.

The first step in node community detection is to define a way to measure the similarity between the network's nodes. One of the widely-accepted ways to quantify node similarity,

$S(i,j)$, between nodes *i* and *j* is defined below, where. $n_+(i)$ and $n_+(j)$ represent the inclusive neighbors of nodes *i* and *j*, respectively[1,2].

$$S(i,j) = \frac{|\, n_+(i) \cap n_+(j) \,|}{|\, n_+(i) \cup n_+(j) \,|} \qquad (1)$$

As an analog to this method, link similarity will be defined as a comparison between two edges that share an impost node[1]. Because networks typically contain more edges than nodes, the number of pairwise similarity calculations that must be performed increases significantly when we adopt a link community approach.

The *Les Misérables* character interaction network[2] provides a comprehensible example of this phenomenon. The nodes in this network represent characters from the original novel. There exists a link between two nodes if their respective characters appeared in the same scene together. There are 77 nodes and 254 edges in the network, for an average degree of approximately 6.6. Figure 1 below shows a node similarity matrix and a link similarity matrix graphed on the same scale. A similarity matrix provides a visual representation of all similarity calculations using a color gradient. As the similarity between two entities increases, the shading in the appropriate cell of the plot darkens. In Figure 1, the same unit of area on each plot corresponds to one pairwise similarity comparison between either two nodes (at left) or two links (at right). We use the area of each matrix to visualize the extent to which the number of calculations needed to generate a link similarity matrix outweighs the number required to create a node similarity matrix for the same network.

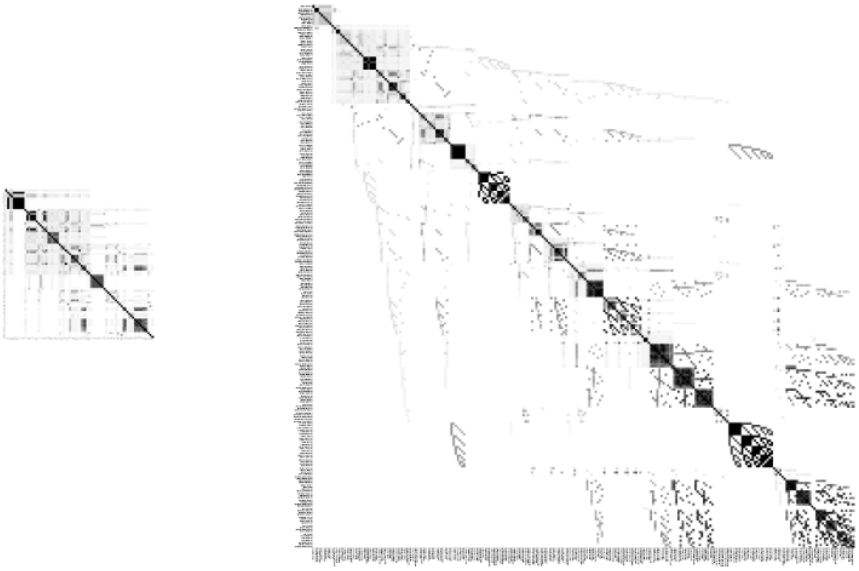


Figure 1

Link communities contain far more potential information than node communities. Here, two similarity matrices were created for the *Les Misérables* character interaction network, which has 55 nodes and 254 edges. Generating the node similarity matrix (left) requires $55^2$ = 5,929 calculations, while generating the link similarity matrix (right) requires over 10 times more, or $254^2$ = 64,516 calculations.

From the similarity matrices, we observe a common problem in studying networks: networks quickly become too large to be interpreted using static visualizations. Even though it is relatively small in comparison to many naturally occurring networks, the *Les Misérables* network appears to be approaching the maximum network size for which a link similarity matrix is a useful and interpretable visualization. The labels on the left and bottom edges of the matrix (which list each edge in the network) are illegibly small, so it is difficult even for a viewer to intuitively make sense of the pairs of highly similar edges in the network, even if s/he has contextual expertise. Figure 1 is intended to demonstrate a key difference between node and link community detection methods and to be evidence that new tools, such as interactive or

dynamic visualizations, should be explored as potentially useful alternatives to these similarity matrices.

After defining a way to quantify the similarity between nodes, traditional node community methods then perform a version of agglomerative clustering to group the nodes into communities. Agglomerative (bottom-up) algorithms start by assigning each node to its own community, which is then merged with other communities in successive iterations[2,13]. Alternatively, divisive (top-down) methods start with all nodes in a single community, which splits during each iteration[2,13]. We will focus on single-linkage clustering, in which communities containing the pair (or pairs) of nodes with the greatest similarity merge at each step. Other agglomerative clustering techniques are complete-linkage and average-linkage, which employ a different criterion for determining the pair of communities to merge at each step[2,13]. The whole procedure of hierarchical clustering can be summarized in a dendrogram. The leaf nodes of this diagram correspond to the nodes in the network[12]. At each merging step, the threshold at which a pair of communities merges is encoded in the height of the connection between them. This tool for tracking the history of hierarchical clustering of nodes has a direct translation to link community methods. The key difference is that the leaf nodes on a link dendrogram represent a network's edges, as opposed to its nodes[1]. In both cases, we can "cut" the dendrogram at a specific threshold to obtain the community structure at that merging threshold. As is the case with similarity matrices, dendrograms lose their visual interpretability when networks are large.

After performing hierarchical clustering on a network, we identify the network with the optimal set of communities and analyze their structure. There have been several proposed

methods for how to determine the best set of communities to reveal the underlying organization of a network[1,11,12,14]. In our link community method, we compute the partition density of the set of communities at each merging step and select the partition that has the maximum partition density to be the basis for subsequent analyses.

Following the general ideas of the node community algorithms, we use an analogous procedure, proposed in 2010 by Ahn, Bagrow, and Lehmann, and explained in the Methods section below, to define a link community method. Following hierarchical clustering and the identification of the set of communities with the greatest partition density, we define and investigate relevant statistics for quantifying the overlap that exists between communities. We utilize these methods on a corpus of five networks, some of which are known to have densely overlapping communities.

## Datasets

We have identified a corpus of five networks from different disciplines to analyze using link community methods. The first four listed below are naturally-occurring, while the fifth was constructed with some amount of (partially understood) human intervention. The description of these networks is followed by their visual representations, created using Cytoscape, and designed to give the reader a sense of the size and density of the networks we are studying. These basic properties are summarized in Table 1.

1. Word association: Nodes in the word association network represent words in the English language. Two nodes are connected by an edge if they were 'associated' with one another by a participant in the study from which the network was formed[15]. This

dense network is the largest graph in the corpus, in terms of number of nodes and number of edges.

2. Protein-protein interaction (PPI): The protein-protein interaction network provides insight into the realm of biological systems. Nodes represent proteins, and edges represent interactions between them. The data for this network were collected by yeast two-hybrid interaction mating[16]. This network has the lowest average degree of the networks in our corpus.

3. Primary school: The primary school network was released in a 2014 study published in *BMC Infectious Diseases*. Nodes represent students and teachers in a primary school, and edges represent face-to-face interactions between them[17].

4. Airports: Nodes represent the 500 airports in the United States with the most traffic. Two nodes are connected by an edge if there existed a direct flight (as of 2007) between the two airports that they represent[14].

Figure 2A
Top row (L-R): word association, PPI
Bottom row (L-R): primary school, airports

5. Football games: A node in this network represents a Division 1A college football team. An edge exists between a pair of teams if they played a regular season game against one another in Fall 2000[12]. The colors of the nodes represent the known conference structure of the teams, as shown in the legend. Note that there are five teams (Central Florida, Connecticut, Navy, Notre Dame, and Utah State) that do not belong to a conference, so they are grouped as Independents in gray.

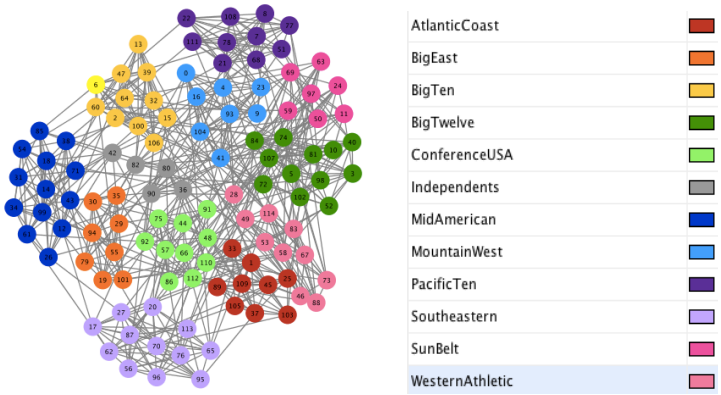



Figure 2B
Football conference network

Table 1

| *Network* | *Number of nodes* | *Number of edges* | *Average degree* |
|---|---|---|---|
| **Word association** | 5,018 | 55,232 | 22.01 |
| **PPI** | 1,647 | 2,518 | 3.06 |
| **Primary school** | 243 | 8,318 | 68.46 |
| **Airports** | 500 | 2,980 | 11.92 |
| **Football conferences** | 115 | 613 | 10.66 |

## Methods

The steps laid out below explain the details of our link community method. A complete implementation of these algorithms was written in Python, but ultimately we utilized a C++ implementation from a previous study[1] of link communities due to its higher efficiency. Minor modifications were made to its functionality to suit the scope of this research. We returned to Python to create the plots accompanying these analyses. The analyses of the community structure at the networks' maximum partition densities were dependent upon striking features of various plots and statistics. The link community method, in its entirety, follows in the steps below.

### Convert network data to a standard format.

A python script authored by Ahn, et. al. defines a standard network encoding in which a network's nodes are mapped to integers, and the network itself is represented by a list of space-separated integers that represent its links. This script was used to format each network in the corpus. The resulting file storing the network (characterized by the .pairs file extension) is compatible with the implementation of future procedures for calculating link similarities and performing hierarchical clustering. A useful

product of running the aforementioned python script is the .int2node file which contains the mapping of integers to the nodes they represent.

Calculate pairwise link similarities. The similarity $S(e_{ik},e_{jk})$ between two links $e_{ik}$ and $e_{jk}$ in an undirected, unweighted network is defined as follows, where $n_+(i)$ denotes the inclusive neighbors of node $i$ (the set containing node $i$ and its neighbors)[1].

$$S(e_{ik}, e_{jk}) = \frac{| n_+(i) \cap n_+(j) |}{| n_+(i) \cup n_+(j) |} \quad (2)$$

Note that link similarity ranges from zero (if two links do not share any impost nodes) to one (if there exists an edge between nodes $i$ and $j$ and these two nodes have an identical set of neighbors). Performing this calculation for each pair of links in the network has the potential to be an expensive operation. A file with the .jaccs extension stores an exhaustive list of edge pairs and their respective similarities. The information contained in this file can be visually represented in a similarity matrix (Figure 1), where darker shading indicates greater similarity between two links, and lighter colors on the gradient correspond to smaller similarity.

Perform hierarchical clustering. We elect to use single-linkage clustering, due to its efficiency over complete- and average-linkage clustering. The algorithm for carrying out this agglomerative procedure starts by assigning each link to its own community. Each future iteration identifies the pair of links that has the greatest similarity and merges the links in their respective communities. If there is a tie for maximum pairwise link similarity, the appropriate communities are merged simultaneously. The algorithm terminates when there exists one community that contains all of a network's edges. A link dendrogram stores all of the information about hierarchical clustering. Each leaf of the dendrogram represents an edge in

the network. The threshold at which two communities merge is represented by the height of the line representing their merge on the dendrogram[18]. By "slicing" the dendrogram at a particular merging threshold, we obtain the set of communities at this threshold. A clustermap (Figure 3) juxtaposes a dendrogram on two edges of a link similarity matrix and invites simultaneous analysis of these figures.

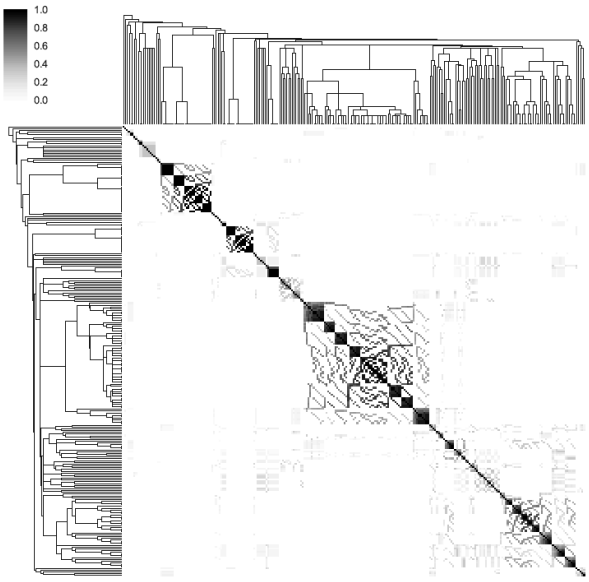



Figure 3
A clustermap of the *Les Misérables* character interaction network.

Owing to the computational cost of performing hierarchical clustering on the five-network corpus, we stepped through a finite set of merging thresholds, from [0,1] in increments of 0.005, and saved the resulting community structure at each step.

Compute the partition density of each link partition of the network. Partition density is a statistic used to determine the optimal link community structure for a network. Heuristically, partitions at the top of a dendrogram are dense, because they are comprised of few communities containing large numbers of edges. Partitions near the bottom of a dendrogram

typically contain many communities, each with relatively few edges, and thus, are less dense. The partition density, D on a partition of a network with M links is defined below.

$$D = \frac{2}{M}\sum_{c} m_c \frac{m_c - (n_c - 1)}{(n_c - 2)(n_c - 1)} \tag{3}$$

In this formula, $m_c$ and $n_c$ are the number of links and nodes, respectively, in a cluster within the partition[1]. A community containing two disconnected edges reaches the minimum quantity of D=-2/3. The maximum value for partition density, D=1, occurs when a community is a fully connected clique. It follows that as the value of D increases, a community bears less resemblance to a tree and progresses toward becoming a clique. We calculated the partition density at each recorded link partition of the network from the hierarchical clustering step. Figure 4 shows the partition density at a coarser resolution (merging thresholds in the interval [0,1] in increments of 0.05) for each of the networks in the corpus. We see that the range of partition densities differs greatly between networks.

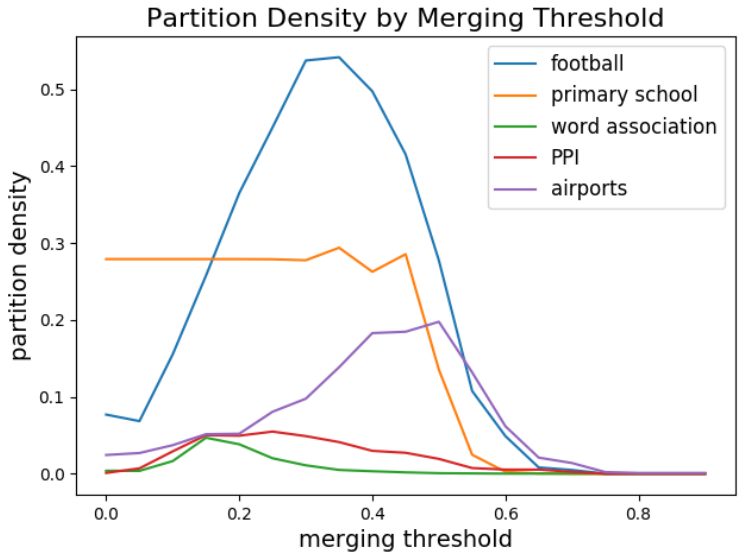



Figure 4

Partition density and corresponding merging threshold for five-network corpus.

After performing these calculations, we identify the merging threshold at which the maximum partition density is achieved, and we used the corresponding community structure for subsequent analyses.

Define and compute overlap statistics on networks. We define two measures by which to quantify the overlap between two link communities: the Jaccard *J* and the overlap coefficient Ω. The Jaccard extends directly from our link similarity calculation and is defined as follows, where A and B represent the set of nodes induced by the links of two communities, $c_A$ and $c_B$.

$$J(c_A, c_B) = \frac{|\ A\ \cap B\ |}{|\ A\ \cup B\ |}$$

Simply put, this quantity is the ratio between the number of shared nodes in two communities and the size of the union of their nodes. The overlap coefficient Ω determines overlap slightly differently by comparing the proportions of the size of the difference between two communities to the size of each of them. The smaller of these two proportions is subtracted from the statistic's maximum value (one).

$$\Omega(c_A, c_B) = \ 1 - \min\left(\frac{|A - B|}{|A|}, \frac{|B - A|}{|B|}\right)$$

Figure 5 below provides a visual explanation of this measure.

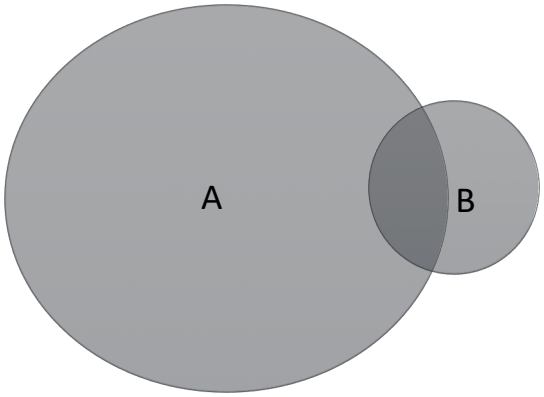



Figure 5

Using communities A and B represented above, we see that $\frac{|B-A|}{|B|} < \frac{|A-B|}{|A|}$, so the overlap coefficient $\Omega$ of these two communities is one minus the proportion of nodes in community B and not in community A, relative to the size of community B.

Due to the fact that all possible values of $\Omega$ are rational numbers, we observe step-like patterns and features similar to those of discrete data in the plots that describe this statistic.

Generate networks using the configuration model. The configuration model provides a framework for generating random graphs that preserves a given degree sequence. An important requirement for this model is that the sum of the degree sequence must be even. In this study, we use the configuration model to generate one hundred models that correspond to the degree sequences of each of the networks in the corpus (for a total of 500 network models). Python's networkx package supports this model[19], which makes the generation of these models a simple process. The graph that results from using the built-in configuration models functions is a multigraph, so we simply remove any self-loop edges, without concern that this will have a significant impact on the resemblance of the model's degree sequence to the degree sequence of the original network. Figure 6 shows one of the random models generated from the degree sequence of each network. We see that they are similar in

appearance to the original networks. One observation, however, is that model of the PPI network has many connected components. A simple investigation revealed that, on average, the largest connected component of a random model of the PPI network contains 91.5% of the model's nodes.

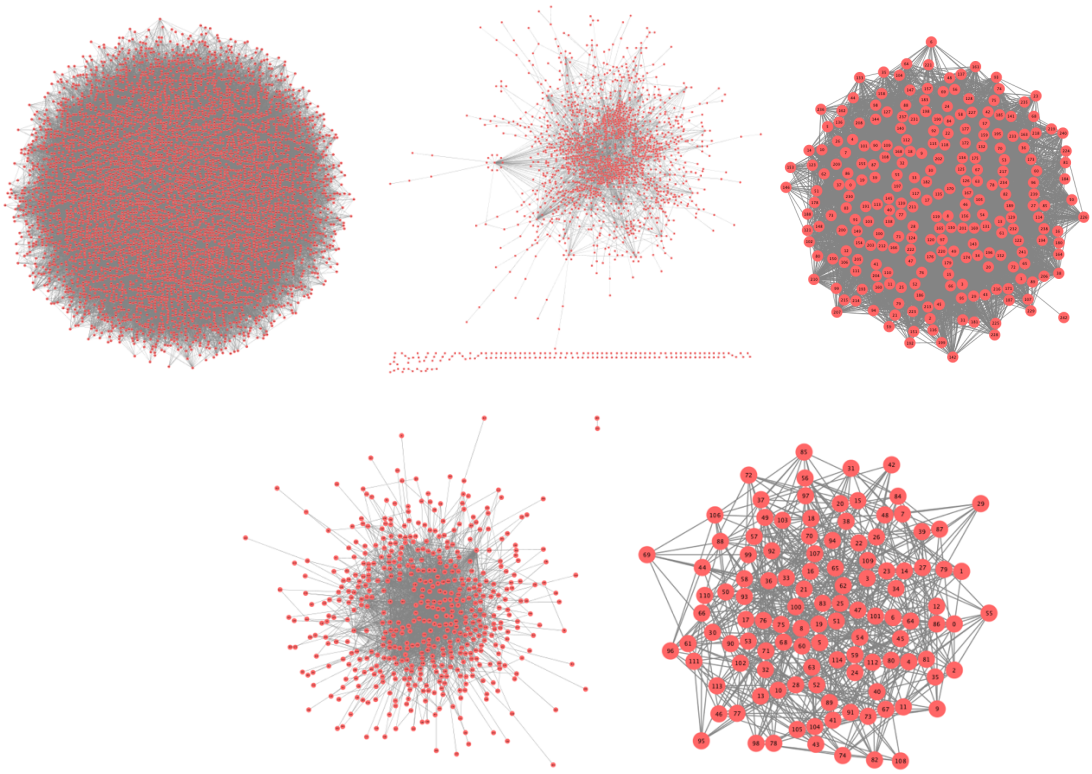
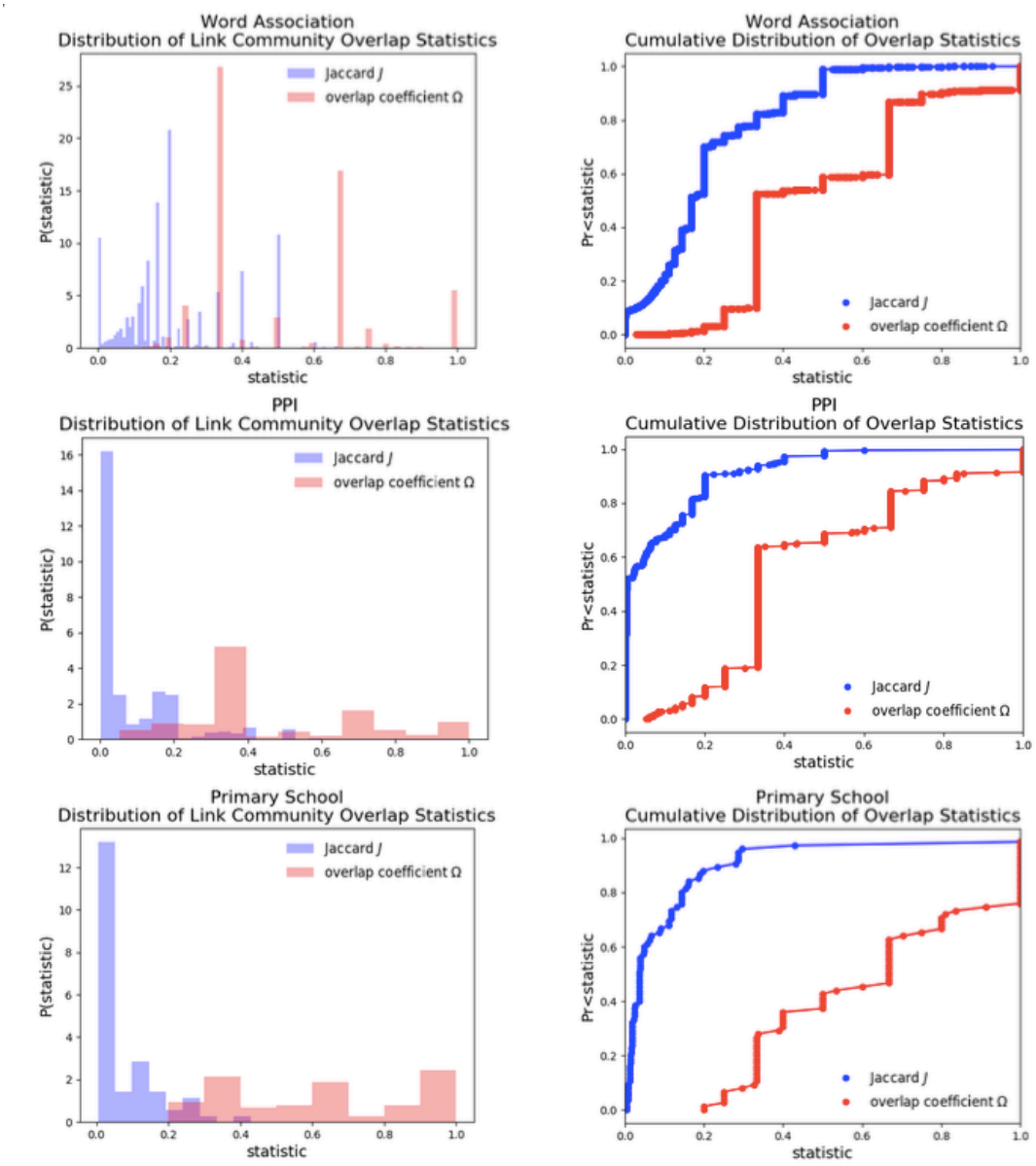


Figure 6
Models generated using the configuration model.
Top row (L-R): word association, PPI, primary school interactions
Bottom row (L-R): US airports, football teams

Investigate overlap statistics. Finally, we investigate the similarities and differences between the overlap statistics we calculated on the random models and those we computed on the real network data. This step involves analyzing distributions, calculating summary statistics, and performing statistical analyses. The beginning of this process was guided by the general task of making comparisons between the real and randomly-generated networks.

## Results

For each of the five networks in our corpus, we have identified the link partition that achieves the greatest partition density. Using this set of communities, we compute *J* and Ω (defined in Methods) between every pair of overlapping communities. We can visualize these results in the histograms and cumulative density plots in Figure 7 below.



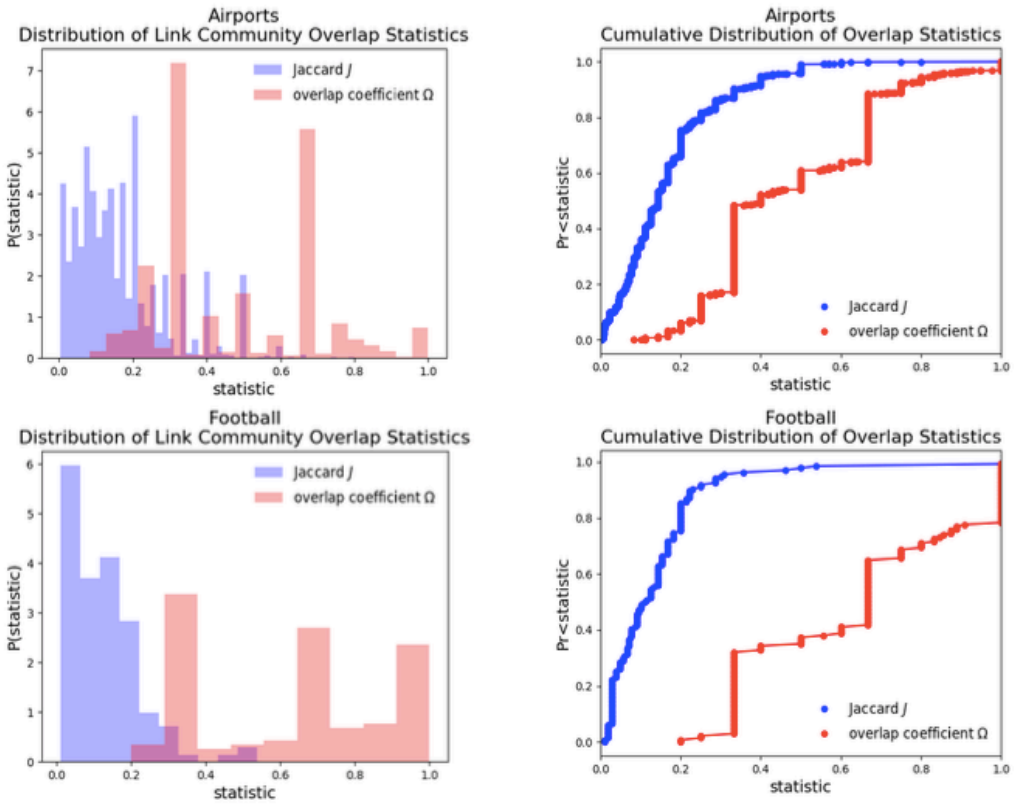


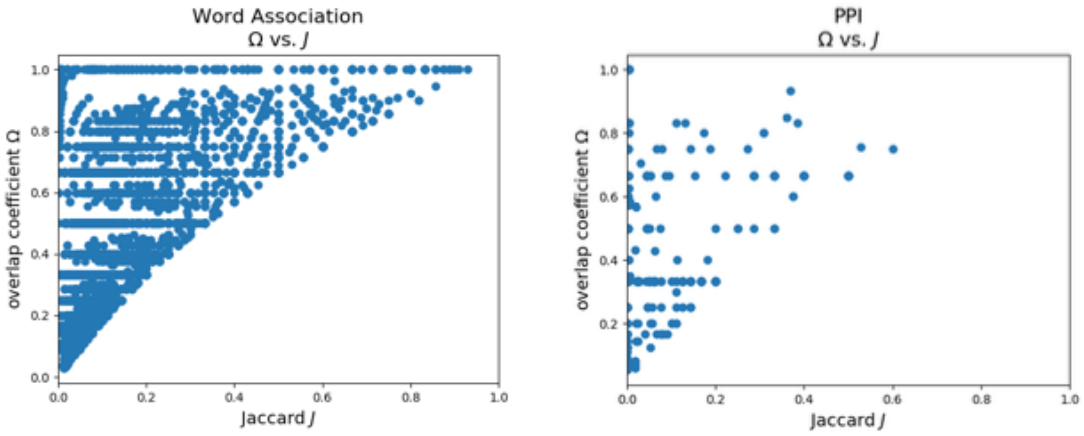

Figure 7

Histograms and cumulative density plots of overlap statistics for each network. From the histograms, we observe that Ω is generally greater than *J*, and in the cumulative density plots, we note a stepwise pattern of increasing probability.

There are a few noteworthy observations that we make from these plots. First, in the histogram for each of the networks, we see that Ω is generally greater than *J*. Considering the definition of each of these statistics, we note that if a small community is almost completely contained within a considerably larger one, their value of Ω will be quite high (close to the maximum value of one). However, *J* may not necessarily be as high, since the larger community inflates the size of the two communities' union and effectively lowers this statistic. The cumulative distribution plots are provided to make the discretization of Ω more apparent, and to eliminate the visual effects of binning on the interpretation of the statistics' distributions. In the cumulative distribution plots, we observe noticeable stepwise increases in probability at

$\Omega$ values of $^1/_3$, $^1/_2$, and $^2/_3$, while *J* generally increases more smoothly. This pattern is consistent across each of the five networks in the corpus, which range considerably in terms of context, size, and average degree.

Next, we investigate the relationship between *J* and $\Omega$ calculated on each pair of overlapping communities. Both of these statistics can take any value in the range 0 to 1. Had these statistics quantified overlap by the exact same criteria, we would have seen a scatterplot that could be modeled by a line with slope=1. However, as shown in Figure 8 below, we observe that a pair of communities can have values of *J* and $\Omega$ that are quite dissimilar from one another. In particular, there are many cases in which the value of $\Omega$ calculated between two communities is much higher than the value of *J* calculated on these same communities (which corresponds to the points in the upper left corner of the scatterplots in Figure 8). However, we do not see any cases of a pair of communities having high *J* and low $\Omega$, which would lie in the bottom right corner of these scatterplots. This observation is consistent with our claim that $\Omega$ tends to be greater than *J*. This result suggests that *J* and $\Omega$ are not redundant statistics; taken together, they provide more information about overlapping community structure than either statistic can provide on its own.



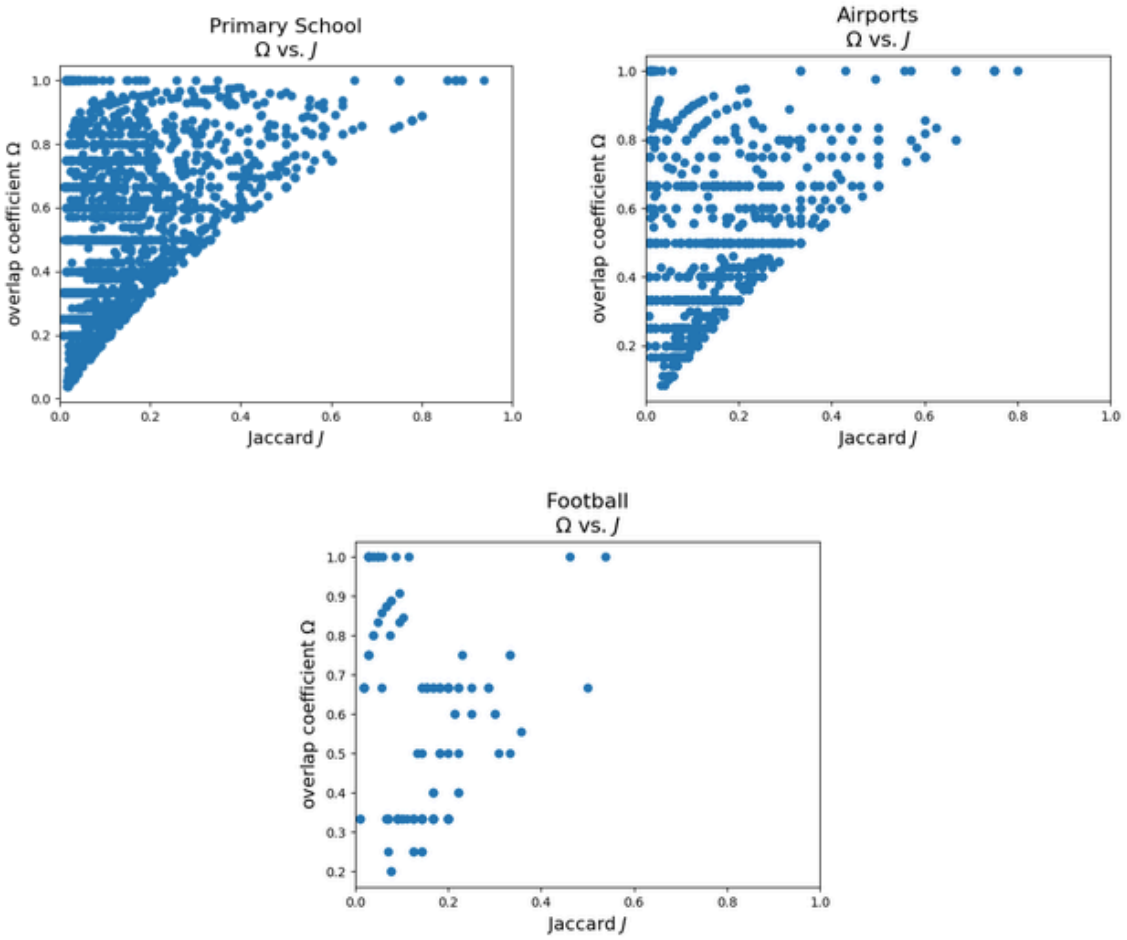



Figure 8

Scatterplots of overlap coefficient Ω by Jaccard *J* for each of the networks. We note that a pair of communities may have disparate values of Ω and *J*, but that if this is the case, it is Ω that significantly exceeds *J*.

To further explore the differences between Ω and *J*, we constructed histograms to visualize the distribution of Ω minus *J* in each of our constituent networks (Figure 9). By visual inspection, we see that these distributions are quite similar to one another; they are skewed to the right, but there is a sizable group of community pairs whose difference in overlap statistics is very close to one (the maximum possible difference).

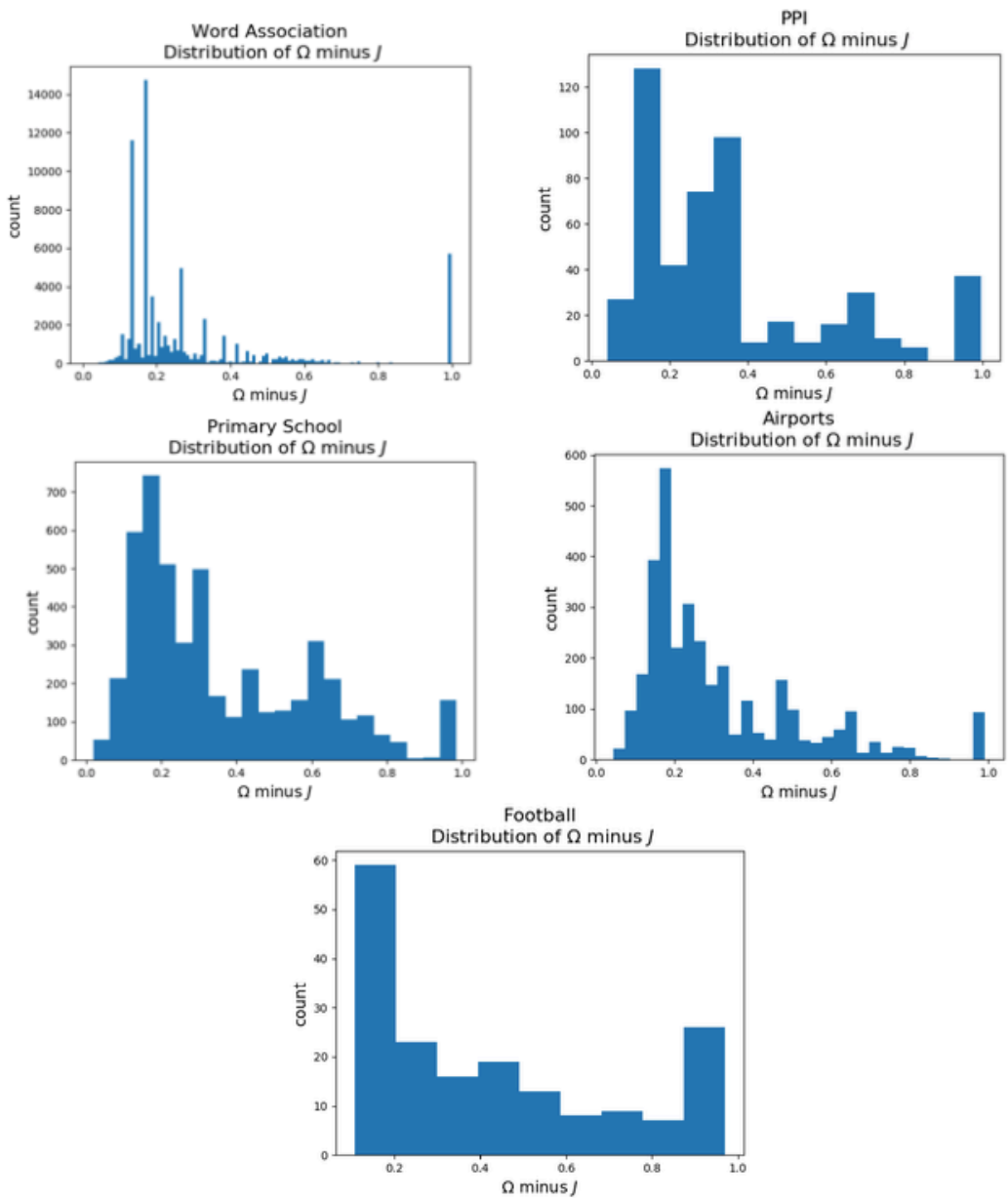



Figure 9

Histograms showing the distribution of the differences between the overlap coefficient Ω and Jaccard *J* of each pair of communities in each of the networks. These distributions are quite similar to one another in shape, center, and spread.

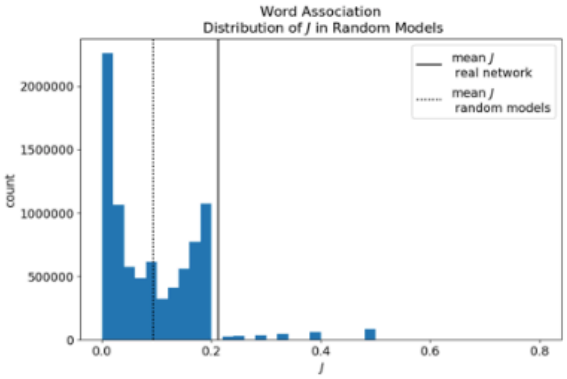
The summary statistics provided in Table 2 confirm that the mean and standard deviation of the overlap statistic differences are quite similar across each of the five networks in our corpus.

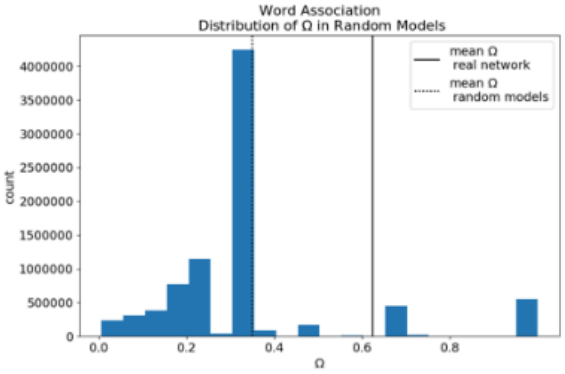
Table 2

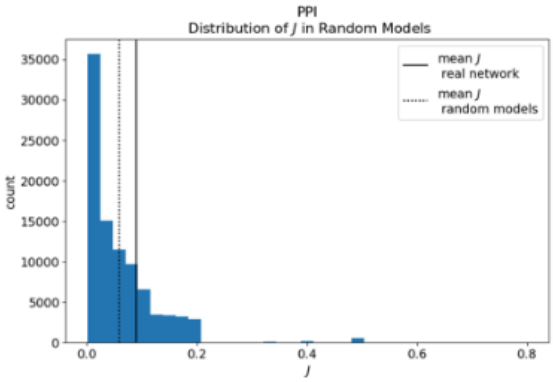
| *Network* | *mean difference* $(\Omega - J)$ | *standard deviation* $(\Omega - J)$ |
|---|---|---|
| **Word association** | 0.297 | 0.246 |
| **PPI** | 0.356 | 0.252 |
| **Primary school** | 0.356 | 0.231 |
| **Airports** | 0.312 | 0.205 |
| **Football conferences** | 0.434 | 0.290 |

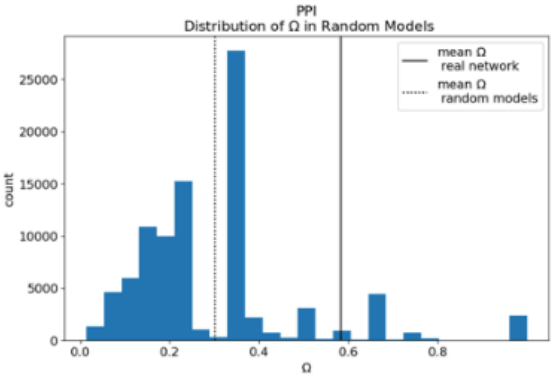
The fact that the mean difference is positive in every case and the histograms in Figure 9 have only positive values on the x-axis reveals that $\Omega$ always exceeds $J$. This is not an unexpected result given that $\Omega$, by definition, subtracts the minimum of two values from the maximum possible value of the statistic.

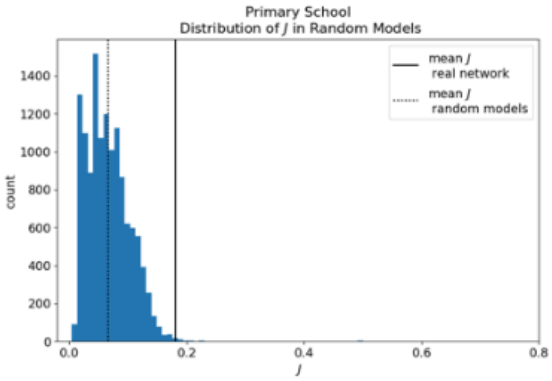
We have investigated basic observations regarding the overlap statistics calculated on each of our five networks. Next, we compare the overlap statistics computed on the real networks to the overlap statistics computed on 100 null models for each network, created using the configuration model, which preserves the network's degree sequence. For each random model of each network, $J$ and $\Omega$ were computed for each pair of link communities sharing one or more nodes at the maximum partition density. The distribution of these statistics by network (and thus, preserved degree distribution) are shown in Figure 10.

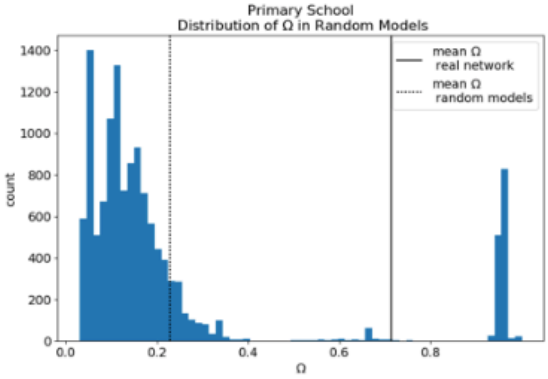

Word Association
Distribution of J in Random Models
mean J
real network
mean J
random models
count
J

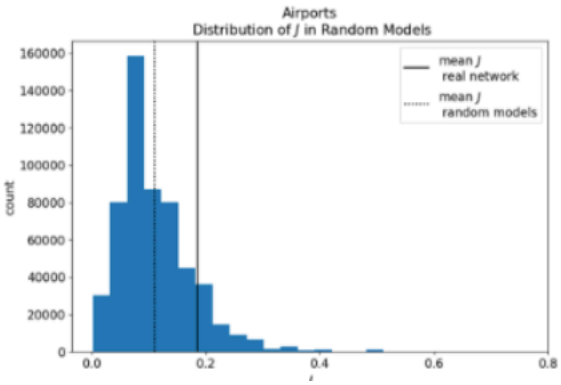

Word Association
Distribution of Ω in Random Models
mean Ω
real network
mean Ω
random models
count
Ω

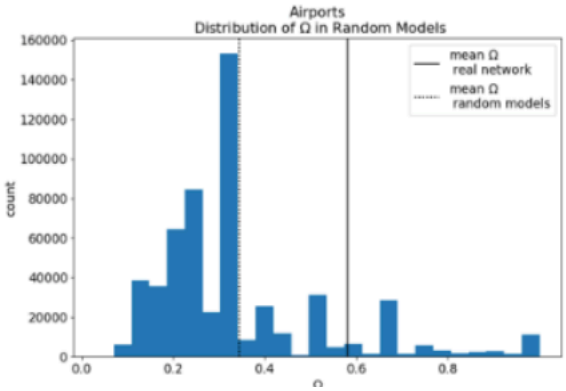

PPI
Distribution of J in Random Models
mean J
real network
mean J
random models
count
J


PPI
Distribution of Ω in Random Models
mean Ω
real network
mean Ω
random models
count
Ω


Primary School
Distribution of J in Random Models
mean J
real network
mean J
random models
count
J


Primary School
Distribution of Ω in Random Models
mean Ω
real network
mean Ω
random models
count
Ω


Airports
Distribution of J in Random Models
mean J
real network
mean J
random models
count


Airports
Distribution of Ω in Random Models
mean Ω
real network
mean Ω
random models
count
Ω

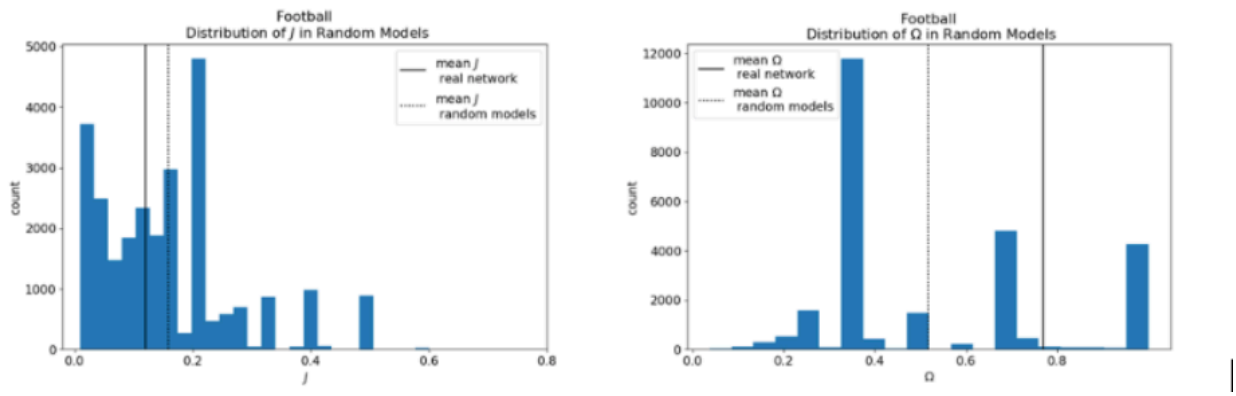



Figure 10

Histograms showing the distribution of the overlap coefficient Ω and Jaccard *J* on the random models for each network. The mean of each of these statistics is plotted for both the random and real networks.

We observe that, in all cases except for the distribution of *J* for the football network, the mean of the overlap statistics on the real network are less than the mean of the overlap statistics on the random models. This is summarized below in Table 3. Using this tabulated information, we perform z-tests to determine whether the means of *J* and Ω calculated on the random networks are different from the true means observed in the real networks. In every case, we obtain a p-value of approximately zero, indicating that the mean statistics on the null models differ from the true mean statistics on the real networks. This is evidence that overlapping community structure is not determined by a network's degree distribution.

Table 3

| *Network* | *mean J real network* | *mean J null models* | *std. dev. J null models* | *mean Ω real network* | *mean Ω null models* | *std.dev. Ω null models* |
|---|---|---|---|---|---|---|
| **Word association** | 0.212 | 0.094 | 0.088 | 0.624 | 0.350 | 0.215 |
| **PPI** | 0.089 | 0.059 | 0.068 | 0.583 | 0.302 | 0.185 |

| | | | | | | |
|---|---|---|---|---|---|---|
| **Primary school** | 0.181 | 0.066 | 0.036 | 0.713 | 0.228 | 0.268 |
| **Airports** | 0.186 | 0.111 | 0.065 | 0.581 | 0.344 | 0.186 |
| **Football conferences** | 0.119 | 0.156 | 0.115 | 0.769 | 0.517 | 0.261 |

Alternative random models of these networks should be used in future research to establish whether or not there is another feature of the network that may be underlying the observed overlapping community structure.

## Discussion

One of the fundamental challenges in studying networks, which is applicable to this exploration, is the difficulty of creating interpretable visualizations of large networks. Many naturally-occurring networks are incredibly large and dense, which suggests that there is a wealth of information to glean from them. However, these networks are often several orders of magnitude too large to visualize on a typical computer screen or piece of paper. In this research, this issue was most apparent in trying to produce meaningful visualizations of the word association network, which had over 55,000 edges. Even by experimenting with various layout heuristics using tools such as Cytoscape and Python's networkx package, there were too many nodes and edges to be able to pinpoint any notable structural features. Addressing this problem by creating tools to make visualizations dynamic and/or interactive is a body of research that could enrich the quality of network analyses like this one.

Another challenge in studying partitioning networks into communities is how to define the optimal set of communities. For this project, we utilized the partition at which the maximum partition density was achieved. However, by nature of calculating this value at a

finite range of merging thresholds, there is a risk that we missed a better partition (ie: one with a greater partition density) in between these thresholds, despite the fact that we stepped through them at a very fine interval (increments of 0.005). Further, using this criteria to select a link community structure resulted in there being between 53 and 6,152 communities in the optimal partition of the real networks. It is difficult to investigate what these communities represent, even when we understand the context of the network, simply because there are so many distinct groups. Regarding the issue of visual representation, it is also not particularly illustrative to encode the community structure in the color of the edges in a network drawing because it would be nearly impossible for viewers to distinguish between so many different colors.

There are several limitations that are specific to this project. First, we used a small corpus containing only five networks. Therefore, the results are not generalizable to a greater category of networks. The fact that there were some similarities between statistics computed on the networks (for example, average difference between $\Omega$ and *J*, Table 2) suggest that there are underlying properties to be discovered, regardless of the context of the network. Thus, there would be value in gathering a larger corpus of networks and investigating the durability of this apparent result. However, in Figure 4 we also saw that the maximum partition density for each of the networks and the merging threshold at which this value was achieved varied considerably between networks, indicating that structural features specific to contextual factors should be studied using more networks similar to each of the ones in our corpus.

The comparisons to randomly generated networks are based off of 100 networks created using the configuration model for each network. The reliability of the observed results

could be improved by increasing the number of models generated. If the efficiency of the calculations were improved, it would be reasonable to increase the number of random models by orders of magnitude.

The football network is not naturally-occurring. There are many factors at play to determine a collegiate game schedule, including, for example, geography, school budgets, and conference alignment. It would be worthwhile to study this network with other sports-related networks and analyze their link community structure as a way of looking beyond the role of conference structure in determining competition schedules.

There are several areas to extend this research. Most notably, we should continue to define and analyze new measures of community overlap. Using $\Omega$ and $J$, we should define a way to classify community pairs into subgroups indicating the extent to which they overlap, ranging from barely overlapping to completely nested. Finally, we should use alternative random models to generate the null models for each network and compare their overlap statistics to those computed on the real networks.

## Conclusion

We have utilized link communities to investigate the overlapping community structure in a corpus of five networks from various disciplines. Using two measures, the Jaccard $J$ and the overlap coefficient $\Omega$, we examine the extent to which nodes are shared between pairs of link communities. Both of these statistics are defined as rational numbers, so we observe stepwise patterns in their cumulative densities. We found that the value of $\Omega$ between two communities always exceeds their value of $J$, and the distribution of the differences between these two measures is quite consistent across each of the networks in the corpus.

We used the configuration model to generate 100 null models of each network and compared overlap statistics on their link communities to overlap statistics on the real networks' link communities. Preliminary results suggest that overlapping community structure is not a product of a given network's degree distribution, but this question is a natural basis for future research on this topic.

## Acknowledgements

I would like to thank Professor Bagrow for his support and understanding throughout the research process. He helped me develop the knowledge on which to base my research and provided guidance in the project design, project management, implementation, analysis, and writing steps of the undergraduate thesis process.

## References


1. Ahn, Y. Y., Bagrow, J. P. & Lehmann, S. Link communities reveal multiscale complexity in networks. *Nature* **466**, 761-765 (2010).

2. Barabàsi, A.-L. *Network Science*. (Cambridge Univ. Press, 2015).

3. Yang, J. & Leskovec, J. Defining and evaluating network communities based on ground-truth. *Knowledge and Information Systems* **42**, 181-213 (2015).

4. Newman, M.E.J. & Girvan, M. Finding and evaluating community structure in networks. *Phys. Rev. E* **69**, 026113 (2004).

5. Newman, M.E.J. Modularity and community structure in networks. *Proc. Natl. Acad. Sci.* **103**, 8577-8582 (2006).

6. Abrahao, B., Soundarajan, S., Hopcroft, J. & Kleinberg R. On the separability of structural classes of communities. In *Proc. Of 18$^{th}$ ACM SIGKDD international conf. on knowledge discovery and data mining*, 624-632. (2012).

7. Radicchi, F. *et al*. Defining and identifying communities in networks. *Proc. Natl. Acad. Sci.* **101**, 2658-2663 (2004).

8. Fortunato, S. Community detection in graphs. *Physics Reports* **486**, 75-174 (2010).

9. Evans, T. S. & Lambiotte, R. Line graphs, link partitions, and overlapping communities. *Phys. Rev. E* **80**, 016105 (2009).

10. Girvan, M. & Newman, M. E. J. *Proc. Natl. Acad. Sci.* **99**, 7821-7826 (2002).

11. Rencher, A. C. & Christensen W. F. *Methods of Multivariate Analysis*. (John Wiley & Sons, Inc., Hoboken, NJ, 2012).

12. Latora, V., Nicosia, V. & Russa G. *Complex Networks: Principles, Methods, and Applications*. (Cambridge Univ. Press, 2017).

13. Nelson, D.L. McEvoy, C. L. & Schreiber, T. A. The University of South Florida word association, rhyme, and word fragment norms. (1998).

14. Stelzl, U. *et al*. A human protein-protein interaction network: a resource for annotating the proteome. *Cell* **122**, 957-968 (2005).

15. Stehlé, J. *et al*. High-resolution measurements of face-to-face contact patterns in a primary school. *PloS one* **6**, e23176 (2011).

16. Kalinka, A. T. & Tomancak, P. linkcomm: an R package for the generation, visualization, and analysis of link communities in networks of arbitrary size and type. *Bioinformatics* **27**, 2011-2012 (2011).

17. Hagberg, A., Swart, P. & Schult, D. Exploring network structure, dynamics, and function using NetworkX. Los Alamos National Lab, United States (2008).

18. Forunato S. & Barthélemy M. Resolution limit in community detection. *Proc. Natl. Acad. Sci.* **104**, 36-41 (2007).

19. Lancichinetti A. & Fortunato S. Limits of modularity maximization in community detection. *Phys. Rev. E* **84**, 066122 (2011).