Towards a Better Quality Metric for Graph Cluster Evaluation

Hélio Almeida[1], Dorgival Guedes Neto[1], Wagner Meira Jr[1], Mohammed J. Zaki[2]

[1] Universidade Federal de Minas Gerais, Brazil
{helio, dorgival, meira}@dcc.ufmg.br
[2] Rensselaer Polytechnic Institute
zaki@cs.rpi.edu

Abstract. The process of discovering groups of similar vertices in a graph, known as graph clustering, has interesting applications in many different scenarios, such as marketing and recommendation systems. One of the most important aspects of graph clustering is the evaluation of cluster quality, which is important not only to measure the effectiveness of clustering algorithms, but also to give insights on the dynamics of relationships in a given network. Many quality metrics for graph clustering evaluation exist, but the most popular ones have strong biases and structural inconsistencies that cause the quality of their results to be, at least, doubtful. Our studies showed that, while in general those popular quality metrics do a good job evaluating the external sparsity between clusters, they do poorly when evaluating the internal density of those clusters, ignoring essential information (such as a cluster's vertex count) or having its internal density component ignored in practice because of its computational cost. In this article, we propose a new method for evaluating the internal density of a given cluster, one that not only uses more complete information to evaluate that density, but also takes into consideration structural characteristics of the original graph. With our proposed method, the internal density of a cluster is evaluated in terms of the expected density of similar clusters in that same graph, in contrast to the traditional quality metrics available, where clusters from different graphs are compared by the same standards. We believe that, if used in conjunction with a good external sparsity evaluation metric, like conductance, this method will help to obtain better, more significant graph clustering evaluation results.

Categories and Subject Descriptors: H.2.8 [**Database Applications**]: Data Mining

Keywords: Graph Clustering, Quality Metrics, Quality Evaluation, Graph Mining

1. INTRODUCTION

Graph clustering, the process of discovering strongly related groups in a graph [Schaeffer 2007], has many applications in areas such as social network analysis (finding groups of related people), e-commerce (recommendations based on a group's opinions) and bioinformatics (gene expression data classification, spreading of epidemics).

The basic structure of a well formed group is open to discussion, but the classical and most adopted view is based on the concept of homophily, which states that similar elements have a greater tendency to group with each other than with other, less similar elements [Newman and Girvan 2003]. When working with simple and undirected graphs, homophily is usually evaluated in terms of edge densities, with a cluster having more edges linking its own elements among themselves (higher internal density) than linking them to the rest of the graph (sparser external connections). However, discovering such edge-dense clusters in graphs is a complex task since, by this definition, a cluster can be anything between a connected subgraph and a maximal clique, with the better results leaning towards the latter.

This work was partially sponsored by CNPq, CAPES, Fapemig and Instituto Nacional de Ciência e Tecnologia para a Web - InWeb (MCT/CNPq 573871/2008-6)

Since the problem of clustering does not have an exact solution, many heuristics have been proposed to find clusters which maximize both intracluster density and intercluster sparsity. Examples of such algorithms are MCL [van Dongen 2000; 2008], K-means [Steinbach et al. 2000] and Spectral [Kannan et al. 2004; Schaeffer 2007]. However, since those algorithms are heuristics, there is no formal guarantee that the results obtained through them are the best possible.

When a graph is small, with only a few vertices and edges, results of a clustering found for it can be evaluated manually. However, as the size of the graph grows, manual evaluation becomes unfeasible. If that is the case, quality evaluation metrics may be used as indicators of cluster quality. Those metrics aim to evaluate a cluster (or clustering) in terms of the most important characteristics expected from good clusters, providing scores that can be used to compare the quality of clustering results.

However, even the most popular and well regarded quality evaluation metrics for graph clustering, such as modularity [Newman and Girvan 2004] and conductance [Kannan et al. 2004], possess strong structural biases that make them favor clusterings mainly because of how many clusters they have, making them unreliable [Almeida et al. 2011].

So, there is a great need to find other, more accurate ways to evaluate cluster quality. In this article, we extend the evaluation of cluster quality metrics, and conclude that they do not correctly evaluate one of the key elements of cluster quality: the cluster's internal density. Also, they evaluate all clusters by the same standards, causing clusters from naturally sparser graphs (such as technological networks, for example) to be unfairly penalized.

With that in mind, we propose a new method to evaluate a cluster's internal density. Our method uses more complete information to evaluate the internal density of a cluster. Also, it uses the characteristics of the studied graph in order to discover what are the density thresholds that identify interesting clusters in that particular case. This allows for a more "local" evaluation, removing the penalization that sparser graphs usually receive.

2. RELATED WORK

The automatic evaluation of cluster quality is crucial to graph clustering, as it is increasingly harder and expensive to evaluate cluster quality manually as graphs grow in size. Many quality metrics were proposed in the literature, such as modularity [Newman and Girvan 2004], conductance [Kannan et al. 2004], and silhouette [Tan et al. 2005], among others.

Many works rely on such quality metrics as a source of "ground truth", using them to score clusterings obtained by selected graph clustering algorithms in order to compare them in terms of result quality. However, they do so assuming that those metrics are good enough, without any evaluation of the correctness of the chosen metric. Examples of such works are [Satuluri and Parthasarathy 2009], which uses conductance to compare its proposed clustering algorithm to normalized cut and multilevel clustering algorithm implementations, and [Danon et al. 2005], which uses modularity to compare agglomerative, divisive and modularity maximization techniques for graph clustering.

Other works try to do something similar, only using more than one quality metric in their evaluation in order to minimize the impact of any possible biases of the quality metrics. For example, [Brandes et al. 2008] compared three algorithms for graph clustering (Markov clustering, iterative conductance cut and geometric MST clustering) using conductance, coverage and performance. Modularity and silhouette are used in [Gustafson et al. 2006] to compare K-means and hierarchical clustering using.

Those popular quality metrics are not only used to compare clustering algorithms. In [Leskovec et al. 2008], the authors try to evaluate what is the basic structure of a good community. To do so, they used Network Community Profile (NCP) plots, which showed the best external conductance values for communities of different sizes, which were obtained through traditional graph clustering algorithms. In a follow-up work, the authors add other quality metrics, such as modularity, to the

evaluation of the same problem [Leskovec et al. 2010]. Nevertheless, those two papers assume that the quality metrics used are good enough to correctly evaluate the quality of a cluster, which is a disputed claim. Also, the only clusters evaluated in said works are be the ones found by the clustering algorithm chosen, and this might add the biases from the clustering algorithms to their final results.

All of those works assume that the metric or metrics used are good enough to be used as ground truth in their evaluations. However, it has been shown in [Almeida et al. 2011] that even widely used quality metrics such as modularity and conductance have strong biases which make them unreliable, specially when working with larger graphs. Also, another paper from [Zaidi et al. 2010] claims that edge density, one of the main characteristics observed by most of the popular quality metrics, might not be good for cluster evaluation, as it alienates star-like formations, which are more common in technological networks then edge-dense clusters. It proposes a metric that uses the distance between vertices to evaluate internal density (like the silhouette index), comparing this new metric to modularity. Although we agree that a star-like formation might mean a good cluster in cases like technological networks, it doesn't mean that they will be as good as more edge-dense clusters in complex networks of a more social origin. Also, like silhouette, a distance-based metric might be biased towards smaller clusters, as they will always have the smallest distances among the clusters studied.

The previously discussed papers used or proposed generic evaluation metrics for cluster evaluation, which were based on graph characteristics. Another way to evaluate cluster quality is to use external information that can be considered ground truth. This approach is good, but it is very dependent on the existence of such adequate external information. One example of such work is [Qi et al. 2012], where the authors try to cluster Flickr images using social links. They use image tags, which are added by the community, as the ground truth for each image's category. Another example is [Gargi et al. 2011], that tries to find communities of similar YouTube videos and generate named clusters that are coherent with their content, using the YouTube video graph. They use manual evaluation of the discovered name for each cluster as a form of validation of their results. Such works assume that the connections in a graph and the labels found in them are strongly related, which is not necessarily the case in general. In our approach, we rely solely on the structure of the graph, which is available in all cases.

3. CURRENT GRAPH CLUSTERING EVALUATION METRICS AND THEIR PROBLEMS

The most accepted definition of a graph cluster is based on the concept of homophily or assortative mixing: elements have a greater tendency to form bonds with other elements with whom they share common traits than with others [Newman 2003]. Applying this concept to graphs, a cluster's elements will display stronger than expected similarity among themselves, while also having sparser than expected connections to the rest of the graph. Element similarity can be derived from many graph characteristics, such as edge density [Girvan and Newman 2002], vertex distance [Tan et al. 2005] or labels [Zhou et al. 2009].

A theoretically perfect clustering, structurally-wise, would be one formed by clique-like clusters completely disconnected from the others, as this configuration would allow for maximum internal density for each of the clusters and maximum sparsity of connections between them. Graph clustering quality metrics are models that try to evaluate how close to theoretically perfect a given cluster is, scoring them accordingly.

Unfortunately, the most commonly used cluster quality evaluation metrics in the literature possess fundamental flaws that make them strongly biased. In this section we will briefly present two of the most popular quality metrics presented in the literature and show why they are inadequate for cluster quality evaluation. We will also discuss if those metrics behave consistently with what is expected of good clusterings, that is, high internal edge density and sparse connections with other clusters.

The metrics studied in this article use only a graph's topological information, like vertex distance

and edge density, to evaluate the quality of a given cluster. A discussion on the biases of those quality metrics and others (silhouette, coverage and performance) can be found in previous work [Almeida et al. 2011].

3.1 Modularity

One of the most popular validation metrics for topological clustering, modularity states that a good cluster should have a larger than expected number of internal edges and a smaller than expected number of inter-cluster edges when compared to a random graph with similar characteristics [Newman and Girvan 2004]. The modularity score Q for a clustering is given by Equation 1, where e is a symmetric matrix whose element e_{ij} is the fraction of all edges in the network that link vertices in communities i and j, and $Tr(e)$ is the trace of matrix e, i.e., the sum of elements from its main diagonal.

$$Q = Tr(e) - ||e^2|| \tag{1}$$

The modularity index Q often presents values between 0 and 1, with 1 representing a clustering with very strong community characteristics. However, some limit cases may even present negative values. One example of such cases is in the presence of clusters with only one vertex. In this case, those clusters have 0 internal edges and, therefore, contribute nothing to the trace. Sufficiently large numbers of singleton clusters in a given clustering might cause its trace value to be so low as to overshadow other, possibly better formed, of its clusters and lead to very low modularity values regardless.

3.2 Conductance

The conductance [Kannan et al. 2004] of a cut is a metric that compares that cut's size (the number of edges that, if removed, would make the graph disconnected) and the weight of the edges in either of the two sub-graphs induced by that cut. The conductance $\phi(G)$ of a graph G is the minimum conductance value between all of its clusters.

Consider a cut $K = (S, V \setminus S)$ that divides G into k non-overlapping clusters $C_1, C_2 \dots C_k$. The conductance of any given cluster $\phi(C_i)$ can be obtained as shown in Equation 2, where $a(C_i) = \sum_{u \in C_i} \sum_{v \in V} w(u, v)$ is the sum of the weights of all edges with at least one endpoint in C_i. This $\phi(C_i)$ value represents the cost of one cut that bisects G into two vertex sets C_i and $\bar{C_i} = V \setminus C_i$. Since we want to find a number k of clusters, we will need $k-1$ cuts to achieve that number. In this article we assume the conductance for the whole clustering to be the average value of those $(k-1)$ ϕ cuts, as formalized in Equation 3.

$$\phi(C_i) = \frac{\sum_{u \in C_i} \sum_{v \notin C_i} w(\{u, v\})}{min(a(C_i), a(\bar{C_i}))} \tag{2}$$

$$\phi(G) = avg(\phi(C_i)) \ , \ C_i \subseteq V \tag{3}$$

Based on this information, it is possible to define the concept of intracluster conductance $\alpha(C)$ (Eq. 4) and the intercluster conductance $\sigma(C)$ (Eq. 5) for a given clustering $C = C_1, C_2, \dots, C_k$.

$$\alpha(C) = min_{i \in \{1,\dots,k\}} \phi(G[C_i]) \tag{4}$$

$$\sigma(C) = 1 - max_{i \in \{1,\dots,k\}} \phi(C_i) \tag{5}$$

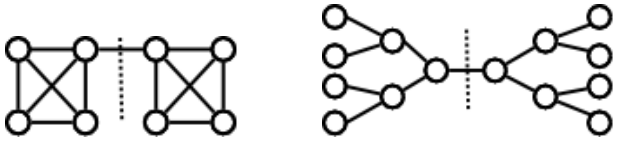
Intracluster conductance considers the conductance of a minimum cut with maximum flow obtainable from a subgraph induced by a cluster $c \in C_i$ (represented as $G[C_i]$ in Eq. 4). A graph's intracluster conductance is the minimum of such conductances, with a low value meaning that at least one of those clusters may be too sparse to be good. The intercluster conductance is the complement of the maximum conductance value of the clustering, so lower values might show that at least one of the clusters have strong connections outside of it and, therefore, is externally dense. So, a good clustering should have high values of both intra and intercluster conductance.

Alas, even though the use of both inter and intracluster conductance would give a better, more complete view of cluster quality, many works use only the external conductance as the cluster evaluation metric [Leskovec et al. 2008; Leskovec et al. 2010]. Because of that, unless otherwise stated, when we discuss conductance in this article, we mean external conductance.

3.3 The Problem with Internal Density

Observing the formulation of both modularity and conductance, it is possible to see something in common - both metrics use edge counts, but not vertex counts, to infer internal density and external sparsity. Modularity uses the matrix e, which counts edges connecting elements in and outside each cluster as the base of its formulation, while conductance uses the ratio between edges connecting elements from the evaluated cluster to other clusters and all edges with at least one endpoint on the evaluated cluster.

When evaluating external sparsity, it is not a problem to use only edge counts and ignore the vertex count, as this is a relationship between two clusters and, therefore, has little to do with the vertices themselves.



(a) Two dense clusters. (b) Two sparse clusters.

Fig. 1: Very different clusterings that are equally good for current quality metrics.

However, internal cluster density does not benefit from such restrictions. Density is a measure of *concentration* and, as such, needs to represent not only the number of elements observed, but the space they occupy. In this case, it means that the concept of internal cluster density can not be fully represented by just using the number of internal edges of a given cluster and ignoring the number of vertices of said cluster. For example, a clustering with 2 clusters composed of 1 external edge and 12 internal edges can represent many different scenarios if we ignore the vertex count of each cluster, like two 4-cliques connected by one edge (Figure 1a, definitely dense clusters) or two 6 vertex trees connected by one edge (Figure 1b, not dense at all). As they were proposed, both conductance and modularity would score those two clusterings equally, and with very good scores of 0.14 and 0.34, respectively.

Another problem is that not all networks are made equal. Some networks represent some kind of technological infrastructure, like airport flight connections, power grids or computer networks, where the monetary cost of creating an edge tends to generate sparser, more deliberate structures. Other networks can represent social networks, such as friendship networks on websites like Facebook or Twitter, where the cost of creating new connections is negligible and, therefore, tend to be denser. To assume that a cluster in a technological network with the same density as one from a social network should be similarly scored can be unnecessarily unfair, but that's what quality metrics like conductance, modularity, and others do.

The edge count based scoring method adopted by conductance and modularity causes the results obtained with those quality metrics to be always smaller for graphs that are naturally sparser, a behavior that can be seen in different works in the literature [Almeida et al. 2011]. Because of that, it's difficult to compare how good a clustering algorithm is for graphs of different origins, as the metric results will be mainly incomparable.

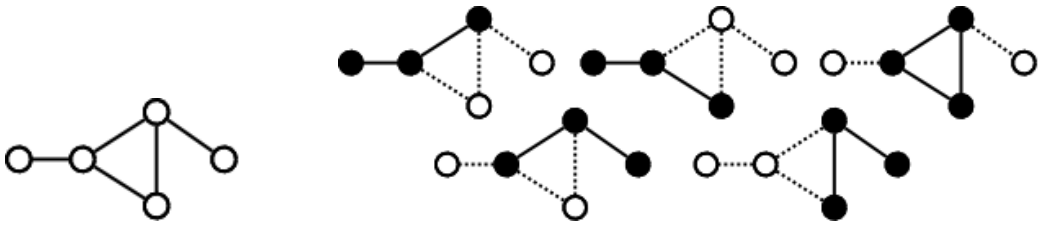
4. INTERNAL DENSITY EVALUATION METHOD

In the previous section, we saw that there is a strong need to find better ways to evaluate internal cluster density. The currently used quality metrics have two problems that affect their ability to correctly evaluate the internal density of a cluster: they only use edge count information and they evaluate networks of the most diverse origins by the same standards.

To solve the first problem, vertex count information should be added to the cluster evaluation process. One way this can be done is by comparing only subgraphs of the same size (vertex wise). This way, edge counts can be correctly seen as density indicators.

As for the second problem, it is necessary that the quality metric use information derived from the studied graph to identify the thresholds that indicate density/sparseness of the clusters found. Doing so would allow our new method to positively score clusters that can be considered dense for a given graph even if they would not be considered dense in other cases.

One way to employ those two kinds of information in order to evaluate a cluster could be as follows: the population of all subgraphs of the same size could be used to identify what are the expected density values for clusters of that size. This way, we can use the structural characteristics of that specific graph to discover the threshold that separates rare and dense clusters from common and sparse ones.



(a) Original graph. (b) Population of size 3 subgraphs.

Fig. 2: A simple graph and its subgraphs.

For example, consider that we want to define the density thresholds for a size 3 cluster obtained from the graph in Figure 2a. If we enumerate all possible size 3 induced connected subgraphs from this graph (as shown in Figure 2b), we can see that the majority of them have only 2 edges, while only one possible subgraph has 3 edges. Since this 3 edged subgraph has more edges than 80% of all the possible subgraphs with the same size, then a size 3 cluster with 3 internal edges found in this same graph should be positively scored based on this proportion.

However, the example in Figure 2 is small and very simple. As the graph grows in size, the population of possible connected subgraphs of a given size becomes so large as to be intractable to enumerate. So, it is necessary to use a sample of this population in order to use this kind of approach.

4.1 Sampling for Density Evaluation

Some works in the literature use clusters found through traditional graph clustering algorithms to evaluate cluster characteristics [Leskovec et al. 2008]. Other works use cluster quality metrics to evaluate results from traditional graph clustering algorithms [Brandes et al. 2008; Du et al. 2007;

Brandes et al. 2003]. In all those cases, the universe of evaluated clusters is limited to what the clustering algorithms identify as a (good) cluster. This kind of filtering may introduce biases on what kinds of structures will be evaluated and, therefore, may bias the final quality evaluation results.

To avoid this kind of bias in our evaluation process, we can use samples from the universe of all possible connected induced subgraphs with s vertices that exist in a given graph. Doing so, our results will represent, within statistical guarantees, the expected internal density values for a size s cluster found in a given graph. A sampling process that provides such guarantees will be discussed in Section 5.

The internal density values obtained in a couple of instances of this form of sampling can be seen in Figure 3, which plots the number of sampled 25-vertex subgraphs by edge count for three complex networks of different origins: Amazon Co-purchase, Yeast protein interactions and Google websites (social, biological and technological, respectively). Those networks will be better described on Section 6.

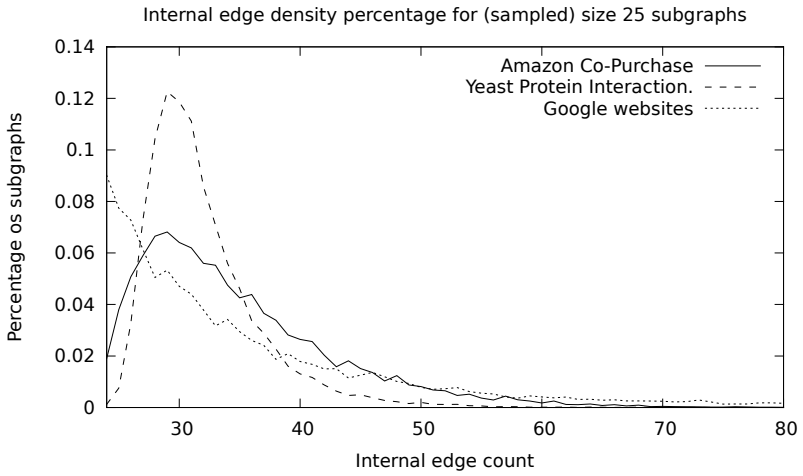


Fig. 3: Example of internal densities from subgraph samples ($s = 25$).

Since we are looking at induced subgraphs of a fixed vertex count, it is valid to equate edge counts and density here. It is possible to see that those density curves have their maxima very near the minimum density possible (since the subgraphs are connected), and that they are somewhat heavy tailed. That means that not only most of the sampled subgraphs of the given size are way too sparse to be considered good clusters, but also that there are a few subgraphs that are more dense than what is expected from the average case. We believe that it is possible to use the information presented in this kind of graph to evaluate cluster quality in a way that bears more significance to the studied graph.

5. METHODOLOGY

The methodology used in our experiments will be described in this section, and their results will be presented in Section 6. Our goal is to understand what are the expected values for internal cluster density for clusters of a given size. We'll use this information to generate inputs for a simple internal density evaluation model, which will be described at the end of Section 6.

The experiments consist of the sampling of induced connected subgraphs of different sizes from 14 different real world complex networks. The details of the sampling process and datasets used will be discussed shortly. We sampled subgraphs of 15 different sizes, from 5 to 300 vertices.

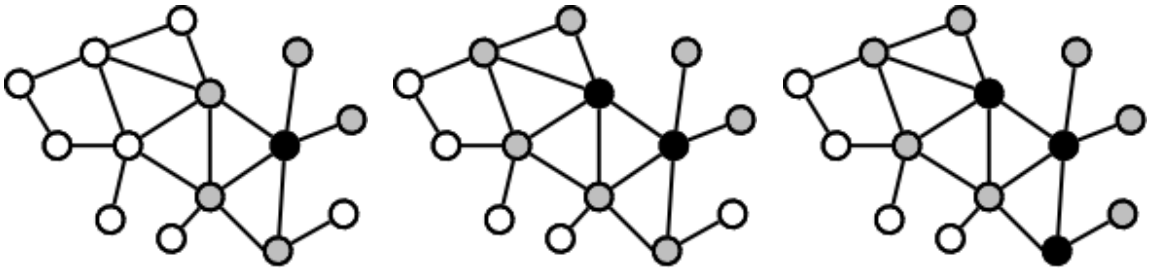
5.1 Sampling Process

Given a graph $G = (V, E)$, we want to identify what is the expected edge count for a connected induced subgraph with s vertices. Also, we want to do it without having to enumerate all possible of such subgraphs, as this can be unfeasible for larger graphs.

One way to do that is through sampling of the universe of all possible s-sized connected induced subgraphs from G. In order to obtain an estimate of the proportion of edge densities of this universe, with a confidence level of 99% and margin of error of 1%, we can perform a simple random sampling without replacement of at least 2477 subgraphs from this universe, a value obtained through Eq. 6, where $Z = 1 - \frac{\alpha}{2}$ (α is the confidence level desired), ME is the margin of error desired and p is the probability for one sampled element to have a given characteristic. Variable p was set with the suggested (and conservative) value of 0.5, which guarantees that a sample of the calculated size would have *at least* the level of confidence and measurement error chosen[1] [Lohr 2010]. We were even more conservative and did samples of 16000 subgraphs, more than enough to assure the measuring error and confidence level desired.

$$n = \frac{[Z^2 \times p \times (1-p)] + ME^2}{ME^2} \tag{6}$$

The process of choosing one random connected subgraph from said universe is done as follows: we pick one vertex randomly from all vertices in the graph (Figure 4a). We pick the next vertex randomly from the immediate neighborhood of the chosen vertex (Figure 4b). For the third and following vertices to be chosen, we pick randomly one of the vertices from the set of the immediate neighbors from all vertices already chosen (Figure 4c). It is important to note that if one vertex is in the neighborhood of more than one vertex already chosen, this does not make it more nor less likely to be picked. This process aims to be as random as possible in order to emulate the process of randomly picking one element from the universe of all s-sized connected induced subgraphs of G.



(a) One vertex chosen. (b) Two vertices chosen. (c) Three vertices chosen

Fig. 4: Example of the sampling process. Black vertices were chosen and gray vertices are the extended neighborhood.

We applied this sampling process to each one of the studied graphs, for s values of 5, 10, 15, 25, 30, 40, 50, 75, 100, 125, 150, 200, 250 and 300.

5.1.1 *Subgraph and Graph Sampling.* It is important to notice that what we are doing is different from the problem of graph sampling. We use a sample from the set of all possible subgraphs of the same size in order to identify the internal edge density from that set, based on the results obtained from the sample. Graph sampling, as discussed in works such as [Leskovec and Faloutsos 2006] , [Ribeiro and Towsley 2010] and [Clauset and Moore 2005], aims to discover subgraphs that present the same characteristics (such as degree distribution, clustering coefficient and diameter, for example)

[1] `http://www.stattrek.com`

as the original graph, only in a smaller scale. This kind of sample is used when the original graph is too large to be analyzed.

Graph sampling adds many restrictions to the sampling process, making simple random sampling inadequate to obtain a sample that can be considered valid, as can be seen in the already cited works. Fortunately, our situation is not as complex as graph sampling. For example, consider that the universe we want to sample is a cake with filling and frosting. For a simple random sampling, one could cut the cake into small cubes and randomly pick some of them. With a large enough sample, it would be possible to identify the proportion of each one of the components on the original cake. This kind of information is good enough for our problem, but would be inadequate for the problem of graph sampling, as it would not be possible to infer the structure of the cake, i.e., you could not guarantee that the frosting was indeed covering the cake. For graph sampling, a slice would be a better sample.

5.2 Graphs Used

We have studied graphs from different origins. Doing so allows us to evaluate different kinds of complex graph cluster structures. Table I shows the graphs we have used in our experiments. All graphs used were presented in other works in the literature and made available to the community by their authors.

Type	**Network**	**# Vertices**	**# Edges**
Social	Les Miserables	77	254
	Epinions	75879	508837
	Slashdot (11/2008)	77360	905468
	Slashdot (02/2009)	82168	948464
	General Relativity Collab.	5242	28980
	Condensed Matter Collab.	23133	186936
	Amazon Co-Purchasing	262111	1234877
Technological	Google Web Graph	875713	5105039
	College Football	115	616
	Power Grid	4941	6594
	Gnutella Snap. (31/08/02)	62586	147892
	Gnutella Snap. (30/08/02)	36682	88328
Biological	C. Elegans Neural Net.	297	2359
	Yeast	2361	7182

Table I: Datasets studied.

Those graphs can be roughly grouped in 3 different kinds of complex networks: social, biological and infrastructure (or technological). In social networks, edges represent relationships between people (or other social animals). The Les Miserables character interaction [Knuth 1993], the Epinions trust relationships, the Slashdot user discussions, the General Relativity and Condensed Matter scientific collaboration relationships and the Amazon.com product co-purchase[2] graphs are examples of this kind of network.

In biological networks, edges represent connections that happened between biological entities. Examples used here are the protein-protein interaction network in yeast [Bu et al. 2003] and the C. Elegans neural network [Watts and Strogatz 1998].

The edges in an infrastructure or technological network connect its vertices through an algorithmically defined process. Also, each edge has a real world cost, making them generally sparser then other kinds of complex networks. Examples of such networks are the World Wide Web site network (collected and published by Google)[2], the topology of the Western States Power Grid of the United States [Watts and Strogatz 1998] and the network of American football games between Division I-A colleges during regular season Fall 2000 [Girvan and Newman 2002].

[2] `http://snap.stanford.edu/snap/index.html`

6. EXPERIMENTS

This section will show the experiments we've made in order to test our proposed method for internal cluster density evaluation. We will discuss the results obtained and propose a simple and naive model as an example of how our method could be used for internal cluster density evaluation.

6.1 Results

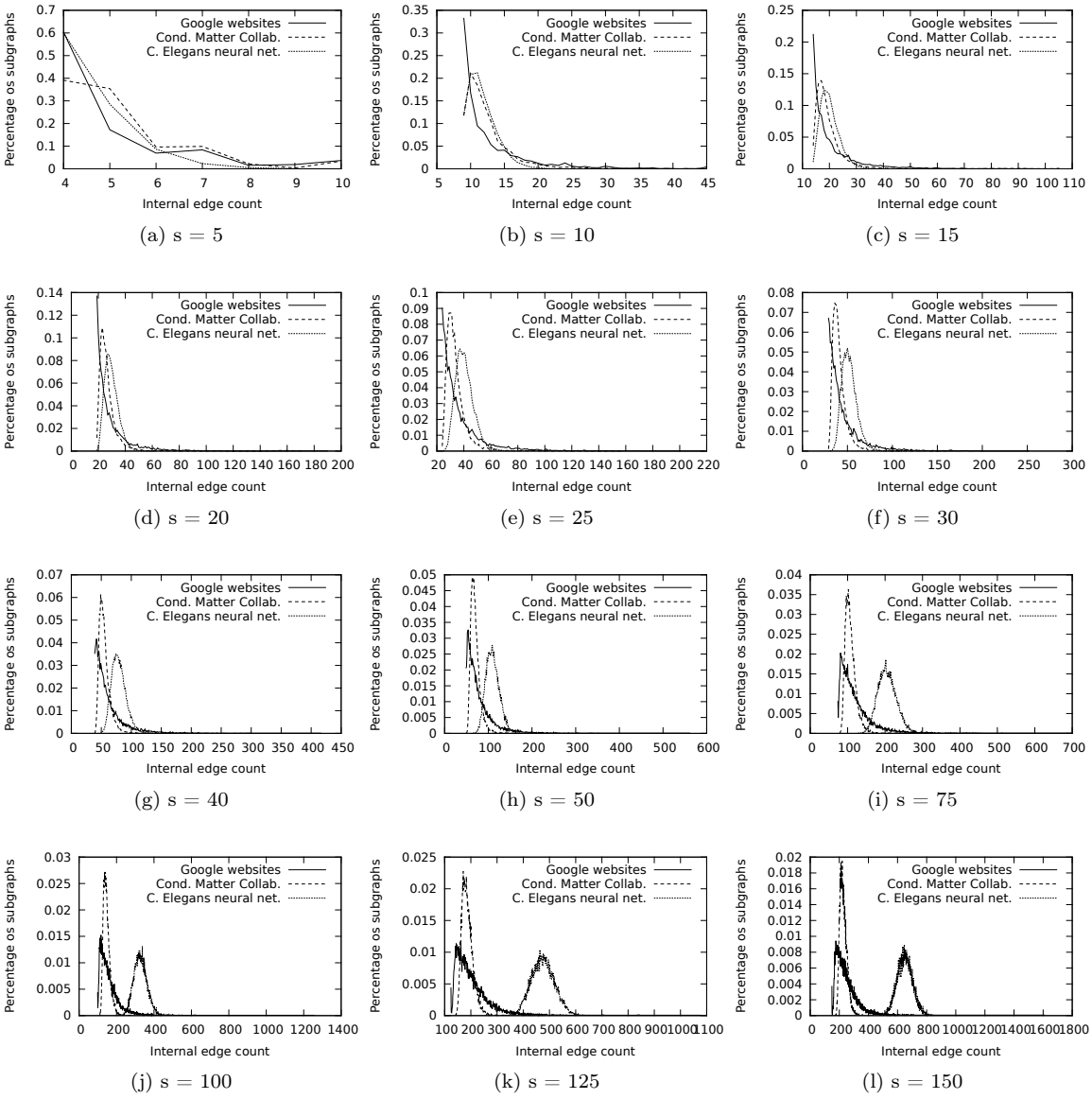
We now discuss the results from our experiments. For space reasons, not all results will be shown. Results for just one graph in each broad origin category will be shown and discussed. However, results for graphs in the same category show very similar behavior, so this poses no great problem. Also, although results for all subgraph sizes are not shown, the ones presented are enough to point out the most interesting tendencies in subgraph density behavior.

Figure 5 presents the density curves obtained from our samples for three networks: the condensed matter collaboration (social), the Google websites (technological) and the C. Elegans neural network (biological). It is possible to see that all three curves have very similar formats for the smallest of our sample sizes ($s = 5$). As s grows, however, the curves' behaviors begin to differ.

The Google websites' density curves have tree-like densities as their maxima for s values up to 30, something that is not surprising, given the natural sparsity of technological networks. Nonetheless, for all values of s sampled, even $s \leq 30$, the density curves show a heavy tail, with the highest edge count values far from their curves' maximum values. Those denser subgraphs (which are also denser than the average, as presented in Table II) show us that even sparser graphs, such as the ones from technological networks, may have components that are dense, even if only in its own context. It is also interesting to notice that, for $s > 30$, the density curves start to change shape, becoming more bell-like as their maximum and average values get farther from the minimum density possible. This process might mean many things: the subgraphs of this size might be agglomerations of smaller, tree-like subgraphs; or they might just become more similar to social or biological networks, which also have more bell-shaped curves. Unfortunately, since we are just evaluating one of the two main cluster characteristics, we can not yet say what this change in curve shape means in regard to cluster quality.

The density curves of the condensed matter collaboration network already show a tendency for a bell-like shape since values of $s > 5$. However, its curves have a sharper increase and a smoother decrease rate. That is interesting, because it means that the denser spectrum of observed densities, the ones we are most interested in, have a greater spread of possible values. This allows our kind of proposed evaluation to be more responsive. It is curious to notice that the bell-shape of the density curves from this network are slimmer than the curves from the other networks, which means that there is a large concentration of subgraphs with a significantly small difference in edge counts. This can also be seen on this network's sharply increasing Cumulative Distribution Function (CDF), as presented in Figure 6. This probably happens because of the origin of this network, as scientific collaborations groups might have a most common size of participants that most groups follow. Fortunately, a metric that uses the method we are proposing would use this kind of structural information and score clusters found here based on this particularities.

Compared to the other two, the C. Elegans neural network (biological) has the most normal-shaped density curves, which is slightly wider and shorter than the collaboration network discussed before. That means, and this is confirmed by its CDF (Figure 6), that subgraphs of the same size in this network tend to be a little more spread through different densities than the ones in the collaboration network. Another interesting thing to notice is that, as s grows, the density curve of this network becomes more right-shifted in relation to the curves from the other graphs, which could indicate that the larger clusters from this algorithm would be way denser than clusters of the same size from other networks. Nonetheless, if we pay attention to the size of this network (297 vertices) and the size of



(a) s = 5 (b) s = 10 (c) s = 15

(d) s = 20 (e) s = 25 (f) s = 30

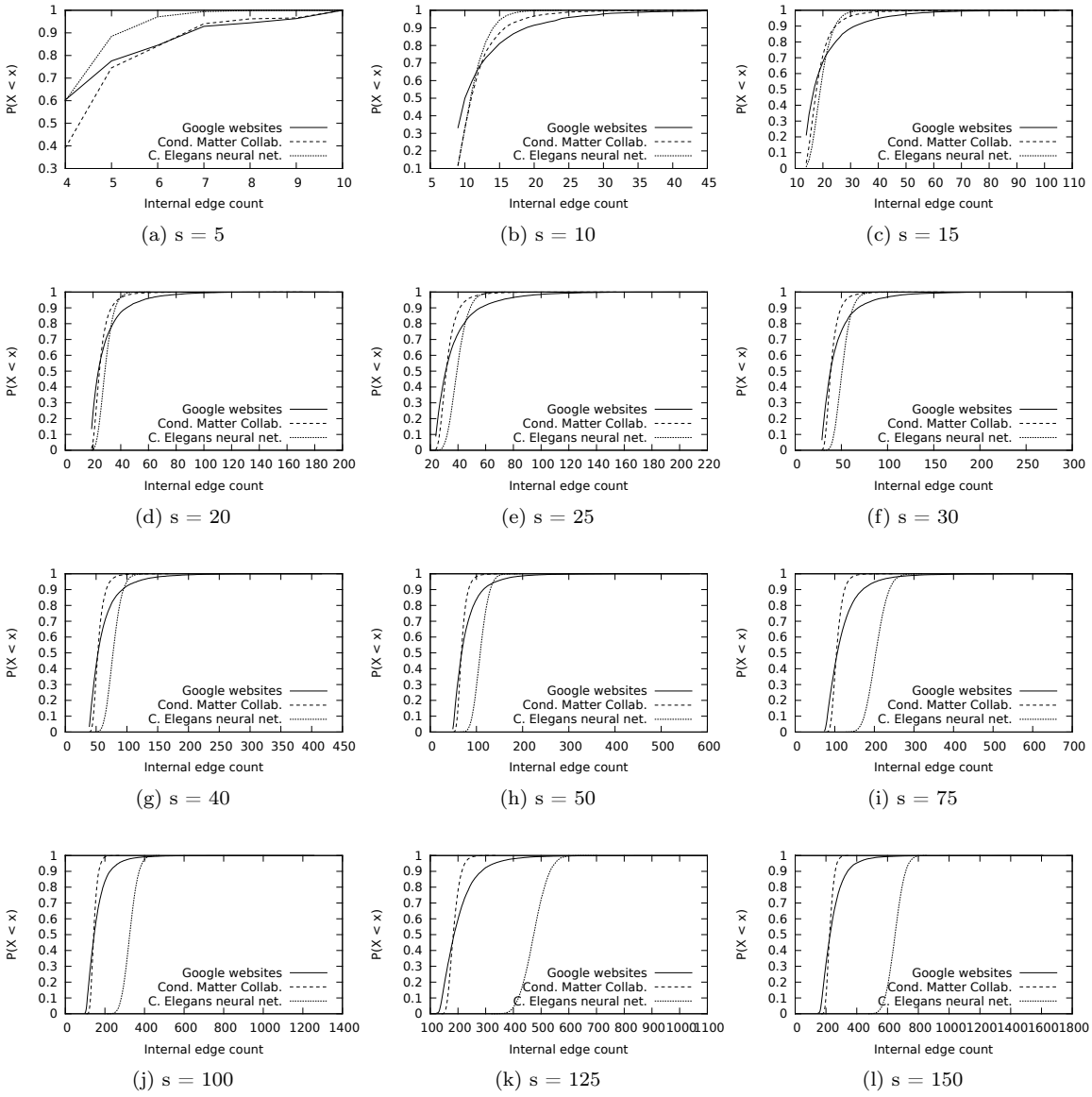
(g) s = 40 (h) s = 50 (i) s = 75

(j) s = 100 (k) s = 125 (l) s = 150

Fig. 5: Internal densities for different subgraph sizes - Condensed Matter collaboration, Google webpages and C. Elegans neural networks.

the larger s evaluated (150), it is possible to see that those subgraphs are almost half the size of the whole graph. Also, we couldn't make samplings larger than $s = 150$, since we could not pick enough random samples in order for the process to converge. So, it is highly probable that the size of universe of all subgraphs for larger values of s in this network is not as large as we assumed and, therefore, breaks our statistical guarantees. So, as cluster sizes approach the size of the graph, the proportion of cluster densities found becomes untrustworthy.

Also, it is interesting to notice that, as subgraph sizes grow, even the densest subgraph will be much sparser than a clique of the same size and, therefore, farther away from the canonical view of a dense cluster. For example, the largest 150-vertex subgraph sampled from the Condensed Matter collaboration network had 388 edges, which is equivalent to 3.47% of the edges of a same sized clique.



(a) s = 5 (b) s = 10 (c) s = 15

(d) s = 20 (e) s = 25 (f) s = 30

(g) s = 40 (h) s = 50 (i) s = 75

(j) s = 100 (k) s = 125 (l) s = 150

Fig. 6: Internal densities for different subgraph sizes - Condensed Matter collaboration, Google webpages and C. Elegans neural networks.

In comparison, largest 25-vertex subgraph sampled from the same graph had 132 edges, 44% of a clique of the same side.

6.2 Simple Model for Internal Density Evaluation

With those internal density distributions in mind, we should be able to define one model that implements the proposed method for internal density evaluation. We will now propose a very simple and naive example of such model.

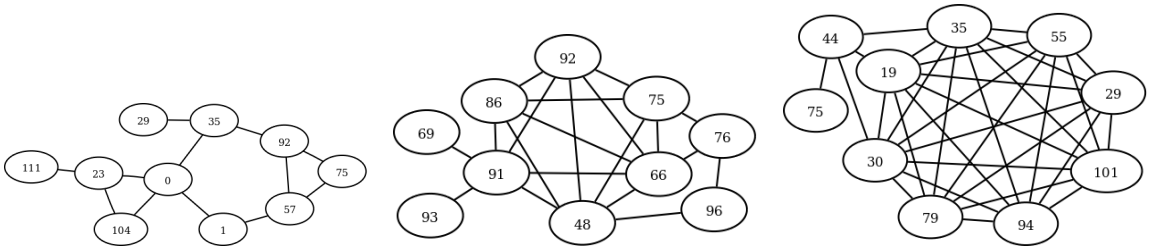
For this model, in order to evaluate the density of one given cluster of size s and e edges from a given graph, we will use the expected density for a subgraph of the same size in this same graph. To

	C. Elegans			Cond. Matter			Google webpages		
s	Avg.	SD	Median	Avg.	SD	Median	Avg.	SD	Median
5	4.551	0.968762	4	5.15331	1.41056	5	4.939	1.80163	4
10	11.7176	2.23384	11	12.3592	3.52315	11	12.7496	5.69013	11
15	20.1366	3.54596	20	19.5486	5.68677	18	20.9241	10.1242	17
20	29.7542	5.11063	29	26.7346	7.42872	25	29.1843	14.0542	24
25	40.5176	6.65246	40	33.7707	7.56748	32	37.4639	17.4538	32
30	52.1579	8.30064	51	40.8651	8.32429	39	45.7049	21.4857	39
40	79.0847	11.7994	78	55.298	9.75313	53	62.4682	28.7237	54
50	109.94	15.4519	109	70.0061	10.9282	68	78.9287	34.4551	69
75	205.707	24.9589	204	107.424	14.1964	105	119.905	46.6871	107
100	327.322	35.1251	326	146.622	17.5158	144	161.672	59.3562	146
125	475.018	44.1401	474	186.669	20.9741	183	204.363	69.4035	186
150	650.167	52.7181	650	227.666	24.1786	224	247.72	83.509	227
200				313.032	31.5184	309	332.867	101.615	309
250				401.726	38.3786	397	419.19	123.792	389
300				493.788	45.6575	489	509.185	144.865	475

Table II: Average, standard deviation and median for some of the studied graphs.

do so, we could use the percentile of value e in the internal density curve obtained from the sampling of size s clusters in the studied graph as the quality score for this cluster. This simple model would allow better scores for larger internal density values, with the 50^{th} percentile of the distribution being assumed to be a "neutral" density score. Greater percentiles would mean better scores, and lower percentiles, worse.

However, this schema would have poor discerning power to compare internal density values in the tail section of the curve. For example, Figure 7 presents 3 different clusters of the same size, but with different densities, obtained from the college football graph. If we use this simple percentile model, than the cluster in Figure 7a would be scored with 0.2598, the one from Figure 7b would receive 0.9623 and the one from Figure 7c would score 0.9999. The first cluster, which is almost as sparse as a tree, scored poorly, as expected. The two other graphs received high, but very approximate, scores. The problem here is that the third cluster has more than 50% more edges than the second one, and their scores should better reflect that disparity.



(a) 12 Edges (b) 20 Edges (c) 32 Edges

Fig. 7: Example of sampled size 10 subgraphs from the College Football graph.

One way to improve this model would be to create a two-phase scoring system. In such a system, we could shift the percentile value that would score as the "neutral" value of 0.5 to a higher value, so that the first scoring phase (lower than the shifted percentile) would cover more sparser density values, leaving less possible density values to be scored in the second phase of the model.

$$IDI = \begin{cases} \frac{P(X<x)}{2D} & if\ [P(X<x)] < D \\ \frac{[P(X<x)]-2D+1}{2D-2} & if [P(X<x)] \geq D \end{cases} \quad (7)$$

This simple model is presented in Eq. 7. It has a parameter D, which represents the percentile value that should be scored as "neutral". For example, if we assumed a value of $D = 0.75$, it would mean that the 75^{th} percentile would score 0.5 in our internal density index (IDI). This shift allows for an amplified descriptive power when evaluating the density values that matter the most: the ones in the tail section of the density curve. One description of how this scoring method would work for $D = 0.75$ can be seen in Figure 8, with the values of the IDI rising slowly until the defined percentile and faster afterwards.

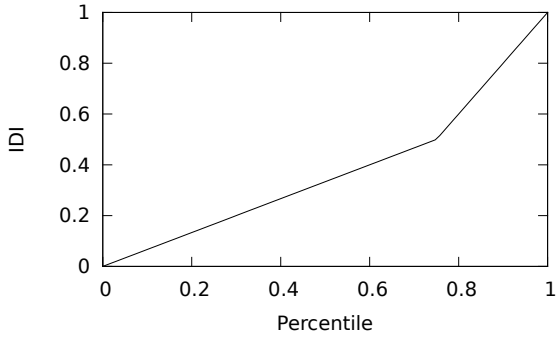


Fig. 8: Representation of the scoring method of the IDI naive model ($D = 0.75$).

For the same example in Figure 7, the first subgraph would be scored with 0.1732, the second with 0.9246 and the third with 0.9998. This simple adjustment penalized even further the sparser subgraph, while punctuating (even if slightly) the difference in density of the other two.

7. FUTURE WORK

The method proposed here is interesting, as it gives us a greater insight on the internal density of a cluster. However, it still presents some limitations that must be addressed in the future.

First, the model proposed in this article was very simple and done just as a proof of concept. Also, our method only evaluates half of the evidences for cluster quality. A better model that represents both our method for internal cluster density and an external sparsity index will be proposed in the future, in order to supply a complete evaluation of cluster quality.

Second, our method gives more specific results, but it comes with a cost: a relatively large sampling that has to be done for all cluster sizes of interest. At the moment, we are working on ways of using a graph's structural characteristics to derive a probability density model that can give us the same kind of answer without sampling. It is our expectation that such relations can be found for graphs with the same origin, based on the observation of our results so far. Nevertheless, a wider analysis will be necessary to find a derivation that fits each case.

8. CONCLUSION

In this article, we have studied the reason that causes some very popular and widely used graph cluster quality metrics to present undesired biases in their scores. We observed that those quality metrics did not correctly evaluate the clusters' internal density, one of the two key aspects that define a well-formed cluster. Also, they used the same standards to evaluate clusters from graphs with wildly different structural characteristics, such as social and technological ones. This causes clusterings from naturally sparser graphs to be severely penalized in their quality scores.

To solve this problem, we proposed a novel method to evaluate internal cluster density. To evaluate a cluster, it uses the expected edge count from subgraphs of the same size as the cluster, induced from the graph being considered. This way, not only are we using both vertex and edge information to evaluate internal density, but we are also using the graph itself to determine the threshold that identifies what is dense and sparse in the context at hand. We also presented a naive model as an example of how our method could be used to evaluate a cluster's internal density. Results are promissing, and we hope new ways to apply this method will be devised in the near future.

REFERENCES

Almeida, H., Guedes, D., Meira, W., and Zaki, M. J. Is there a best quality metric for graph clusters? In *Proceedings of the 2011 European Conference on Machine Learning and Knowledge Discovery in Databases - Volume Part I*. Athens, Greece, pp. 44–59, 2011.

Brandes, U., Gaertler, M., and Wagner, D. Experiments on graph clustering algorithms. In *Proceedings of the 11th Annual European Symposium on Algorithms*. Budapest, Hungary, pp. 568–579, 2003.

Brandes, U., Gaertler, M., and Wagner, D. Engineering graph clustering: Models and experimental evaluation. *Journal of Experimental Algorithmics* 12 (1): 1–26, 2008.

Bu, D., Zhao, Y., Cai, L., Xue, H., Zhu, X., Lu, H., Zhang, J., Sun, S., Ling, L., Zhang, N., Li, G., and Chen, R. Topological structure analysis of the protein–protein interaction network in budding yeast. *Nucleic Acids Research* 31 (9): 2443–2450, 2003.

Clauset, A. and Moore, C. Accuracy and scaling phenomena in internet mapping. *Physical Review Letters* 94 (1): 018701, 2005.

Danon, L., Díaz-Guilera, A., Duch, J., and Arenas, A. Comparing community structure identification. *Journal of Statistical Mechanics: Theory and Experiment* 2005 (09): P09008, 2005.

Du, N., Wu, B., Pei, X., Wang, B., and Xu, L. Community detection in large-scale social networks. In *Proceedings of the 9th WebKDD and 1st SNA-KDD Workshop on Web Mining and Social Network Analysis*. San Jose, California, pp. 16–25, 2007.

Gargi, U., Lu, W., Mirrokni, V. S., and Yoon, S. Large-scale community detection on youtube for topic discovery and exploration. In *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media*. Barcelona, Spain, pp. 486–489, 2011.

Girvan, M. and Newman, M. E. J. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences of the United States of America* 99 (12): 7821–7826, 2002.

Gustafson, M., Hörnquist, M., and Lombardi, A. Comparison and validation of community structures in complex networks. *Physica A: Statistical Mechanics and its Application* 367 (1): 559–576, 2006.

Kannan, R., Vempala, S., and Vetta, A. On clusterings: Good, bad and spectral. *Journal of the ACM* 51 (3): 497–515, 2004.

Knuth, D. E. *The Stanford GraphBase: a platform for combinatorial computing*. ACM, New York, NY, USA, 1993.

Leskovec, J. and Faloutsos, C. Sampling from large graphs. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge discovery and data mining*. Philadelphia, USA, pp. 631–636, 2006.

Leskovec, J., Lang, K. J., Dasgupta, A., and Mahoney, M. W. Statistical properties of community structure in large social and information networks. In *Proceeding of the 17th International Conference on World Wide Web*. Beijing, China, pp. 695–704, 2008.

Leskovec, J., Lang, K. J., and Mahoney, M. Empirical comparison of algorithms for network community detection. In *Proceedings of the 19th international conference on World wide web*. Raleigh, USA, pp. 631–640, 2010.

Lohr, S. L. *Sampling: Design and Analysis*. Brooks/Cole, Boston, USA, 2010.

Newman, M. E. and Girvan, M. Finding and evaluating community structure in networks. *Physical Review E* 69 (2): 026113, 2004.

Newman, M. E. J. Mixing patterns in networks. *Physical Review E* 67 (2): 026126, 2003.

Newman, M. E. J. and Girvan, M. Mixing Patterns and Community Structure in Networks. *Statistical Mechanics of Complex Networks* 625 (1): 66–87, 2003.

Qi, G.-J., Aggarwal, C. C., and Huang, T. S. On clustering heterogeneous social media objects with outlier links. In *Proceedings of the 5th ACM International Conference on Web Search and Data Mining*. Seattle, USA, pp. 553–562, 2012.

Ribeiro, B. and Towsley, D. Estimating and sampling graphs with multidimensional random walks. In *Proceedings of the 10th Annual Conference on Internet Measurement*. Melbourne, Australia, pp. 390–403, 2010.

Satuluri, V. and Parthasarathy, S. Scalable graph clustering using stochastic flows: applications to community discovery. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Paris, France, pp. 737–746, 2009.

Schaeffer, S. E. Graph clustering. *Computer Science Review* 1 (1): 27–64, 2007.

Steinbach, M., Karypis, G., and Kumar, V. A comparison of document clustering techniques. In *Proceedings of the Workshop on Text Mining*. Boston, MA, pp. 109–111, 2000.

Tan, P.-N., Steinbach, M., and Kumar, V. *Introduction to Data Mining*. Addison-Wesley Longman Publishing Co., Inc., Boston, USA, 2005.

van Dongen, S. *Graph Clustering by Flow Simulation*. Ph.D. thesis, University of Utrecht, The Netherlands, 2000.

van Dongen, S. Graph clustering via a discrete uncoupling process. *SIAM Journal on Matrix Analysis and Applications* 30 (1): 121–141, 2008.

Watts, D. J. and Strogatz, S. H. Collective dynamics of 'small-world' networks. *Nature* 393 (6684): 440–442, 1998.

Zaidi, F., Archambault, D., and Melançon, G. Evaluating the quality of clustering algorithms using cluster path lengths. In *Proceedings of the 10th Industrial Conference on Advances in Data Mining: Applications and Theoretical Aspects*. Berlin, Germany, pp. 42–56, 2010.

Zhou, Y., Cheng, H., and Yu, J. X. Graph clustering based on structural/attribute similarities. *Proceedings of the VLDB Endowment* 2 (1): 718–729, 2009.