Is There a Best Quality Metric for Graph Clusters?

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

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

Abstract. Graph clustering, the process of discovering groups of similar vertices in a graph, is a very interesting area of study, with applications in many different scenarios. 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 evaluation metrics for graph clustering have been proposed in the literature, but there is no consensus on how do they compare to each other and how well they perform on different kinds of graphs. In this work we study five major graph clustering quality metrics in terms of their formal biases and their behavior when applied to clusters found by four implementations of classic graph clustering algorithms on five large, real world graphs. Our results show that those popular quality metrics have strong biases toward incorrectly awarding good scores to some kinds of clusters, especially seen in larger networks. They also indicate that currently used clustering algorithms and quality metrics do not behave as expected when cluster structures are different from the more traditional, clique-like ones.

1 Introduction

The problem of graph clustering consists of discovering natural groups (or clusters) in a graph [17]. Graph clustering has become very popular recently, given the large number of applications it has in areas like social network analysis (finding groups of related people), e-commerce (doing recommendations based on relations in a group) and bioinformatics (classifying gene expression data, studying the spread of a disease in a population).

What is the basic structure of a group is open to discussion, but the most classical and adopted view is the one based on the concept of homophily: similar elements have a greater tendency to group with each other than with other elements [16]. When working with graphs, homophily is usually viewed in terms of edge densities, with clusters having more edges linking their elements among themselves (high internal density) than linking them to the rest of the graph

D. Gunopulos et al. (Eds.): ECML PKDD 2011, Part I, LNAI 6911, pp. 44–59, 2011.

(sparser external connections). However, discovering those 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.

When a graph has only a small number of vertices, the result of its clustering can be evaluated manually. However, as the size of the graph grows, manual evaluation becomes unfeasible. For those cases, evaluation metrics, such as modularity [14] and conductance [8], which try to encapsulate the most important characteristics expected from a good cluster, may be used as indicative of cluster quality. The quality of a clustering algorithm is then estimated in terms of the values its output gets for that metric. That allows researchers to compare different proposed clusterings to each other in order to find the best one among them.

Still, as much as those metrics try to identify good clusters by quantifying the quality of a grouping, they are not perfect. It is actually very difficult to determine whether a given quality metric gives the expected answer for a given clustering of a graph, since typically there is no ground truth for comparison. That is particularly true for larger datasets derived from the relationships of real groups.

In this work we compare some of the most popular quality metrics for graph clustering. We evaluate if those metrics really represent the classical view of what a cluster should be and how do they behave in larger, less predictable cases. Our results show that the currently used quality indexes for graph clustering have a series of structural anomalies that cause them to be biased and unreliable in bigger, real world graphs. Also, we observed that the type of network (social, technological, etc) can cause the structure of its clusters to become different from what is typically expected by the clustering algorithms and quality metrics studied.

2 Related Work

Many papers have focused on the problem of comparing the effectiveness of clustering algorithms. To do so, they use different cluster quality evaluation metrics. The problem is that, in general, authors just assume those metrics are good enough to represent good clusters, without concerning themselves with the evaluation of such claims. Most of the related work presented here falls into this category.

Brandes et al. [1] compared three algorithms for graph clustering: Markov clustering, iterative conductance cut and geometric MST clustering. To evaluate their results, they used three validation metrics, namely, conductance, coverage and performance. Their experiments used synthetic clusters with 1000 nodes. However, the goal of that work was to compare the results from the three algorithms, assuming that the validation metrics used represented the ground truth for the "real" clustering of the graphs. Because of that, they do not compare metrics against each other. Our work differs from theirs in that we want to know whether those validation metrics in fact identify a good clustering correctly.

Gustafson and Lombardi [7] compare K-means and hierarchical clustering using modularity and silhouette index as quality metrics. They use Zachary's karate club, the American college football, the gene network of yeast and synthetic graphs as the datasets for their work. Once again, the focus of their work is on the clustering algorithms, while ours is on the quality of the validation metrics.

Danon et al. [2] compare many different algorithms for graph clustering, including agglomerative, divisive and modularity maximization techniques. However, the only quality metric used to compare those algorithms is modularity. They use very small (128 vertices) synthetic datasets for evaluation.

One paper by Good et al. [6] discusses the effectiveness of modularity maximization methods for graph clustering. This is interesting because, in a way, they are evaluating how good modularity is as a quality index. However, they use a different formula to calculate modularity than the one we use in this paper, one that clearly generates unbounded scores and is, therefore, inadequate as a validation index. Unbounded scores can be used to compare clusterings, but are of no use to evaluate the quality of a single cluster in terms of its own structure, since there are no upper or lower bounds of metric values to compare its result to.

Another work, by Leskovec et al. [10], uses external conductance to evaluate the quality of clusters in terms of their size, to determine if there is a maximum or expected size for well formed clusters. The problems with this approach is that, similarly to other works discussed, it assumes that conductance is the best quality index to evaluate said clusters. In a follow-up work, Leskovec at al. [11] use other quality metrics to evaluate the same problem, but those new metrics also focus only on the external sparsity of the clusters.

Tan et al. [20] present a very interesting comparison between many metrics used to determine the "interestingness" of association metrics like lift, confidence and support, widely used in data mining. They show that there is no single metric that is consistently better than the others in different scenarios, so that the metrics should be chosen case-by-case to fit the expectations of the domain experts. Our work does a similar comparison for graph clustering validation metrics.

3 Quality Metrics

The most accepted notion of a cluster is based on the concept of assortative mixing: elements have a greater tendency to form bonds with other elements with whom they share common traits than with others [15]. 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 or vertex characteristics, such as edge density [5], vertex distance [21] or labels [24].

In this section we will present some of the most popular cluster quality metrics present in the literature. 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 paper use only a graph's topological information, like vertex distance or edge density, to evaluate the quality of a given cluster.

3.1 Graph Definitions

A *graph* $G = (V, E)$ is composed of a set V of *vertices* and a set $E = (u, v)|u, v \in V$ of *edges*. If nothing is said in opposition, assume that the graphs discussed are undirected, so that $(u, v) = (v, u)$. The number of edges of a graph G is $|E(G)| = m$, and the number of edges linked to a given vertex v is represented as $deg(v)$. Edges may have an associated weight $w(u, v)$. In unweighted cases, we assume that $w(u, v) = 1$ for all $(u, v) \in E$. Also, consider $E(C_i, C_j)|i \neq j$ as the set of edges linking clusters C_i and C_j and $E(C_i)$ as the set of edges $(u, v)|u, v \in C_i$. Then, $E(C)$ is the set of all internal edges for all clusters in C, and $\bar{E}(C)$ is the set of all *inter-cluster edges* in the graph $((u, v)|u \in C_i, v \in C_j, i \neq j)$.

A *clustering* C is the set of all *clusters* of a graph, so that $C = C_1, C_2, \ldots, C_k$, and the number k of clusters may be a parameter of some clustering algorithms. Also, unless stated otherwise, $C_i \cap C_j = \varnothing, \forall i \neq j$. A cluster C_i that is composed by only one vertex is called a *singleton*. The weight of all internal edges of a single cluster is given by $w(C_i)$, a shortcut for $\sum_{e \in E(C_i)} w(e)$. By the same logic, $\bar{w}(C)$ is the sum of the weights of all inter-cluster edges.

A graph cut $K = (S, \bar{S})$, where $\bar{S} = V \setminus S)$, divides a set of vertices V into two disjoint groups $(S \cap \bar{S} = \varnothing)$. The *cost* of a cut is given by the sum of the weights of the inter-cluster edges. Another important concept is that of an *induced graph*, which is a graph formed by a subset of the vertices and edges of a graph so that $G[C_i] = (C_i, E(C_i))$.

3.2 Modularity

One of the most popular validation metrics for topological clustering, modularity states that a good cluster should have a bigger 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 [14]. 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 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.

3.3 Silhouette Index

This metric uses concepts of cohesion and separation to evaluate clusters, using the distance between nodes to measure their similarity [21]. The silhouette index for a given vertex i is given by Equation 2

$$S(C_i) = \frac{\sum_{v \in C_i} S_v}{|C_i|} \ , \ where \ \ S_v = \frac{b_v - a_v}{max(a_v, b_v)} \tag{2}$$

Where a_v is the average distance between vertex v and all the other vertices in the same cluster as it is, and b_v is the average distance between v and all the vertices in the nearest cluster that is not v's. The silhouette index for a given cluster is the average value of silhouette for all its member vertices. The silhouette index can assume values between -1 and 1, with a negative value being undesirable, as it means that the average internal distance of the cluster is greater than the external one.

The silhouette index presents some limitations, though. First of all, it is a very expensive metric to calculate, requiring an all pairs shortest path execution. The other is how it behaves in the presence of singleton clusters. Since a singleton possesses no internal edges, its internal distance will be 0, causing its silhouette to wrongly score a perfect 1. This way, clusterings with many singletons will always have high silhouette scores, no matter the quality of the other clusters.

3.4 Conductance

The conductance [8] of a cut is a metric that compares the size of a cut (i. e., the number of edges cut) and the weight of the edges in either of the two subgraphs induced by that cut. The conductance $\phi(G)$ of a graph is the minimum conductance value between all its clusters.

Consider a cut that divides G into k non-overlapping clusters $C_1, C_2 \ldots C_k$. The conductance of any given cluster $\phi(C_i)$ can be obtained as shown in Equation 3, 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 $V \setminus C_i$. Since we want to find a number k of clusters, we will need $k - 1$ cuts to achieve that number. In this paper we assume the conductance for the whole clustering to be the average value of those $(k - 1)$ ϕ cuts, as formalized in Equation 4.

$$\phi(C_i) = \frac{\sum_{u \in C_i} \sum_{v \notin C_i} w(\{u, v\})}{min(a(C_i), a(\bar{C_i}))} \tag{3}$$

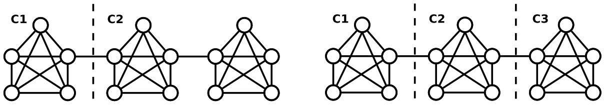
$$\phi(G) = avg(\phi(C_i)) \ , \ C_i \subseteq V \tag{4}$$

Based on this information, it is possible to define the concept of intra-cluster conductance $\alpha(C)$ (Eq. 5) and the inter-cluster conductance $\sigma(C)$ (Eq. 6) for a given clustering $C = C_1, C_2, \ldots, C_k$.

$$\alpha(C) = min_{i \in \{1,\ldots,k\}} \phi(G[C_i]) \tag{5}$$

$$\sigma(C) = 1 - max_{i \in \{1,\ldots,k\}} \phi(C_i) \tag{6}$$

The intra-cluster conductance will be the minimum conductance value of the graphs induced by each cluster C_i, with a low value meaning that at least one of the clusters may be too coarse to be good. The inter-cluster conductance is the complement of the maximum conductance value of the clustering, so that lower values might show that at least one of the clusters have strong connections outside of it, i. e., the clustering might be too fine. So, a good clustering should have high values of both intra- and inter-cluster conductance.



(a) Two clusters (b) Three clusters

Fig. 1. Two possible clusterings of a same graph

Although the use of both internal and external conductance gives a better, well rounded view of both internal density and external sparsity of a cluster, many works use only the external conductance while evaluating cluster quality [10,11]. So, in this paper we will likewise use only the external conductance, referred from now on simply as *conductance*, to evaluate if it is a good enough quality metric by itself. One negative characteristic of conductance that can be pointed out is that it might have a tendency of giving better scores to clusterings with fewer clusters, as more clusters will probably have more cut-edges. Also, the lack of internal edge density information used in this kind of conductance may cause problems, as can be seen in Figure 1, where both clusterings presented would have the same conductance score, even though the one in Figure 1b is obviously better.

3.5 Coverage

The coverage of a clustering C (where $C = C_1, C_2, \ldots, C_k$) is given as the fraction of the weight of all intra-cluster edges with respect to the total weight of all edges in the whole graph G [1], as shown in Equation 7:

$$coverage(C) = \frac{w(C)}{w(G)}, where \tag{7}$$

$$w(C) = \sum_{i=1}^{k} w(E(v_x, v_y));\ v_x, v_y \in C_i$$

Coverage values usually range from 0 to 1. Higher values of coverage mean that there are more edges inside the clusters than edges linking different clusters, which translates to a better clustering. From its formulation, we can observe that the main clustering characteristic needed for a high value of coverage is inter-cluster sparsity. Internal cluster density is in no way taken into account by this metric, and it probably causes a strong bias toward clusterings with less clusters. This can be seen in the example on Figure 1, where the clustering with two clusters would receive a better score than the clearly better clustering with three clusters.

3.6 Performance

This metric counts the number of internal edges in a cluster along with the edges that don't exist between the cluster's nodes and other nodes in the graph [22], as can be seen in Equation 8

$$\begin{aligned} perf(C) &= \frac{f(C)+g(C)}{\frac{1}{2}n(n-1)}, where \qquad (8)\\ f(C) &= \sum_{i=1}^{k} |E(C_i)| \\ g(C) &= \sum_{i=1}^{k} \sum_{j>i} |\ \{\{u,v\} \notin E | u \in C_i,\ v \in C_j\}| \end{aligned}$$

This formulation assumes an unweighted graph, but there are also variants for weighted graphs [1]. Values range from 0 to 1, and higher values indicate that a cluster is both internally dense and externally sparse and, therefore, a better cluster. However, if we consider that complex networks tend to be sparse in nature, when performance is applied to larger graphs, there is a great possibility that $g(C)$ becomes so high that it will dominate all other factors in its formula, awarding high scores indiscriminately.

4 Clustering Algorithms

To be able to compare different clusterings with the validation metrics available, we selected representatives from four different, representative categories of clustering algorithms. The chosen algorithms were Markov clustering, bisection K-means, spectral clustering and normalized cut.

4.1 Markov Clustering

The Markov clustering algorithm (MCL) [22,4] is based on the simulation of stochastic flows in a graph. The basic idea behind MCL is that the distances between vertices are what identify a cluster, with small distances between vertices indicating that they should belong to the same cluster and large distances meaning the opposite. By that logic, a random walker would have greater probability to stay inside a cluster than to wander to neighboring ones, and the algorithm explores that to identify clusters.

The clustering process of MCL consists of two iterative steps: expansion and inflation. The expansion step of the algorithm is done taking the power of the normalized adjacency matrix representing the graph using traditional matrix multiplication. The inflation step consists in taking the Hadamard power of the expanded matrix, followed by a scaling step to make the matrix stochastic again, with the elements of each column corresponding to a probability value. MCL does not need to have a pre-defined number of clusters as input, it's only parameter being the inflation value, which affects the coarsening of the graph (the lower the value, the coarser the clustering).

4.2 Bisecting K-means

In the traditional K-means algorithm, k elements are chosen as the centroids of each one of the k clusters to be found and other elements closer to a given centroid than to others are added to that cluster. With this basic cluster in hand, a new centroid is calculated for each cluster, reflecting their new "centers", and the process is repeated until the centroids calculated do not change anymore.

Bisecting K-means [19] differs from the traditional algorithm in the following way: the whole graph is considered to be a cluster, which we bisect using traditional K-means, the topological distance between the nodes acting as the vertex similarity function. One of the new clusters is chosen to be once more bisected and the process repeats until the desired number of clusters is found.

4.3 Spectral Clustering

Spectral clustering [8,17] is a technique that uses the eigenvectors (spectrum) and eigenvalues of a matrix to define cluster membership. It is based on the fact that if a graph is formed by k disjoint cliques, then it's normalized Laplacian will be a block-diagonal matrix with eigenvalue of zero and multiplicity k. Also, its eigenvectors function as indicators of cluster membership. More than that, small perturbations like adding a few edges linking clusters or removing edges from inside the clusters will make the eigenvalues become slightly higher than zero and change its eigenvectors, but not enough to cause the underlying structure to be lost. This clustering technique requires the number of desired clusters as an input.

4.4 Normalized Cut

This method, proposed By Shi and Malik [18], tries to find the best possible clustering through the optimization of an objective function, in this case, a cut. Consider the cost of $cut(A, B)$, that divides the vertices V of a graph $G = (V, E)$ in two sets $A, B | A \cup B = V, A \cap B = \emptyset$, as the sum of the weights of all edges linking vertices in A to vertices in B. We want to find the cut that minimizes the cost function given by Equation 9, where the volume of a set is the sum of the weights of all edges with at least one endpoint inside it.

$$cut(A, B) = \left(\frac{1}{Vol(A)} + \frac{1}{Vol(B)} \right) \tag{9}$$

This cost function is designed to penalize cuts that generate subsets with highly different sizes. So, by minimizing the normalized cut of a graph, we are dividing sets of vertices with low similarity and that potentially have high internal similarity. This technique also requires the desired number of clusters to be given as an input.

5 Experiments

This section presents the experiments used to help evaluating the quality metrics studied. We will briefly describe our methodology and graphs used, following with a discussion of the obtained results.

5.1 Methodology

We implemented the five quality metrics discussed in Section 3. To evaluate their behavior, we applied them to clusters obtained through the execution of the four classical graph clustering algorithms discussed in Section 4 on five large, real world graphs that will be briefly discussed in the next subsection. This variety of clustering algorithms and graphs is necessary to minimize the pollution of the results by possible correlations between metrics algorithms and/or graph structures.

We used freely available implementations for all clustering algorithms: the *MCL* implementation by Van Dongen, which is available within many Linux distributions, the implementation of bisecting K-means available in the Cluto[1] suite of clustering algorithms, the spectral clustering algorithm implementation available in SCPS, by Nepusz [13] and the normalized cut clustering implementation GRACLUS, by Dhillon [3].

Three different inflation indexes where chosen for the MCL algorithm, based on the values suggested by the algorithm's documentation: 1.5, 2, and 3. The number of clusters found by each MCL configuration was used as the input for the other algorithms, so that we could compare clusterings with roughly the same number of clusters.

Graphs. We used 7 different datasets derived from real complex networks. Two of them are smaller, but with known expected partitions that could be used for comparison, and the other five are bigger and with unknown expected partitions. All graphs used are undirected and unweighted.

[1] `http://glaros.dtc.umn.edu/gkhome/cluto/cluto/overview`

The first small dataset is the Karate club network. It was first presented by Zachary [23] and depicts the relationships between the students in a karate dojo. During Zachary's study a fight between two teachers caused a division of the dojo in two, with the students more related to one teacher moving to his new dojo. Even though this dataset is small (34 vertices), it is interesting to consider because it possesses information about the real social partition of the graph, providing a ground truth for the clustering.

The other small dataset used was the American College football team's matches [5]. It represents a graph where the vertices are football teams and an edge links two teams if they have played against each together. Since the teams play mostly with other teams in the same league as theirs, with the exception of some military school teams, which belong to no league and can play against anyone, there is also an expected clustering already known for this graph. It is composed of 115 vertices and 616 edges.

The five remaining networks were obtained from the Stanford Large Network Dataset Collection[2]. Two of them represent the network of collaborations in papers submitted to the arXiv e-prints in two different areas of study, namely Astrophysics and High Energy Physics. In those networks, researchers are the vertices, and they are linked by edges if they collaborated in at least one paper. The Astrophysics network is composed by 18,772 vertices and 396,160 edges, while the High Energy Physics has 12,008 vertices and 237,010 edges. Another network based on the papers submitted to the arXiv e-prints was used, but covering the citation network of authors in the High Energy Physics category. In this case, an edge links two authors if one cites the other. This network has 34,546 vertices and 421,578 edges.

The last two networks are snapshots from a Gnutella P2P file sharing network, taken in two different dates. Here the vertices are the Gnutella clients and the edges are the overlay network connections between them. The first snapshot was collected in August, 4 2002 and comprises 10,876 vertices and 39,994 edges. The second one was collected in August, 30 2002 and has 36,682 vertices and 88,328 edges.

5.2 Results

We first used the smaller datasets, the karate club and the college football, in order to check how the algorithms and quality metrics behaved in small networks where the expected result was already known. The results for the Karate club dataset can be seen on Table 1. The College Football dataset gave similar results and was omitted for brevity. The results shown represent the case with two clusters, which is the expected number for this dataset. It can be observed that the scores obtained were fairly high. Also, the resulting clusters were very similar to the expected ones, with variations of 2 or 3 wrongly clustered vertices. However, those two study cases were very small and classical, so good results

[2] `http://snap.stanford.edu/data/`

Table 1. Karate Club dataset and its quality indexes for two clusters

Algorithm	SI	Mod	Cov	Perf	Cond
MCL	0.13 ± 0.02	0.29	0.71	0.55	0.55 ± 0.15
B. k-means	0.081 ± 0.001	0.37	0.87	0.62	0.26 ± 0.13
Spectral	0.13 ± 0.02	0.36	0.87	0.61	0.30 ± 0.15
Norm. Cut	0.14 ± 0.017	0.18	0.68	0.56	0.65 ± 0.32

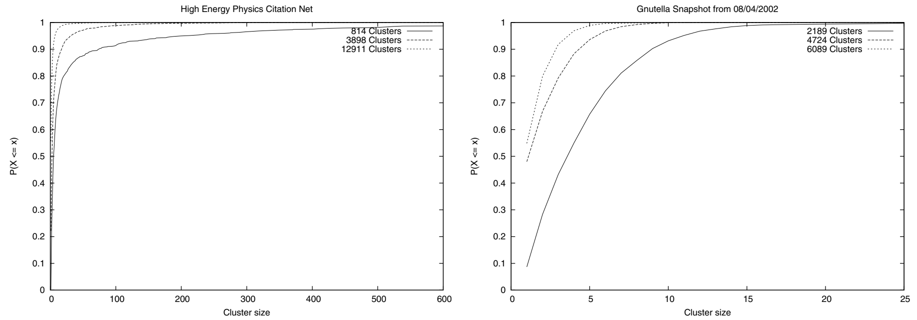
here were more than expected, as most of the quality metric biases we pointed out in Section 3 were connected to bigger networks with many clusters.

Now, for the larger datasets. The quality metric values for the Astrophysics Collaboration network are available in Table 2. It's already possible to observe some trends on the quality metrics' behavior, no matter what clustering algorithm is used. For example, modularity, coverage and conductance always give better results for smaller numbers of clusters. Also, we can see that, as expected from our observations in Section 3, performance values have no discriminating power to compare any of our results. The silhouette index presents a somewhat erratic behavior in this case, without a clear tendency of better or worse results for more or less clusters.

Table 2. Astrophysics collaboration network clusters and their quality indexes

Algorithm	# Clusters	SI	Mod.	Cover.	Perf.	Cond.
MCL	1036	-0.22 ± 0.038	0.35	0.42	0.99	0.55 ± 0.02
MCL	2231	-0.23 ± 0.026	0.28	0.31	0.99	0.70 ± 0.006
MCL	4093	0.06 ± 0.015	0.19	0.27	0.99	0.82 ± 0.003
B. k-means	1037	-0.73 ± 0.017	0.25	0.28	0.99	0.70 ± 0.002
B. k-means	2232	-0.48 ± 0.005	0.21	0.24	0.99	0.70 ± 0.002
B. k-means	4094	-0.21 ± 0.01	0.17	0.19	0.99	0.76 ± 0.001
Spectral	1034	-0.15 ± 0.036	0.34	0.38	0.99	0.53 ± 0.015
Spectral	2131	-0.26 ± 0.027	0.25	0.28	0.99	0.66 ± 0.007
Spectral	3335	0.04 ± 0.017	0.19	0.21	0.99	0.78 ± 0.004
Norm. Cut	1037	-0.69 ± 0.021	0.23	0.25	0.99	0.66 ± 0.006
Norm. Cut	2232	-0.51 ± 0.019	0.17	0.19	0.99	0.73 ± 0.015
Norm. Cut	4094	-0.31 ± 0.006	0.13	0.15	0.99	0.81 ± 0.0004

For the High Energy Physics Collaboration network, as we can see on Table 3, the tendencies observed in the last network are still true. Also, silhouette index shows a more pronounced bias toward larger numbers of clusters. If we look at the cumulative distribution function (CDF) of cluster sizes (as shown in Figure 2 for just two instances of our experiments, but that is consistent with the rest of the obtained results), we can see that bigger clusterings tend to have a larger number of smaller clusters. So, this bias of the silhouette index is expected from our observations in Section 3. Those same tendencies occur in the High Energy Physics Citation network, as seen in Table 4.



(a) High Energy Physics Citation (b) Gnutella Snapshot 08/04/2002

Fig. 2. Some cluster size's Cumulative Distribution Functions (bisecting k-means)

The quality metric scores for one of the Gnutella snapshot networks can be seen in Table 5. The scores for the other one were very similar to it, so we suppressed them for brevity. It is possible to notice that the results for those graphs still present the same tendencies shown in the other cases, but with a key difference: while silhouette and performance results show no big difference from the other datasets, as they are easily fooled by high numbers of singleton clusters and network size, respectively, modularity, coverage and conductance give abysmally low quality results. This happens because the structure of a Gnutella network, with common peers connected only to "superpeers", and those superpeers also connected with each other. This structure leads to a very low occurrence probability of 3-cliques (0.5% for the Gnutella networks against 31.8% for the Astrophysics Collaboration network, for example). Also, the Gnutella networks presented here are way sparser than the other studied networks, with only 6.76% of all possible edges present in the graph for the 08/04/2002 snapshot against 32.88% for the High Energy Physics citation one, for example.

Discussion. For all the generated cases, coverage, modularity and conductance have better values for smaller numbers of clusters. This behavior is expected from the formulation of coverage, since it observes the number of inter-cluster edges, which tends to be smaller if there are less clusters to link to. The same thing happens to conductance, as more inter-cluster edges mean more expensive cuts. Without balancing the external conductance with the internal conductance, results will only give us a partial and biased results.

Concerning modularity, we already know that singleton clusters have a very bad impact on the modularity score, and the more the clusters, the bigger the chance for singletons to occur. It is interesting to notice that giving low scores to singleton clusters is not wrong *per se*, but since those scores will influence in the overall score, they can obfuscate the existence of well scored clusters in the final tally.

Silhouette Index generally gives better results for more clusters, which can also be attributed to the larger occurrence of singletons, clusters that wrongly give optimal results for SI.

Table 3. High energy physics collaboration network clusters and their quality indexes

Algorithm	# Clusters	SI	Mod	Cov	Perf	Cond
MCL	1002	-0.17 ± 0.037	0.35	0.52	0.99	0.51 ± 0.016
MCL	1742	-0.17 ± 0.028	0.33	0.42	0.99	0.62 ± 0.009
MCL	2650	0.005 ± 0.019	0.22	0.27	0.99	0.73 ± 0.005
B. k-means	1005	-0.54 ± 0.012	0.33	0.41	0.99	0.61 ± 0.007
B. k-means	1744	-0.30 ± 0.004	0.30	0.37	0.99	0.61 ± 0.006
B. k-means	2652	-0.14 ± 0.016	0.25	0.31	0.99	0.68 ± 0.003
Spectral	1005	-0.16 ± 0.037	0.34	0.44	0.99	0.53 ± 0.015
Spectral	1710	-0.04 ± 0.025	0.29	0.35	0.99	0.64 ± 0.009
Spectral	2525	0.019 ± 0.019	0.25	0.29	0.99	0.71 ± 0.006
Norm. Cut	1005	-0.59 ± 0.025	0.26	0.33	0.99	0.64 ± 0.02
Norm. Cut	1744	-0.37 ± 0.01	0.18	0.21	0.99	0.70 ± 0.01
Norm. Cut	2652	-0.25 ± 0.014	0.18	0.23	0.99	0.76 ± 0.015

Table 4. High energy physics citation network clusters and their quality indexes

Algorithm	# Clusters	SI	Mod	Cov	Perf	Cond
MCL	814	-0.07 ± 0.037	0.41	0.43	0.98	0.58 ± 0.015
MCL	3898	-0.039 ± 0.017	0.26	0.26	0.99	0.81 ± 0.003
MCL	12911	0.41 ± 0.005	0.12	0.12	0.99	0.93 ± 0.0006
B. k-means	814	-0.71 ± 0.014	0.25	0.25	0.99	0.71 ± 0.005
B. k-means	3898	-0.64 ± 0.008	0.14	0.14	0.99	0.80 ± 0.004
B. k-means	12911	-0.077 ± 0.01	0.06	0.056	0.99	0.90 ± 0.0008
Spectral	812	-0.236 ± 0.04	0.34	0.35	0.99	0.59 ± 0.014
Spectral	3490	0.043 ± 0.016	0.20	0.21	0.99	0.81 ± 0.003
Norm. Cut	814	-0.74 ± 0.006	0.25	0.25	0.99	0.65 ± 0.003
Norm. Cut	3898	-0.70 ± 0.005	0.10	0.10	0.99	0.82 ± 0.002
Norm. Cut	12845	-0.004 ± 0.006	0.06	0.06	0.99	0.92 ± 0.0006

For performance, as we already expected from the observations we did on the formula itself, the sheer size of the networks we worked with here eclipsed any kind of meaningful results we could gather from the clusterings themselves. The results here serve as a confirmation that the expected behavior really happens on real networks.

Another important point raised by our experiments is that networks of different origins might have clusters with very different characteristics. Clusters obtained from technological networks (in our case, the Gnutella snapshots) got markedly poor quality metric results, especially when compared to the results from social networks (all the other networks used). It could be argued those technological networks in particular might not have clusters, but we know that there should be community-like structures in a Gnutella network: a superpeer and its neighboring peers form a fairly cohesive subset, even though it is a sparse one.

It seems that the network structure in this case, with its non clique-like communities, affects very negatively the ability of both clustering algorithms and

Table 5. Gnutella peers network (08/04/2002) clusters and their quality indexes

Algorithm	# Clusters	SI	Mod	Cov	Perf	Cond
MCL	2189	-0.81 ± 0.039	0.0004	0.001	0.99	0.99 ± 0.0
MCL	4724	-0.037 ± 0.015	0.0003	0.0007	0.99	0.99 ± 0.0
MCL	6089	0.10 ± 0.011	0.00003	0.0003	0.99	1.00 ± 0.0
B. k-means	2189	-0.88 ± 0.0001	0.0004	0.001	0.99	0.99 ± 0.00034
B. k-means	4724	-0.52 ± 0.02	0.00007	0.0004	0.99	0.99 ± 0.0
B. k-means	6089	-0.18 ± 0.01	-0.00006	0.0002	0.99	1.00 ± 0.0
Spectral	2158	-0.90 ± 0.0006	0.0004	0.001	0.99	0.99 ± 0.0
Spectral	4079	-0.94 ± 0.0005	0.0001	0.0005	0.99	0.99 ± 0.0
Spectral	6089	-0.30 ± 0.02	-0.00007	0.0002	0.99	1.00 ± 0.0
Norm. Cut	2189	-0.90 ± 0.002	0.0003	0.001	0.99	0.99 ± 0.0
Norm. Cut	4616	-0.2 ± 0.012	0.00025	0.0006	0.99	0.99 ± 0.0
Norm. Cut	5690	0.1 ± 0.012	0.0002	0.0005	0.99	0.99 ± 0.0

quality metrics to identify said clusters. This observation that different kinds of cluster structures exist and that the usual clustering methods wouldn't work with them was already discussed by Nepusz [12]. In that case, she defended that, in a bipartite graph, each one of the sides of the bipartition should be considered as a cluster. Kumar et al. [9] also cite the existence of this kind of cluster structure, pointing out that there are many on-line communities that behave as bipartite subgraphs and giving the websites of cellphone carriers as an example: they represent the same category of service, but will not have direct links to each other.

It is interesting to notice that even the most simple instances of a bipartite graph would score poorly on the quality metrics studied in this paper, as their internal density is nonexistent and all their edges connect to other clusters. For example, consider a small, 10 vertex bipartite graph with two 5 vertex partitions connected by 11 edges. This simple case would give scores such as -0.29 for silhouette index, -5 for modularity, 0 for coverage, 0.31 for performance and 2 for conductance, results that are indeed very poor.

6 Conclusion

In this paper we presented a study of some of the most popular quality metrics for graph clustering, namely, the Silhouette Index, Modularity, Coverage, Performance and Conductance. To evaluate those metrics, we compared their results for clusters generated by four different clustering algorithms: Markovian, Bisecting K-means, Spectral and Normalized Cut. We used seven different real datasets in our experiments, with two of them having an already known optimal clustering based on the semantics of the relationships between the elements represented by their graphs.

Based on our experiments, we could identify some interesting behaviors for those cluster quality assessing metrics. For example, Modularity, Conductance and Coverage have a bias toward giving better results for smaller numbers of clusters, while the other studied metrics have a completely opposite bias. This indicates that all those metrics do not share a common view of what a true clustering should look like.

Our results suggest that there is no such a thing as a "best" quality metric for graph clustering. Even more, the currently used quality metrics have strong biases that do not always point in the direction of what is assumed to be a well-formed cluster. Also, those biases can get even more pronounced in large graphs, which are the ones that depend on those metrics the most, as they are the hardest to manually evaluate any results.

Another point observed was that the structure of clusters can be different for graphs with different origins. In our case, we saw clear differences in the results of technological and social networks. Current clustering and evaluation techniques seem to be inadequate to tackle those different kinds of complex networks.

As future work, we intend to study how particular aspects of a graph topology can affect the structure of a cluster, so that we can evaluate clusters with different characteristics, and not only the clique-like ones. We will also consider adding other dimensions to the graph, such as weights and labels.

Acknowledgments. This research was partially funded by FAPEMIG, CNPq, CAPES, Finep and the Brazilian National Institute for Science and Technology of the Web — InWeb (MCT/CNPq 573871/2008-6). MJZ was supported in part by NSF grant EMT-0829835, and NIH grant 1R01EB0080161-01A1.

References

1. Brandes, U., Gaertler, M., Wagner, D.: Engineering graph clustering: Models and experimental evaluation. J. Exp. Algorithmics 12, 1–26 (2008)
2. Danon, L., Díaz-Guilera, A., Duch, J., Arenas, A.: Comparing community structure identification. Journal of Statistical Mechanics: Theory and Experiment 2005(09), P09008 (2005)
3. Dhillon, I.S., Guan, Y., Kulis, B.: Weighted graph cuts without eigenvectors a multilevel approach. IEEE Trans. Pattern Anal. Mach. Intell. 29(11), 1944–1957 (2007)
4. Van Dongen, S.: Graph clustering via a discrete uncoupling process. SIAM Journal on Matrix Analysis and Applications 30(1), 121–141 (2008)
5. Girvan, M., 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)
6. Good, B.H., de Montjoye, Y.A., Clauset, A.: Performance of modularity maximization in practical contexts. Physical Review E 81(4), 046106+ (2010)
7. Gustafson, M., Lombardi, A.: Comparison and validation of community structures in complex networks. Physica A: Statistical Mechanics and its Application 367, 559–576 (2006)
8. Kannan, R., Vempala, S., Vetta, A.: On clusterings: Good, bad and spectral. J. ACM 51(3), 497–515 (2004)
9. Kumar, R., Raghavan, P., Rajagopalan, S., Tomkins, A.: Trawling the web for emerging cyber-communities. Comput. Netw. 31, 1481–1493 (1999)

10. Leskovec, J., Lang, K.J., Dasgupta, A., Mahoney, M.W.: Statistical properties of community structure in large social and information networks. In: WWW 2008: Proceeding of the 17th International Conference on World Wide Web, pp. 695–704. ACM, New York (2008)
11. Leskovec, J., Lang, K.J., Mahoney, M.: Empirical comparison of algorithms for network community detection. In: Proceedings of the 19th International Conference on World Wide Web, WWW 2010, pp. 631–640. ACM, New York (2010)
12. Nepusz, T., Bazso, F.: Likelihood-based clustering of directed graphs, pp. 189–194 (March 2007)
13. Nepusz, T., Sasidharan, R., Paccanaro, A.: Scps: a fast implementation of a spectral method for detecting protein families on a genome-wide scale. BMC Bioinformatics 11(1), 120 (2010)
14. Newman, M.E., Girvan, M.: Finding and evaluating community structure in networks. Physical Review E 69(2) (February 2004)
15. Newman, M.E.J.: Mixing patterns in networks. Phys. Rev. E 67(2), 26126 (2003)
16. Newman, M.E.J., Girvan, M.: Mixing Patterns and Community Structure in Networks. In: Pastor-Satorras, R., Rubi, M., Diaz-Guilera, A. (eds.) Statistical Mechanics of Complex Networks. Lecture Notes in Physics, vol. 625, pp. 66–87. Springer, Berlin (2003)
17. Schaeffer, S.E.: Graph clustering. Computer Science Review 1(1), 27–64 (2007)
18. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 22, 888–905 (2000)
19. Steinbach, M., Karypis, G., Kumar, V.: A comparison of document clustering techniques. In: Grobelnik, M., Mladenic, D., Milic-Frayling, N. (eds.) KDD-2000 Workshop on Text Mining, Boston, MA, August 20, pp. 109–111 (2000)
20. Tan, P.-N., Kumar, V., Srivastava, J.: Selecting the right interestingness measure for association patterns. In: KDD 2002: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 32–41. ACM, New York (2002)
21. Tan, P.-N., Steinbach, M., Kumar, V.: Introduction to Data Mining. Addison-Wesley Longman Publishing Co., Inc., Boston (2005)
22. van Dongen, S.M.: Graph Clustering by Flow Simulation. PhD thesis, University of Utrecht, The Netherlands (2000)
23. Zachary, W.W.: An information flow model for conflict and fission in small groups. Journal of Anthropological Research 33, 452–473 (1977)
24. Zhou, Y., Cheng, H., Yu, J.X.: Graph clustering based on structural/attribute similarities. Proc. VLDB Endow. 2(1), 718–729 (2009)