Vladimir Coxall
CSE 8803

Evaluating a Randomized Greedy Agglomerative Community Detection Method by Surprise

Real world interactions between entities can often give rise to a connected graph structure. For example, users of a social network can be represented as a graph of vertices with edges representing connections or friendships between them. Such graphs are not only limited to social network data. Road networks, power transmission infrastructure and even scientific literature citations can be represented as graphs. A current problem is how to partition similar groups of users in such a graph. This can prove useful to understand how they form and change over time, in the case of populations to market goods and services to them, or in a defense context, to model their behavior and assess their threat risk. Since these graphs can span millions or even billions of vertices with complex edge structures, an accurate and fast method for finding community structure is necessary. This paper explores such a method using a new basis for measuring the quality of the partition.

Graphs representing social network data often exhibit what is known as a power law distribution: A large number of vertices have low degree and a small number of vertices have very high degree. In the context of the universe of Facebook users, this can be thought of as a few number of Facebook users having a large number of friends while the majority of users will have comparatively fewer. Regardless of the context of the graph, an interesting question arises: “How can one partition the graph so that similar users are together in communities?”. In being able to group and track communities of users, it is possible to understand their properties, how they form and evolve over times. While there are many approaches to solving this problem, the vast majority of community detection methods take ideas proposed by Newman in his paper on community detection [1]. A community in a natural group is densely connected among members within the community relative to those who are nonmembers. Then, a community detection scheme should seek to partition the graph G, such that the density of observed edges within each community is as close to the density of edges expected in clusters of a random graph with similar degree distribution. A metric known as modularity (denoted by Q) was defined as follows:

$$Q = \sum_i \left(e_{ii} - a_i^2\right)$$

$$e_{ij} = \frac{\sum_{v_x \in C_i} \sum_{v_y \in C_j} m_{xy}}{\sum_{v_x \in V} \sum_{v_y \in V} m_{xy}}$$

$$a_i = \sum_j e_{ij}$$

where m_{xy}= 1 if an edge exists between vertex x and vertex y; Modularity as a score measures how accurate a partition of G is to its ideal.

There are various ways of manipulating the graph which hope to find an optimal partition that maximizes modularity. One such group of methods, spectral methods, considers the mathematical properties of the graph through its eigenvalues and eigenvectors. Another, divisive methods, take a full graph G and repeatedly search for vertices which by moving them to a separate group will cause the maximum gain in modularity at that step. The method implemented in this paper considers the reverse approach. An agglomerative community detection method takes a graph G, and considers each vertex to be in its own community. Ever pairwise group of communities is considered for merging. The pair which will give the largest increase in the modularity score are merged and the process repeated until all communities have been merged into a single community. Given a large scale graph with possibly billions of edges, as social networks tend to be, it is infeasible to consider all possible pairwise combinations and their change in modularity given current computing and time limitations.

To overcome this problem, a number of variants have been proposed. The Walktrap algorithm by Pons [2] considers a Markov chain of communities where one takes a random walk among communities probabilistically finding pairs of communities to merge. Another proposed by Ovelgonne et. al [3] (and used as the basis for this work) proposed a randomized greedy approach. In their work, they note that modularity can be viewed as a local property for community detection because only those communities which are directly connected will cause a net increase in the Q score. Thus one does not need to Q score all other communities $\{C_1,...,C_n\}$ in G for a proposed merger with a community C_j but only a subset (those which are connected). Further they have found that in many cases, different cluster candidates may cause the same increase in Q when merged. From this, they argue that it is sufficient to consider a small subset of clusters at each merge step. Varying the size of the subset, Ovelgonne et al. found that beyond considering 2 neighboring communities of a random community C_j there is no appreciable difference in the final quality of partitioning.

Although modularity is widely used as the measure for optimization in community detection schemes, there has been work by Fortunato et al. highlighting problems of using modularity as a metric [4]. Perhaps the most important is a tendency for modularity based schemes to repeatedly form a large cluster and increase the Q score by merging other clusters into the larger cluster even when it is not appropriate to do so. While the Q score may increase, the new partition reduces the actual quality of the community structure. Another measure of the partition quality, “surprise” has been proposed by Aldecoa and Marin in the computational biology domain [5]. Surprise (denoted by S), scores partitions as those which best minimize the cumulative hypergeometric distribution:

$$\sum_{j=p}^{\mathrm{Min}\ (M,n)} \frac{\binom{M}{j}\binom{F-M}{n-j}}{\binom{F}{n}}.$$

where F is the maximum number of edges among all vertices, M is the maximum number of intracluster edges among all clusters, p is the number of actual intracluster edges among clusters, and n is the number of observed edges in the graph. The GNU Scientific

Library (GSL) is used to calcuate the hypergeometric functions.

$$F = \frac{|V| * (|V| - 1)}{2}$$

$$M = \sum_{j=1}^{|C|} |Vj| * (|Vj| - 1) / 2$$

Results

Using Ovelgonne et al's Randomized Greedy Agglomerative Clustering algorithm on the Zachary Karate graph, and scoring using each partition using surprise, it was found that the best partition was after 24 merges. This resulted in five distinct community structures with the remaining elements being singletons. On this network it took 0.002507 seconds to perform the entire community detection scheme. A disadvantage of this process it that pairwise merges are made based on modularity and then scored. A better approach is one which in a similar fashion to modularity, iteratively picks clusters for merging based on the highest increase in the surprise metric. There are three ways of doing this:

1. Consider all possible merges and calculate their surprise score. This will be quadratic in the number of clusters and computationally infeasible for extremely large graphs
2. Using Ovelgonne's argument for cluster locality, find a lower bound on the size of the random subset of clusters to sample, such that the probability of increasing surprise significantly is above a threshold.
3. Treat the partitioning at each step as a combinatorial optimization problem. For a graph G, F and n will be fixed but a process which finds optimal p and M values would provide parameters of how to construct clusters for a local optimum solution that maximizes the S score.

Vladimir Coxall
CSE 8803

References

[1] Newman, Mark EJ. "Modularity and community structure in networks." *Proceedings of the National Academy of Sciences* 103.23 (2006): 8577-8582.

[2] Pons, Pascal, and Matthieu Latapy. "Computing communities in large networks using random walks." *Computer and Information Sciences-ISCIS 2005* (2005): 284-293.

[3] Ovelgönne, Michael, Andreas Geyer-Schulz, and Martin Stein. "Randomized greedy modularity optimization for group detection in huge social networks." *SNA-KDD'10: Proceedings of the 4th Workshop on Social Network Mining and Analysis*. 2010.

[4] Fortunato, Santo, and Marc Barthelemy. "Resolution limit in community detection." *Proceedings of the National Academy of Sciences* 104.1 (2007): 36-41.

[5] Aldecoa, Rodrigo, and Ignacio Marín. "Deciphering network community structure by surprise." *PloS one* 6.9 (2011): e24195.