Approximating the Community Structure of the Long Tail

Akshay Java, Anupam Joshi, Tim Finin
University of Maryland, Baltimore County, MD 21250
{aks1, joshi, finin}@cs.umbc.edu

Introduction

Communities are central to online social media systems and detecting their structure and membership is critical for many applications. The large size of the underlying graphs makes community detection algorithms very expensive. We describe an approach to reducing the cost by estimating the community structure from only a small fraction of the graph. Our approach is based on an important assumption that large, scale-free networks are often very sparse. Such networks consist of a small, but high degree set of core nodes and a very large number of sparsely connected peripheral nodes (Borgatti & Everett 2000). The insight behind our technique is that the community structure of the overall graph is very well represented in the core. The community membership of the long tail can be approximated by first using the subgraph of the small core region and then analyzing the connections from the long tail to the core.

A set of vertices can constitute a community if they are more closely related to one another than the rest of the network. Such vertices connect with a higher density within the group and are very sparsely connected to the rest of the network. An intuitive measure for the quality of any clustering or community detection algorithm is the modularity function (Newman & Girvan 2003). The modularity function, Q, measures the fraction of all the edges, e_{ii}, that connect within the community to the fraction of edges, a_i that are across communities. This measure is defined as $Q = \sum_i (e_{ii} - a_i^2)$. Determining the "best" community structure by finding the optimal modularity value has been shown to be NP-Hard (Duch & Arenas 2005).

Recently, spectral methods have been applied to community detection and shown to have a relation to optimizing the modularity score (Newman 2006). Spectral clustering is based on the analysis of eigenvectors of a graph or more generally, any similarity matrix. Most spectral clustering techniques use the graph Laplacian which is a representation of the similarity matrix that has a number of important properties (Chung 1997). The normalized version for the graph Laplacian is given by: $L = D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$ where $W \in \Re^{n*n}$ is the adjacency matrix and D is a diagonal matrix representing the degrees of nodes in the graph. An important property of the graph Laplacian is that the vector corresponding to the second smallest eigenvalue, the *Fiedler vector*, can be easily used to partition the graph (Chung 1997). It has been used to efficiently cluster data and partition graphs into communities. Shi and Malik (Shi & Malik 2000) developed a normalized cut (Ncut) criteria by recursively partitioning the graph using the eigenvectors of the graph Laplacian.

Sampling Based Low Rank Approximations

Our approach is inspired by the idea of using sampling for clustering as described by Drineas *et al.* (Drineas *et al.* 1999) and Keuchel and Schnorr (J. Keuchel 2003). We first permute the adjacency matrix W, based on the degree. Next, we compute the normalized Laplacian matrix associated with a given graph. Since the original adjacency matrix, W, is sparse, L is also sparse. Also, it has the same structure as W. Thus, L can be partitioned into four submatrices as follows: [A B; A B^T] such that $A \in \Re^{r*r}$, $B \in \Re^{k*(n-r)}$ and $C \in \Re^{(n-r)*(n-r)}$. Here, A represents the connectivity between nodes in the core, and B the connectivity of the nodes outside the core to those in the core. Now using Singular Valued Decomposition (SVD) L can be factorized into it's singular values and corresponding basis vectors: $L = D^{-\frac{1}{2}} W D^{-\frac{1}{2}} = \sum_{i=1}^{n} \rho_i q_i p_i^T$, where ρ_i are the singular values, q_i and p_i are the left and right singular vector correspondingly. If Q_k is a matrix of left orthonormal singular vectors then the best *k* approximation of L is given by: $L = Q_k * Q_k^T * L$.

The approximate value for Q_k (left largest singular vectors) can be obtained by using the eigenvectors corresponding to the k largest eigenvalues of $S^T * S$ where the submatrix $S \in \Re^{nxs}$ is given by [A; B^T]. Let w_i be the eigenvectors corresponding the *k* largest eigenvalues of matrix $S^T * S$. Then the approximated Q_k of the L can be found by

$$qi = S * \frac{w_i}{||Swi||} = S * \frac{w_i}{\sqrt{\lambda i}} \; for \; i = 1......k$$

where λ_i denotes the eigenvalues of the matrix $S^T S$.

Since L is a positive semi-definite matrix, the singular values and singular vectors of L coincide with its eigenvalues and eigenvectors. This leads to Q_k approximating the k eigenvectors needed for community detection. A different interpretation of this approach is in terms of the Nyström method.

Heuristic Based Approximation Method

We propose another approach that uses the structure of the blogs graph directly. We use the head of the distribution to first find the communities in the graph. The intuition is that communities might form around popular nodes. So we can use Ncut (or any of the community detection algorithms) to find the initial communities in a graph that is much smaller than the original one. This leaves the problem of finding the community of the blogs that are not a part of the head. One simple, yet effective heuristic is to look at the number of links from a blog to each community as identified from the clustering of the nodes in the head, and declare it to be a member of the community that it most associates with by this measure. This heuristic can significantly reduce the computation time, while providing a reasonable approximation to the community structure that would be found by running the same Ncut algorithm over the entire graph.

Evaluation

In order to evaluate the quality of approximation we use a blog graph consisting of six thousand nodes. Figure 1 shows the original sparse matrix permuted using the degree of the node to reveal the core-periphery structure of the graph also shows the communities detected using the heuristic method. Since there is no ground truth available, we use the modularity score, Q, as a measure of quality for the resulting communities found by each of the methods. We also compare the approximate methods with Ncut algorithm. Another possible way to approximately calculate communities would be to cluster the singular vectors U obtained using the low-rank approximations of the original large, sparse matrix W.

Figure 2 shows the performance of Ncut, low-rank SVD, approximation method and heuristic method for computing the communities. The results indicate that both the approximation and heuristic method provide high modularity scores even at low sampling rates (10%-50%). Also, the time required to compute the communities is comparable or at times less than that of using Ncut. In addition the memory requirements are much less since only a small fraction of the entire graph is sampled.

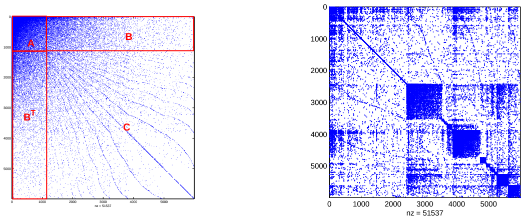

Figure 1: A Webgraph consisting of 6000 blogs (left) is sampled at 30% using the heuristic method. The resulting communities identified are shown on the right. The modularity score was found to be about 0.5 using 20 communities.

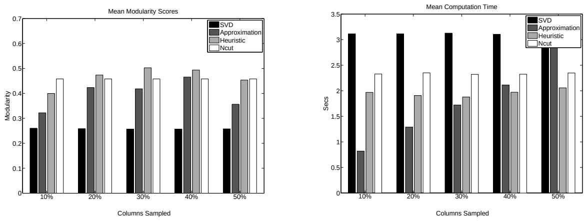

Figure 2: Modularity scores (left) and computation times (right) for different sampling rates (10% to 50%) over 100 runs. Bars are in the following order: SVD, Sampling Based Approximation, Heuristic, Ncut.

In terms of running time the complexity of Ncut is $\mathcal{O}(nk)$ where n is the number of nodes in the graph and k is the number of communities. Thus the heuristic is $\mathcal{O}(rk)$ where r is the number nodes in the head. On the other hand, the complexity of low rank SVD is $\mathcal{O}(nk^2)$ where the original graph is approximated by its rank k basis vectors. Finally the sub-sampling based approximation can be efficiently implemented in $\mathcal{O}(r^3)$ using the Nyström Method (Fowlkes *et al.* 2004).

Conclusions

In this work, we present approximate methods for community detection in large graphs. It has the advantage of quickly and efficiently finding a reasonable approximation to the community structure of the overall network. We also present an intuitive heuristic and show that it results in good performance at a much lower costs.

References

Borgatti, S. P., and Everett, M. G. 2000. Models of core/periphery structures. *Social Networks* 21(4):375–395.

Chung, F. R. K. 1997. *Spectral Graph Theory (CBMS Regional Conference Series in Mathematics, No. 92) (Cbms Regional Conference Series in Mathematics)*. American Mathematical Society.

Drineas; Frieze; Kannan; Vempala; and Vinay. 1999. Clustering in large graphs and matrices. In *SODA: ACM-SIAM Symposium on Discrete Algorithms (A Conference on Theoretical and Experimental Analysis of Discrete Algorithms)*.

Duch, J., and Arenas, A. 2005. Community detection in complex networks using extremal optimization. *Physical Review E* 72:027104.

Fowlkes, C.; Belongie, S.; Chung, F.; and Malik, J. 2004. Spectral grouping using the nyström method. *IEEE Trans. Pattern Anal. Mach. Intell.* 26(2):214–225.

J. Keuchel, C. S. 2003. Efficient graph cuts for unsupervised image segmentation. using probabilistic sampling and svd-based approximation.

Newman, M. E. J., and Girvan, M. 2003. Finding and evaluating community structure in networks.

Newman, M. E. J. 2006. Finding community structure in networks using the eigenvectors of matrices. *Physical Review E* 74:036104.

Shi, J., and Malik, J. 2000. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22(8):888–905.