On the structure of communities in networks

Bruno Abrahao Sucheta Soundarajan John Hopcroft Robert Kleinberg

Department of Computer Science
Cornell University
{abrahao, sucheta, jeh, rdk}@cs.cornell.edu

1 Introduction

Community structure captures the tendency of entities in a network to group together in meaningful subsets whose members have a distinctive relationship to one another. Despite playing a fundamental role in the structure and function of networks, community structure has proved to be frustratingly difficult to define, quantify, and extract. In addition to challenges related to computational tractability, several major factors account for the intricacies of community extraction.

First, the application domain includes a wide variety of networks of fundamentally different natures. Second, the literature offers a multitude of disparate community detection algorithms. Due to differences in concept and design, the output of these procedures exhibits high structural variability across the collection. Next, there is no established consensus on the question of what properties distinguish subgraphs that are communities from those that are not communities. Additionally, it is difficult to obtain negative examples of communities; in theory, we can obtain examples of community structure, e.g., by asking experts to identify communities in a given domain, and then declare that every other subset of nodes in the network is a negative example. However, enumerating all forms of negative examples is obviously intractable, and even if we could enumerate all possible negative examples, we are still faced with the problem that these sets might also be examples of valid communities that the expert failed to identify.

In this work, we address these issues by presenting a framework that enables researchers to assess the structural dissimilarity among the output of multiple community detection algorithms and between the output of algorithms and communities that arise in practice. Our approach analyzes communities by taking account of a broad spectrum of structural properties, and reveals nuances of the structure of communities.

2 Overview

We frame this as a class separability problem, which simultaneously handles many classes of communities and types of structural properties. To this end, we specify a learning problem in which we map the distinct communities into a feature space, where the dimensions represent measures that characterize a community's link structure. The separability of classes provides information on the extent to which different communities come from the same (or different) distributions of feature values. We extract different classes of communities that can be labeled as either intrinsically-defined or extrinsically-defined communities.

We define the first set of communities by properties intrinsic to their link structure. For our purposes, these are the sets that community detection algorithms output. Each class of intrinsically defined communities comprises a set of examples that a specific algorithm extracts. We also define communities by meaningful annotations provided with the datasets, such as explicit declaration of community membership, product categories, grouping by protein function, and so on. In this fashion, for each network, we form a class of extrinsically-defined communities.

To demonstrate our approach, we furnish our framework with a large set of structural properties and consider ten different community detection procedures, representative of various categories of popular algorithms, to produce examples of different structural classes. We consider a diverse collection of large scale real networks whose domains span biology, on-line shopping, and social systems. We then assess separability using super-

vised classifiers both parametric, namely Support Vector Machines [10], and nonparametric, namely k-Nearest Neighbors [1], together with a feature selection analysis using correlation-based methods [5].

3 Methodology

We analyze nine large datasets, namely DBLP, two portions of the LiveJournal social network (denoted as LJ1 and LJ2), two portions of the Facebook network (denoted as Grad and Ugrad), Amazon, and three genetic networks denoted by HS, SC, and Fly. These datasets range in size from 503 to 500,000 nodes.

The networks we analyze contain annotations, which we use to identify extrinsically defined (or annotated) communities. Some of these sets are user-defined, i.e., users explicitly declare their participation in the community, while others reflect contextual information of the underlying process or organization, e.g., university department, protein function, product category, etc.

To study classes of intrinsically defined communities, we selected a collection of 10 community detection procedures, which are representative of strategies employed by a broad range of algorithms in the literature. We applied these procedures to each of the nine network datasets, and labeled the resulting sets with the identity of the community detection procedure that produced them. In total, for each network, we created 11 structural classes of communities: one class of extrinsically-defined communities, which comprises examples of annotated communities, and each of the other 10 classes corresponding to intrinsically-defined communities, which comprise examples extracted by each of the 10 community detection algorithms respectively.

The algorithms that we consider are breadth-first search (BFS), random walk with and without restart (RW15 and RW0), an algorithm to identify $\alpha-\beta$ communities [7] (AB), InfoMap [9], Link Communities (LC) [2], Louvain Modularity [3], Newman-Clauset-Moore Modularity [8], Markov Clustering Algorithm (MCL) [4], and Metis [6].

In the next phase, we measure the subgraphs induced by the communities produced in the previous step and those induced by annotated communities. We use a large spectrum of measurements that cover many properties of both the internal link structure and the external interaction of the community with the rest of the network. These measurements include features such as size, conductance, edge density, and distributions of various centrality measures (such as node betweenness, edge betweenness and information centrality).

By measuring these structural properties for each example of a community derived in the previous phase, we obtain 11 classes of labeled examples in feature space, which constitute the input in our framework.

In this work, we treat the research question of discriminating the structure of different communities as a class separability problem. This analysis is informative of the extent to which different algorithms produce structural differences and the extent to which community detection algorithms succeed in producing sets that resemble annotated communities. More specifically, we employ the Support Vector Machine (SVM) and k-Nearest Neighbor methods to confirm each other's outcomes while ruling out variability due to the specifics of each algorithm.

We perform two experiments. In the first, we are interested in analyzing structural consistency within the 11 classes of communities. For example, do the communities generated by BFS tend to resemble one another, or is there confusion between different classes? For each network, we use cross-validation to train a multi-class classifier on elements from each of the 11 classes of communities from that network. We then evaluate the classifier on the remaining elements from each of the 11 classes. Table 1 contains the results of this experiment. We see that for most networks and classes, a plurality of the probability mass from that class was properly classified. This experiment shows that the classes of communities tend to be internally consistent, demonstrating that different community detection methods produce results that are fundamentally different from each other, and suggesting that for a practitioner who wishes to find a specific type of community, the choice of algorithm is crucial.

While the first experiment showed that the various classes, including the class of annotated communities, tend to be internally consistent, we are also interested in identifying which of the 10 intrinsically defined classes the set of annotated communities most resembles. That is, although no algorithm fully captures the structure of annotated communities, which comes the closest? We train a classifier on elements from the 10 classes of communities identified through algorithms, and then apply this classifier to the set of annotated communities. We see that for nearly every network, the annotated communities are most similar to the two classes of random walk communities.

Finally, we apply the Correlation-based Feature Selection method [5] to each of the 11 classes of communities in order to determine which features are most valuable in discriminating between classes. We see that for most

	Grad	Ugrad	HS	SC	Fly	DBLP	Amaz	LJ1	LJ2
BFS	60%	88%	73%	70%	(40%)	63%	55%	86%	81%
RW0	44%	55%	43%	(39%)	(27%)	52%	43%	61%	63%
RW15	40%	(29%)	44%	42%	34%	46%	39%	57%	57%
AB	83%	91%	90%	71%	60%	70%	74%	90%	89%
IM	27%	(23%)	72%	73%	(2%)	62%	51%	82%	66%
LC	68%	96%	83%	85%	83%	67%	56%	90%	89%
Louv.	24%	(3%)	49%	(1%)	(0%)	45%	58%	38%	49%
Newm.	(14%)	(25%)	(15%)	(0%)	90%	26%	39%	45%	56%
MCL	19%	(22%)	57%	28%	(34%)	59%	46%	80%	74%
Metis	61%	73%	81%	90%	(42%)	88%	66%	92%	86%
Annot.	37%	33%	50%	46%	(8%)	47%	40%	72%	71%

Table 1: Percentage of the probability mass of classification of elements in the test set into the correct class, using SVM, for all networks. Values in parentheses indicate that a plurality of the probability mass from that was classified as some other class.

networks, conductance and diameter are especially valuable.

As illustrated by our experiments, by producing artificial or real examples of communities that possess the structure we wish to find, we can use our framework to enable an informed choice of the most suitable community detection method for a given network. In addition, it allows for a comparison of existing community detection algorithms and may guide the design of new ones.

References

[1] D. W. Aha, D. Kibler, and M. K. Albert. Instance-based learning algorithms. *Machine Learning*, 6:37–66, January 1991.

[2] Y. Ahn, J. P. Bagrow, and S. Lehmann. Link communities reveal multiscale complexity in networks. *Nature*, 466(7307):761–764, 2010.

[3] V. D. Blondel, J. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. Mar. 2008. Journal of Statistical Mechanics.

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

[5] M. A. Hall. *Correlation-based Feature Subset Selection for Machine Learning*. PhD thesis, Department of Computer Science, University of Waikato, 1999.

[6] G. Karypis and V. Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM J. Sci. Comput.*, 20:359–392, December 1998.

[7] N. Mishra, R. Schreiber, I. Stanton, and R. Tarjan. Finding strongly knit clusters in social networks. *Internet Mathematics*, 5(1):155–174, Jan. 2008.

[8] M. Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577 –8582, June 2006.

[9] M. Rosvall and C. Bergstrom. Multilevel compression of random walks on networks reveals hierarchical organization in large integrated systems. *PLoS ONE*, 6(4):e18209, 04 2011.

[10] V. N. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, 1st edition, Sept. 1998.

[11] E. Weinan, T. Li, and E. Vanden-Eijnden. Optimal partition and effective dynamics of complex networks. *Proceedings of the National Academy of Sciences*, 105(23):7907–7912, June 2008.