DETECTING ACTIVITY-BASED COMMUNITIES USING DYNAMIC MEMBERSHIP PROPAGATION

Scott Philips, Michael Yee, Edward Kao and Christian Anderson

MIT Lincoln Laboratory
Lexington, Massachusetts 02420 USA
{scott.philips,myee,edward.kao,christian.anderson}@ll.mit.edu

ABSTRACT

Existing literature on network community detection typically exploits the structure of static associations between entities. However, real world network data often consists of observations of coordinated interactions between members who belong to multiple communities. This paper presents a novel perspective and approach for activity-based community detection, where a community is defined as a group of actors engaged in correlated activities over time. Detection is performed by propagating membership iteratively to neighboring nodes through edges that represent interactions. We compare the proposed approach to two state-of-the-art methods based on modularity, and demonstrate its effectiveness on a simulated vehicle movement dataset and the Enron email corpus.

Index Terms— Graph Theory, Community Detection

1. INTRODUCTION

Community detection remains one of the most prominent applications in the field of network inference. It finds the underlying community memberships of network entities, given the relationships and interactions between them. There are a wide variety of such data on communication, social, and biological networks; for example, email traffic between employees of a company [1], vehicle traffic between physical locations [2], collaborations between scientists [3], protein-protein interactions [4], etc. Discovering community membership is of practical value because it reveals group identities not readily observable and provides insight on the behaviors within and between groups.

Typically, community detection is performed on a static graph by identifying nodes that are more tightly connected to each other than to the rest of the graph (*i.e.* homophily). However, real world network data often challenges the assumption of homophily, as entities (*i.e.* nodes) interact (*i.e.* edges) with members of different communities over time. Under this circumstance, community detection based on static association suffers in performance. A new perspective of community detection based on coordinated activities is needed. This approach should exploit interactions that correlate temporally at the group level; for example, a task is carried out through multiple meetings where actors gather at and depart from the same location at roughly the same time. It also should exploit interactions that correlate temporally at the individual level; for example, a person typically responds to an incoming email within a short period of time, noted by Perry, et al. as the *"reciprocation effect"* [5]. This paper demonstrates a method to detect activity-based communities by exploiting the dynamics of the observed interactions.

The remainder of this paper is organized as follows. *Section 2* describes the background literature that is closely related to the membership propagation method. *Section 3* introduces the static membership propagation method and *Section 4* extends this method with the use of a temporal interaction kernel. Finally, *Section 5* demonstrates effectiveness of the proposed approach on simulated and real-world data sets and *Section 6* concludes with summary and future work.

2. BACKGROUND

There has been a large body of work dedicated to community detection on network data. Fortunado gives a comprehensive survey in [6]. The majority of the work focuses on global partitioning of a static graph. Some of these approaches have been applied locally, given a tip (*i.e.* starting node) into the community. Little work has been done to utilize the dynamics of correlated interactions.

The most well-known approach is probably *"modularity maximization"* proposed by Newman [7]. Modularity quantifies the degree to which within-community edges exceed the expected number of edges of a random graph. Maximizing this quantity provides an intuitive graph partition where members of a community are more tightly connected to each other than to outside communities (*i.e.* homophily). Locally at each node, the modularity can be thought of as the fraction of all edges going to members of the same community. Indeed, Clauset proposes an approach to detect local community from a starting node (*i.e.* tip node), by iteratively adding nodes to the local community to maximize local modularity [8]. Community detection based on the modularity can be implemented efficiently using spectral methods [7]. Recent work by Miller et al. performs community detection through *"eigenspace analysis"* on the modularity matrix [9]. *"Local modularity maximization"* and *"eigenspace analysis"* provide a baseline for performance comparison with the method proposed in this paper.

This body of previous work points to an important intuition used to update local membership: the hidden attribute of a given node is updated according to an aggregate statistics from its neighboring nodes. A well-known algorithm that follows this intuition, although not for community detection, is Kleinberg's work on *page ranking* [10], where the importance of a page is updated by the normalized sum of the importance of its neighbors. Kleinberg also shows through spectral decomposition that the iterative update will converge. Our static membership propagation method described in *Section 3* is originally inspired by this approach.

This work is sponsored by the Office of Naval Research and the National Geospatial-Intelligence Agency under Air Force Contract FA8721-05-C-0002. Opinions, interpretations, conclusions and recommendations are those of the author and are not necessarily endorsed by the United States Government.

All of the work discussed so far does not consider dynamics nor exploit the temporal correlation between edges. There has been some recent emerging work in modeling network dynamics in the statistics community. Fu et al. proposes a dynamic model [11] by extending the *mixed-membership stochastic blockmodels* by Airoldi et al. [12]. This model learns the long-term, association-based community memberships and the interactions between communities over time. However, it does not model the dynamics of interactions that varies at a much shorter time scale. Perry et al. proposes a *point process model* to learn the dynamic processes of an email network [5]. Although his work is not on community detection, the results show that attributes describing process dynamics play a large role in data fitting. Ferry proposes a *Bayesian filter* framework to track group memberships and dynamic processes [13]. His work offers valuable theoretical insights but does not yet scale to practical problems.

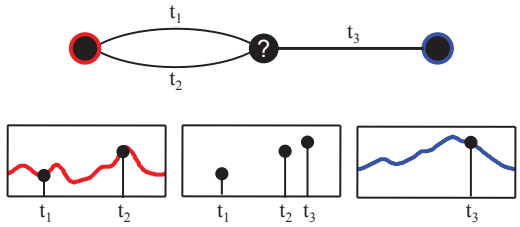

Fig. 1. Notional example of the dynamic community detection problem: given membership probabilities over time on the outside red and blue nodes as well as edge times, estimate the continuous time varying membership probability on the center node.

3. STATIC MEMBERSHIP PROPAGATION

We start with the static local community detection problem, that is, we assume that the presence and weight of each edge in the graph is known and fixed. For example, in the case of an email contact network, once an edge weight between a pair of nodes has been determined by integrating the history of emails between them, the actual timing of the emails is ignored. We detect nodes that belong to the community of interest by propagating community membership from tip nodes (nodes whose community membership is known a priori) to other nodes, along edges which represent interactions.

Inspiration for this approach comes from several sources. In the spread of infectious disease, infectious agents are deposited at sites and transmitted from person to person as infection propagates throughout the graph. In social network analysis, the concept of eigenvector centrality defines the centrality (or importance) of a node as proportional to the sum of the centrality of its neighbors. Google's PageRank algorithm [14] adapts eigenvector centrality to the information retrieval domain and posits that a webpage has high rank if highly ranked pages link to it. Similarly, Kleinberg's HITS algorithm [10] defines hub and authority scores in a mutually recursive way: the hub score of a page is a function of the authority scores of the pages that link to it, and the authority of a page as a function of the hub scores of the pages that link to it. When PageRank and HITS are computed iteratively using power iteration to find the relevant dominant eigenvectors, the quantities of interest can be viewed as propagating along hyperlinks.

We repurpose the family of eigenvector centrality techniques to estimate membership (instead of importance) and generalize in two key ways. First, we incorporate existing knowledge through membership estimates for tip nodes. In this framework, one or more tip nodes can be present, each with a fixed probability of membership between zero and one. Second, we use a non-linear propagation function that can vary by node type. This adds flexibility in modeling the propagation "physics" for a particular domain.

Let G be a graph, N be the set of nodes, and E be the set of edges. Let P_i be the probability that node i is a member of the community of interest. We estimate P_i using estimates from the local neighborhood of i as follows:

$$P_i = \alpha \left(\frac{\lambda}{|N(i)|} \sum_{j \in N(i)} P_j + (1-\lambda) \max_{j \in N(i)} P_j \right), \qquad (1)$$

where $N(i)$ is the set of i's neighbors, $\alpha \in (0,1)$ is a dampening factor, and $\lambda \in [0,1]$ varies the relative contributions of the mean and max terms in the propagation function. Note that α and λ could depend on some attribute of node i depending on the problem domain.

Using a combination of mean and max can be beneficial in situations where a high degree node has a single neighbor with high membership estimate. With only the mean term, the contribution of this neighbor would be washed out. By including the max term, e.g., with $\lambda = 0.5$, the neighbor with the maximum estimate receives a combined weight of $\frac{|N(i)|+1}{2|N(i)|}$ in Equation (1). Sensitivity to degree is thereby reduced as this neighbor gets more than half the weight regardless of node i's degree.

Membership estimates for tips remain fixed throughout the algorithm. Estimates for non-tip nodes are computed iteratively by updating P_i using estimates of neighboring nodes from the previous iteration. We stop when the estimates are mutually consistent according to Equation (1), or in practice when the changes in estimates from one iteration to the next are small. This is essentially a form of fixed point iteration. Similarly to other techniques in the eigenvector centrality family, convergence is guaranteed with $\lambda = 0$. For a general propagation function it has not been shown that the proposed technique has a guarantee of convergence. However, in practice we observed convergence for a wide range of λ over all datasets evaluated.

4. DYNAMIC MEMBERSHIP PROPAGATION

As outlined in the previous section, membership propagation allows the probability of a node's community membership to be estimated locally using only neighboring nodes. In this section we extend this concept to allow a node's membership probability to vary over time. By tracking this membership over time we can determine when a node is acting as a member of a specific community, controlling how and when membership propagates throughout the network.

Consider a node i with M edges connecting to N neighbors. Assume each edge connecting two nodes occur at a specific time and that multiple edges can exist between any two nodes (*i.e.* $M \geq N$). Given dynamic membership probabilities on each neighboring node $P_j(t)$ we wish to estimate the dynamic probability of membership on node i. Figure 1 illustrates this problem. Membership probability on the outer red and blue nodes are to be used to estimate the probability on the center black node. Timing of the edges are shown by the black stem plots.

In the scenario depicted in Figure 1, membership is propagated by evaluating the probability of membership on the neighboring node at the time of the edge $t_{e_{ij}}$ and depositing that probability on

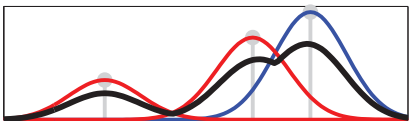

Fig. 2. Notional result from dynamic membership propagation using a Gaussian kernel function. The black curve is a weighted combination of the red and blue membership kernels as defined in Equation 2 and 3

the receiving node at $t_{e_{ij}}$. This provides a measurement of membership at a discrete set of times. In order to estimate membership probability over all time, we must make some assumptions on the dynamic process governing the community we wish to detect.

Equation 2 generalizes the membership propagation equation to allow for time varying probabilities,

$$P_i(t) = \alpha\Big(\frac{\lambda}{|E(i)|}\sum_{e_{ij}\in E(i)} g(t|e_{ij}) + (1-\lambda)\max_{e_{ij}\in E(i)} g(t|e_{ij})\Big). \quad (2)$$

where $E(i)$ represents the set of all edges connecting to node i. The function $g(t|e_{ij})$ is an application specific interaction model describing the effect of an edge between node i and j over all time. This function g can naturally be divided into two terms. The first term is a scale factor defined by the membership probability transferred from j evaluated at the time that the edge is created. The second term is a kernel function defining how the membership probability changes for times different from the edge creation time,

$$g(t|e_{ij}) = P_j\left(t_{e_{ij}}\right) K\left(t - t_{e_{ij}}\right). \quad (3)$$

Naturally, the kernel function $K(t)$ must be defined on an application specific basis as the effect of an edge on membership may change depending upon the process governing the community you wish to detect. In the example of a group of collaborating colleagues, we may expect people in the same community to be ones who attend the same meetings at the same time. Therefore, the interaction kernel could be a Gaussian function whose width is defined by the duration of a typical meeting. In disease spreading applications, a person may only be contagious 24 to 48 hours after contact with another infected person. In this situation the desired kernel may be a shifted Gaussian centered 24 to 48 hours after the edge.

As defined in Equation 2, the overall probability of membership on node i is a weighted combination of all kernel functions arising from incoming edges. This property provides a smoothly varying function of membership that depends upon edge times as well as the community membership kernel (as shown in Figure 2). Note that this formulation of community membership does not necessarily state that a node is a member of a community at one time and not another, but rather provides the probability that the node is acting as a member of that community at a given time. This is analogous to the role indicator variable Z in *mixed-membership stochastic blockmodels* by Airoldi et al. [12].

5. EMPIRICAL RESULTS

We evaluate membership propagation on a simulated vehicle movement dataset as well as the Enron email dataset [1]. Clauset's *"local modularity maximization"* [8] and a cued version of Miller's *"eigenspace analysis"* [9] are used as baselines for performance comparisons against the methods proposed in this paper. These two approaches represent a diverse range of cued community detection techniques which leverage the modularity matrix.

5.1. Data

The performances of the community detection techniques described in this paper were determined by applying them to two network datasets. For each dataset, a subset of the network is chosen as the community of interest (the "foreground") that the community detection algorithms are tasked with distinguishing from the other remaining nodes (the "background").

The first dataset is a simulation of vehicle movement over a 48 hour time period in an urban environment. The simulation was constructed by the National Geospatial-Intelligence Agency (NGA). The nodes in this network correspond to buildings at different locations within the city, and an edge between two nodes exists if a vehicle has traveled between the corresponding buildings. There are approximately 4,400 nodes and over 116,000 edges in the network. A small subset of this network corresponds to the operations of an insurgent cell that conducts activities at 31 different nodes over the course of the 48 hour period. For a more detailed discussion of this dataset see [2].

The second dataset is the Enron email corpus [1], consisting of time-stamped emails exchanged between employees at the Enron Corporation. The entire network consists of 156 nodes and 38,390 edges, where a node corresponds to an individual employee and an edge corresponds to an email sent from one employee to another. The foreground community for this network was chosen to be the 25 employees that the corpus identifies as members of the Enron legal department.

5.2. Methodology

Because methods in this paper are local (or cued) methods, performance will inherently depend on the tip into the community of interest. Depending on the location of the tip node (only one tip is used in any experiment) performance will naturally increase or decrease depending on the information contained in the tip. Therefore, detection results are calculated independently using every possible tip into the foreground. Results are then averaged over all possible tip locations. Declaration of community membership is carried out by setting a desired threshold on the membership probability. In dynamic membership propagation, that probability can vary over time. Therefore, for the purpose of making a single declaration on each node, the membership probability on each node is averaged over time. The intuition being that nodes that spend more time on average acting as a member of the community of interest are more likely to be members of that community.

5.3. Detection Performance

Figure 3 shows detection performance curves for both evaluated networks. Results on the simulated vehicle movement graph are shown in 3(a). The eigenspace and local modularity methods perform below the static and dynamic forms of membership propagation. This result is unsurprising, given that the former methods are designed

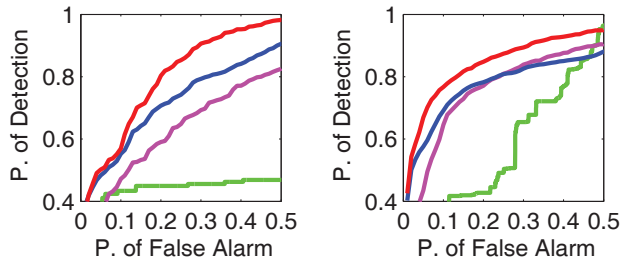

Fig. 3. Community detection results on the a) Enron email graph and b) vehicle movement graph. Plots compare community detection performance on a variety of algorithms including eigenspace detection (magenta), local modularity maximization (green), membership propagation (blue) and dynamic membership propagation (red).

to identify tightly connected static communities that are not exhibited by the topology of the datasets under study. Static membership propagation, on the other hand, shows detection performance above both baseline methods. This performance increase demonstrates the potential power of a tip node even in the absence of static structure. While methods like local modularity also use a tip node, they force a hard decision at every iteration of the algorithm. Once a bad decision is made, that bad decision compounds going forward. In contrast, membership propagation passes soft probability estimates at every iteration, postponing a decision until the end. This feature mitigates the effect of any bad decision. Finally, dynamic membership propagation shows the best performance of all. This boost above static membership propagation is due to its ability to utilize the correlations between interactions over time.

Figure 3(b) shows detection performance for the Enron email graph. This plots show similar performance to the previous results, with dynamic membership propagation having the best detection performance. Static membership propagation and eigenspace detection performance fall off due to their inability to leverage the dynamic process.

6. CONCLUSIONS AND FUTURE WORK

We present a novel perspective and approach to detect activity-based communities by propagating membership potential between neighboring nodes as they interact through time. We demonstrate its utility through a local implementation for community detection given a starting node, on two varied data sets. Performance improvement of the static membership propagation over the baseline methods demonstrates its effectiveness in utilizing information of a tip node into a community. Improvement of the dynamic membership propagation over the static version shows the benefit of using the temporal information of coordinated interactions.

In this paper, it was assumed that the temporal kernel function defining the community of interest was known *a priori*. In the case of the simulated datasets the kernel was defined by the known length of within community interactions; in the case of the Enron dataset, the kernel was defined as the inverse of the known reciprocation effect [5]. However, in many applications these community properties are not known ahead of time. In these cases, the kernel should be defined broadly given one's knowledge of a community and the specifics learned from the data itself. For example, one can calculate the average edge creation rate of the network as a whole and adjust the width of the kernel to achieve something analogous to a constant false alarm rate (CFAR).

7. REFERENCES

[1] W. Cohen, "Enron email dataset," in *http://www.cs.cmu.edu/ enron/*, 2009.

[2] S. Smith et al, "Network discovery using wide-area surveillance data," in *Proc. 14th International Conference on Information Fusion*, 2011.

[3] M. Girvan and M. Newman, "Community structure in social and biological networks," in *Proc. National Academy of Sciences of the United States of America*, 2002.

[4] B. Junker and F. Schreiber, *Analysis of Biological Networks*, Wiley-Interscience, 2008.

[5] P. Perry and P. Wolfe, "Point process modeling for directed interaction networks," in *arXiv:1011.1703v1 [stat.ME]*, 2010.

[6] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3-5, pp. 75–174, 2010.

[7] M. Newman, *Networks: An Introduction*, Oxford University Press, 2010.

[8] A. Clauset, "Finding local community structure in networks," *Physical Review E*, 2005.

[9] M. Beard B. Miller and N. Bliss, "Eigenspace analysis for threat detection in social networks," in *Fusion*, 2011.

[10] J. Kleinberg, "Authoritative sources in a hyperlinked environment," in *Proc. 9th ACM-SIAM Symposium on Discrete Algorithms*, 1998.

[11] L. Song W. Fu and E. Xing, "Dynamic mixed membership blockmodel for evolving networks," in *Proc. 26th International Conference on Machine Learning*, 2009.

[12] S. Fienberg E. Airoldi, D. Blei and E. Xing, "Mixed-membership stochastic blockmodels," *Journal of Machine Learning Research*, vol. 9, 2008.

[13] J. Ferry, "Group tracking on dynamic networks," in *Proc. 12th International Conference on Information Fusion*, 2009.

[14] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," in *Proceedings of the 7th international conference on World Wide Web (WWW)*, 1987.