Overlapping Community Detection Over Temporal Graph Sequences

Kai Sheng Tai
Stanford University
kst@cs.stanford.edu

Stephen Macke
Stanford University
smacke@cs.stanford.edu

Yifei Huang
Stanford University
yifeih@cs.stanford.edu

ABSTRACT

In this paper, we introduce the Temporal Affiliation Model (TEAM), a novel, scalable method for overlapping community detection on dynamic, temporally-evolving networks. The model jointly learns term-community affiliation strengths along with community activity levels at each time step. Our method can further cluster the discovered communities into a community hierarchy. We apply TEAM to a dynamic topic modeling task on a corpus of New York Times politics articles from 1987 to 2007. Our method successfully discovers a topical hierarchy and identifies temporal activity levels for topics that correlate well with significant historical events.

1. INTRODUCTION

The structure of many real-world networks changes dynamically over time. Connections between nodes in the network may appear, disappear, or change in strength. Examples of such networks with temporal dynamics include the network traffic graph over collections of routers [11], networks of emails between individuals [8, 16], and protein-protein interaction networks [6].

These dynamic network structures have previously been explored in a variety of settings. Guo et al. [5] and Kolar et al. [9] use observations of node attributes over time (for example, observations of gene expression levels) to learn latent, temporally-evolving network structures over the nodes. Here, the underlying network topology is unobserved. Hanneke et al. [7] study the problem of predicting the future topology of a network given the previous history of network structures. Using a discrete time series of observations of the network structure, they model the temporal evolution of network topology under a Markov assumption.

In spite of the prevalence of temporally-evolving networks in real-world data and past work on dynamic network structures, the majority of work in community detection in networks has focused on discovering communities in static networks. Comparatively little work has been done on community detection in dynamic networks that evolve over time.

In this paper, we introduce TEAM (Temporal Affiliation Model), a novel, scalable community detection method over temporal graph sequences that jointly learns both the latent community structure and the time-dependent activity of each community. TEAM is an extension of BigClam (Cluster Affiliation Model for Big Networks) [21], an overlapping community detection method for static networks. Our method is able to detect both overlapping and non-overlapping communities in the network. Our method can further cluster the set of discovered communities into a community hierarchy using hierarchical agglomerative clustering.

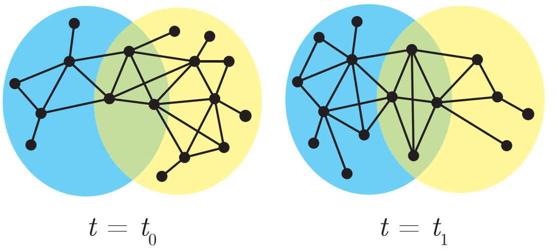

Figure 1: Overlapping communities in a dynamic network.

We evaluate the quality of the communities discovered by the network by evaluating against networks with known ground-truth communities. We find that TEAM achieves comparable performance with BigClam (as measured by average F1 score) on co-authorship networks extracted from the DBLP dataset [13].

As an application of TEAM, we model the temporal evolution of topics in a corpus of New York Times articles spanning the years 1987 to 2007. Unlike many commonly-used topic models, our method does not require any variational inference or Gibbs sampling during learning, which allows us to scale to much larger corpora. Our method successfully identifies a topical hierarchy as well as the temporal activity of each topic. Several of the topics discovered by TEAM on this corpus correlate well with significant historical events.

2. RELATED WORK

The TEAM model is closely related to BigClam [21], a bipartite affiliation network model [12, 22] for overlapping community detection. In bipartite affiliation network models, the nodes of the network and the set of community nodes comprise the two disjoint sets of vertices of a bipartite graph. The edges between network nodes and community nodes represent the affiliation of the node with the community. If the graph is weighted, the weight of the edge corresponds to the strength of the node's community affiliation. The observed edges between the nodes of the network are considered to be generated from the latent community affiliations of the nodes. Our temporal affiliation model, described in Sec. 3, is a natural extension of the affiliation model introduced in [21] to temporal graph sequences. Unlike BigClam, which is specific to unweighted graphs, TEAM takes as input graphs with positive integer edge weights.

In analogy with the non-negative matrix factorization (NMF) [14] interpretation of the BigClam method, TEAM admits a non-negative tensor factorization (NTF) [10] interpretation. The sequence of adjacency matrices over time can be regarded as a 3rd-order tensor with two "spatial" dimensions over the space of nodes and one temporal dimension. There has been previous work where the NTF of the adjacency tensor is used for analyzing community activity over time [1, 4]. However, the direct NTF of the adjacency tensor is not easily interpretable. In the following we describe a variant of NTF that admits a simple probabilistic interpretation.

3. TEMPORAL AFFILIATION MODEL

We now describe the TEAM generative model for temporal graph sequences. Let the number of communities K be given. Given a set of vertices V, at each time step $t = 1, 2, \ldots, T$, generate a weighted, undirected graph using the following process. For each pair of vertices (u, v) and each community $k = 1, 2, \ldots, K$, generate an integer $w_t^{(k)}(u, v)$ from a Poisson distribution with mean $A_{tk} F_{uk} F_{vk}$:

$$w_t^{(k)}(u, v) \sim \text{Pois}\left(A_{tk} F_{uk} F_{vk}\right), \tag{1}$$

where $F_{uk}, F_{vk} \in [0, 1]$ are the *community affiliation strengths* between community k and nodes u and v respectively. The parameter $A_{tk} \in [0, \infty)$ is the *activity* of community k at time t. The *community affiliation matrix* F is the $|V| \times K$ matrix of community affiliation strengths, and the *temporal activity matrix* A is the $T \times K$ matrix of community activities.

The observed edge weight $w_t(u, v)$ is the sum over the weights corresponding to each community:

$$w_t(u, v) = \sum_{k=1}^{K} w_t^{(k)}(u, v) \sim \text{Pois}\left(\sum_{k=1}^{K} A_{tk} F_{uk} F_{vk}\right), \tag{2}$$

since a sum over Poisson random variables is itself a Poisson random variable. If $w_t(u, v) = 0$, then there is no edge between u and v in the graph at time t. All edge weights are therefore positive integers.

This Poisson process for generating edges is similar to the process used by Wang et al. [19] in a topic modeling setting. Unlike [19], we do not impose the constraint that $\sum_u F_{uk} = 1\ \forall k$, *i.e.* that the community affiliation parameters for each community define a multinomial distribution over the set of nodes. This simplifies our optimization problem and, as we show in Sec. 6, still yields good results in topic modeling experiments.

Our model is a natural extension of the BigClam community affiliation model for static networks [21] to sequences of weighted undirected graphs. In BigClam, the (unweighted) edge between nodes u, v is generated if the corresponding Poisson random variable X_{uv} takes a positive value: $X_{uv} > 0$, $X_{uv} \sim \text{Pois}\left(\sum_{k=1}^{K} F_{uk} F_{vk}\right)$. Note that in BigClam, the community affiliation strengths F_{uk} take values in $[0, \infty)$, whereas in our model the values F_{uk} are restricted to the interval $[0, 1]$.

With this restriction, the community affiliation parameters are more interpretable by virtue of the following probabilistic interpretation. Each community affiliation parameter F_{uk} can be interpreted as the probability that node u is generated by community k in a Bernoulli trial, and each community activity parameter A_{tk} can be interpreted as the product of the number of Bernoulli trials for community k at time t (denoted N_{tk}) and the prior probability of the community at time t (denoted a_{tk}): $A_{tk} = N_{tk} a_{tk}$. To see this, consider the Poisson distribution in Eq. 1 as an approximation to a binomial distribution $\text{Bin}(N_{tk}, a_{tk} F_{uk} F_{vk})$: the edge weight $w_t^{(k)}(u, v)$ corresponding to community k is simply the number of successes after $N_t k$ independent trials. For example, consider a sequence of co-occurence graphs of terms over a corpus of documents, where edge weights are given by the number of documents in which a pair of terms co-occur. Each community corresponds to a topic in the corpus, the F parameters give the probability that a given topic generates any term, and N_{tk} is the number of documents at time t that express topic k.

As previously mentioned, the TEAM model can be viewed as a variant of non-negative tensor factorization (NTF) on the $|V| \times |V| \times T$ adjacency tensor of the observed graph sequence. However, instead of decomposing the adjacency tensor directly, as in a direct application of NTF, we instead decompose the 3rd-order tensor of Poisson mean parameters corresponding to the Poisson random variable for each pair of nodes at each time step.

4. LEARNING

4.1 Objective

The parameters F and A are learned from an input sequence of T undirected graphs $\mathcal{G} = \{G(V_1, E_1), \ldots, G(V_T, E_T)\}$ representing snapshots of a network at discrete time steps. Let $V = \bigcup_{t=1}^{T} V_t$ denote the set of vertices over the entire graph sequence.

We maximize the regularized log-likelihood of the observed graph sequence $\mathcal{G}$. From Eq. 2, the distribution of the edge weight $w_t(u, v)$ at time t between nodes u and v is:

$$\Pr[w_t(u, v) = y] = \frac{\lambda_t(u, v)^y}{y!} \exp\left(-\lambda_t(u, v)\right), \tag{3}$$

where:

$$\lambda_t(u, v) = \sum_{k=1}^{K} A_{tk} F_{uk} F_{vk}. \tag{4}$$

Therefore, the log-likelihood of the data $\mathcal{G}$ given the parameters F, A is:

$$\ell(\mathcal{G}; F, A) = \sum_{t=1}^{T} \Bigg(- \sum_{\text{pairs}(u,v)} \lambda_t(u,v) + \sum_{(u,v)\in E_t} \Big(w_t(u,v) \log \lambda_t(u,v) - \log \big(w_t(u,v)! \big) \Big) \Bigg). \quad (5)$$

When computing the log-likelihood in practice, we omit the $\log \big(w_t(u,v)!\big)$ term since it has no dependence on F or A.

In order to encourage sparsity in the F parameters, we add an L_1 regularization penalty to the objective. To encourage smoothness of community activities over time, we add an L_2 regularization penalty on the difference between temporally adjacent A parameters. The full cost function $C(F, A; \mathcal{G})$ is then given by:

$$C(F, A; \mathcal{G}) = - \ell(\mathcal{G}; F, A) + \lambda_1 \sum_{u \in V} \|F_u\|_1 + \frac{1}{2} \lambda_2 \sum_{t=1}^{T-1} \|A_{t+1} - A_t\|_2^2, \quad (6)$$

where $\ell(\mathcal{G}; F, A)$ is the log-likelihood defined in Eq. 5, and λ_1 and λ_2 are hyperparameters. In our experiments, we use $\lambda_1 = 10^2$ and $\lambda_2 = 10^4$.

4.2 Optimization

To optimize this objective, we use projected gradient descent with adaptive per-parameter learning rates computed using AdaGrad [3]. To avoid numerical issues with vanishing $\lambda_t(u,v)$ values, we set a small, nonzero minimum value ϵ for the parameters F and A. In our experiments, we set the AdaGrad parameter η to 0.1 and ϵ to 10^{-10}. After each gradient update, the F parameters are projected back onto the feasible set $[\epsilon, 1]$ and the A parameters are projected onto $[\epsilon, \infty)$.

Let $X \circ Y$ denote the pointwise (Hadamard) product and $\mathcal{N}_t(u)$ the set of neighbors of u at time t. The gradient for F_u is:

$$\nabla_{F_u} C = \sum_{t=1}^{T} A_t \circ \left(\sum_{v \neq u} F_v - \sum_{v \in \mathcal{N}_t(u)} \frac{w_t(u,v)}{\lambda_t(u,v)} F_v \right) + \lambda_1. \quad (7)$$

The gradient for A_t is:

$$\nabla_{A_t} C = \sum_{\text{pairs}(u,v)} F_u \circ F_v - \sum_{(u,v)\in E_t} \frac{w_t(u,v)}{\lambda_t(u,v)} F_u \circ F_v + \lambda_2 \Big(\mathbb{1}\{t > 1\}(A_t - A_{t-1}) + \mathbb{1}\{t < T\}(A_t - A_{t+1}) \Big) \quad (8)$$

where $\mathbb{1}\{x\}$ is the indicator function.

In each iteration, we alternate between F and A, first updating all rows of F, then updating all rows of A. The gradient for F_u can be computed in time $O\left(K \sum_{t=1}^{T} |\mathcal{N}_t(u)|\right)$ by precomputing the value of $\sum_v F_v$ at the start of each iteration, while the gradient for A_t can be computed in time $O\left(K|E_t|\right)$ by precomputing the value of $\sum_{\text{pairs}(u,v)} F_u \circ F_v$ at the start of each iteration. This quantity can be computed in time $O(K|V|)$ (in particular, without iterating over all pairs of vertices) by using the observation that:

$$\sum_{\text{pairs}(u,v)} F_u \circ F_v = \frac{1}{2} \left(\left(\sum_v F_v \right)^2 - \sum_v F_v^2 \right), \quad (9)$$

where the square is taken pointwise over the elements of each vector. Similarly, the log-likelihood can be computed in time $O\left(K \sum_{t=1}^{T} |E_t|\right)$. The total time complexity of each iteration is therefore $O\left(K\left(|V| + \sum_{t=1}^{T} |E_t|\right)\right)$. This is efficient if the input graphs are sparse, as is the case for most real-world networks.

To initialize F, we set each entry to a random value uniformly distributed over $[0.25, 0.75]$. To initialize A, we set each entry to a random value uniformly distributed over $[0.75, 1.25]$.

In our implementation, we terminate the optimization when the relative change in the objective falls below 10^{-3}. The log-likelihood is computed after every 10 iterations to check for convergence.

5. COMMUNITY HIERARCHIES

Many networks derived from real-world data exhibit a hierarchical community structure. For example, a social network of undergraduate students can be composed of communities corresponding to each class year, which in turn can be further decomposed into sub-communities corresponding to academic concentrations.

Our community detection method allows for a simple post-processing step that builds a hierarchy over the communities discovered in the learning step. Each of the K columns of the community affiliation matrix F is a vector in $[0,1]^{|V|}$. To construct a community hierarchy, we use complete-link hierarchical agglomerative clustering over the columns of F. The metric over the vector space of communities is the cosine distance. We refer to clusters of communities as *supercommunities*. The temporal activity of a supercommunity is defined as the mean of the columns of A corresponding to the communities that comprise the supercommunity. To our knowledge, this application of hierarchical agglomerative clustering is a novel method for constructing post-hoc community hierarchies.

This method is not limited to clustering the node affiliation strengths for each community. Instead, each community can be represented by a linear combination Z_k of an F column and an A column:

$$Z_k = \alpha F_{*k} + (1 - \alpha) A_{*k}, \quad (10)$$

where $\alpha \in [0,1]$ is a parameter that trades off between the "spatial" F component and the "temporal" A component. We can therefore construct spatio-temporal hierarchies that can be used, for example, to study which communities tend to co-occur in time. In our experiments, we only consider fully spatial clustering ($\alpha = 1$), but spatio-temporal community hierarchies are nonetheless an interesting line of future work.

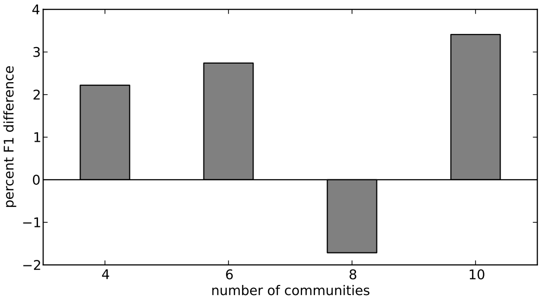

Figure 2: Percentage difference in average F1 of communities discovered by TEAM relative to Big-Clam on subsampled DBLP co-authorship graphs.

6. EXPERIMENTS

6.1 Evaluation using Ground-Truth

We evaluate the communities detected by TEAM by comparing them against ground-truth communities. Specifically, we use the co-authorship network derived from the DBLP dataset [13], where ground-truth communities correspond to conferences and journals.

In our evaluation, we use DBLP journal and conference data from 1974 to 2004. The full co-authorship network is subsampled using a similar subsampling technique as described in [21]. We select a random author u who belongs to at least 2 communities, and create a subgraph by sampling only other authors v that share at least one ground truth community with u. Furthermore, to ensure a connected graph, only those v that belong to the connected component with u are subsampled. Note that TEAM only takes a single graph as input (as opposed to a graph sequence) in our comparisons with BigClam.

We evaluate the discovered communities using the average F1 score. Given the ground truth communities $C_i \in C^*$ and the set of detected communities $\hat{C}_i \in \hat{C}$, the average F1 score is:

$$\mathrm{F1}_{\mathrm{avg}}(C^*, \hat{C}) = \frac{1}{2}\Bigg(\frac{1}{|C^*|}\sum_{C_i \in C^*} F1\left(C_i, \hat{C_{g(i)}}\right) \tag{11}$$

$$+\frac{1}{|\hat{C}|}\sum_{\hat{C}_i \in \hat{C}} F1\left(C_{g'(i)}, \hat{C}_i\right)\Bigg), \tag{12}$$

where:

$$g(i) = \arg\max_j \mathrm{F1}(C_i, \hat{C}_j), \tag{13}$$

$$g'(i) = \arg\max_j \mathrm{F1}(C_j, \hat{C}_i). \tag{14}$$

We first compare our results with those of BigClam [21]. Since it is inherently easier to discover communities in certain subsampled graphs, we calculate the percent difference in average F1 scores between our model and BigClam: $(\mathrm{F1}_{\mathrm{avg}}^{\mathrm{TEAM}} - \mathrm{F1}_{\mathrm{avg}}^{\textsc{BigClam}})/\mathrm{F1}_{\mathrm{avg}}^{\textsc{BigClam}}$. We evaluated on different numbers of ground-truth communities, sampling 20 graphs for each community count.

Term	Affiliation Score
students	0.944695
school	0.942791
education	0.939966
religious	0.937476
schools	0.927772
court	0.915907
religion	0.900700
bush	0.679902
science	0.615520
evolution	0.571594

Table 1: Terms and affiliation scores for an example community.

As evident in figure 2, the percent differences between our model and BigClam are very small (2-3%), suggesting that our model discovers communities of comparable quality.

6.2 Dynamic Topic Modeling

Statistical topic models such as Latent Dirichlet Allocation [2] discover distributions over words — *topics* — that are posited to generate terms in the documents that comprise a corpus. *Dynamic* topic models [20, 18] incorporate temporal information in the form of document timestamps to infer latent topics and their evolution over time. Unfortunately, these probabilistic latent variable models suffer from issues of scalability, since each iteration of learning (for example, using the EM algorithm) requires approximate probabilistic inference via variational inference or Gibbs sampling to estimate the expected distribution over the variables of the model.

We attempt to address the problem of scalability by formulating the problem of dynamic topic modeling as a temporal community detection problem over co-occurrence graphs of terms. Each term co-occurrence graph is derived from documents in a corpus that fall in a given time slice, where the edge weight between two terms is the number of documents in which the two terms co-occur.

We apply our method to politics news articles (*i.e.*, articles human-annotated with the tag "politics") from the New York Times Annotated Corpus [17], which consists of 20 years of New York Times newswire from 1987 to 2007. The entire 16 GB dataset contains roughly 160,000 politics articles.

We generate co-occurrence graphs with a one-week temporal resolution. For each document in the time slice, the top-20 terms ranked by tf-idf score are used to construct the co-occurrence graph. This results in a sequence of 1,067 graphs over which we perform the optimization. Finding 128 communities takes roughly 5 hours using 16 cores.

After convergence, we have a term-community affiliation score in the interval $[0, 1]$ for each term in our vocabulary, and for each community detected. We then run a post-processing step to extract and score representative phrases for each community.

6.2.1 *Phrase Extraction and Scoring*

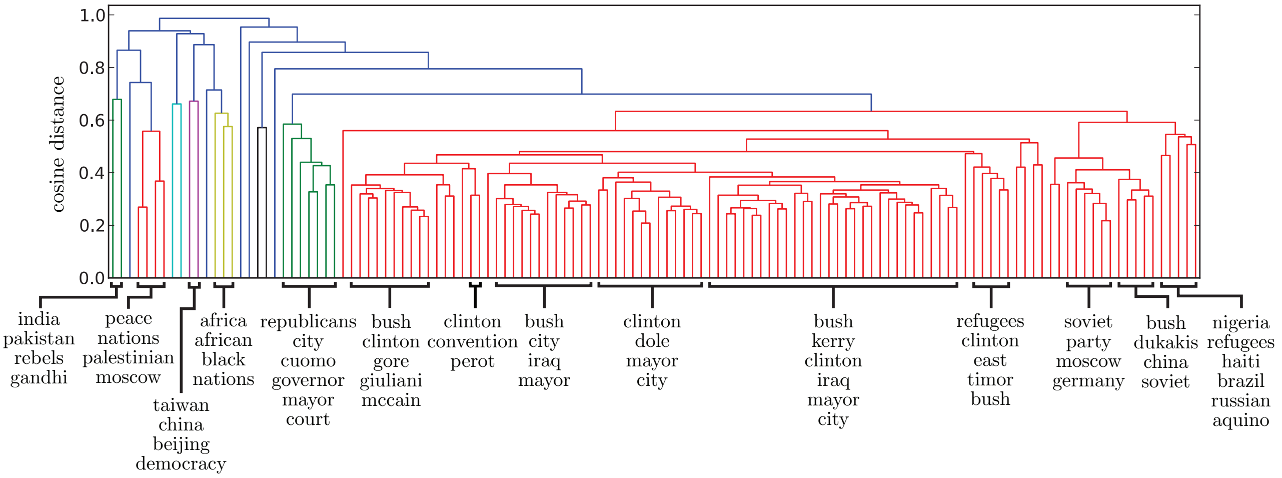

Figure 3: A community hierarchy over topics expressed in New York Times politics articles. Communities are clustered using complete-link hierarchical agglomerative clustering with cosine distance.

The top-ranked terms (in terms of term-community affiliation strength) provide a concise way to represent each discovered community. Unfortunately, communities are not always easily interpretable from their representative terms alone. As an example, consider the top ten terms shown in Table 1. This topic (community) pertains to the debate over the teaching of evolution in schools. However, individuals unfamiliar with U.S. politics may find it difficult to deduce this interpretation given only the list of terms.

To improve the interpretability of the discovered topics, we represent each topic by a ranked list of phrases. Using the term-community affiliation scores learned by TEAM, we rank phrases extracted directly from documents in the corpus. Each phrase P is represented as a bag-of-words and scored for each community k as follows:

$$\text{score}(P) = (P \cdot F_{*k})(1 + \log|P|), \tag{15}$$

where F_{*k} represents the term-community affiliation vector for community k, and the sublinear scaling term $(1+\log|P|)$ rewards longer phrases, which should be more informative than shorter ones.

The phrases themselves can be selected in any number of ways, such as frequent pattern mining, named entity extraction, or noun phrase extraction. Similar to the approach taken in [15], we extract noun phrases by first performing part-of-speech tagging, then selecting consecutive terms whose tags match the following pattern:

(Adj | Noun)* (Noun Prep)? (Adj | Noun)* Noun.

The top ranked phrases extracted for the evolution topic (Table 1) are listed in Table 2.

Phrase	Score
religious school education	1.972859
private religious school education	1.682477
students from religious schools	1.676338
religious education in schools	1.673517
religious students	1.593396
teaching evolution classroom	1.467496

Table 2: Phrases extracted for the community in Table 1

6.2.2 *Analysis of Temporal Topical Activities*

We now examine the temporal activities of several sample topics and explore our results given the historical context for the time period covered by the New York Times corpus. Fig. 4 depicts the community activities for a topic represented by phrases such as `religious conservative republicans`, `abortion rights amendment`, and `conservative supreme court`. Here, we observe that the large activity spike in early 2005 coincides with the introduction of a large amount of abortion-limiting legislation.

Next, Fig. 5 depicts the learned activities for a topic represented by phrases such as `united nations security council resolution on iraq` and `united nations weapons inspectors`. For this topic, the large spike in early 2002 occurs shortly after the September 11 terrorist attacks. Around this time, George W. Bush began efforts to enact sanctions against Iraq over the alleged development and possession of weapons of mass destruction.

As a final example, consider Fig. 6, which depicts temporal topical activities for a topic represented by phrases such as `east germany` and `east german communist party`. Here, the spike in activity around 1990 coincides with the fall of the Berlin Wall. We also observe a minor spike which occurs much later in the activity plot. This smaller spike in activity coincides with the publication of an article describing a memorial at the site of the original wall, commemorating its destruction as a symbolic “lifting of the Iron Curtain”.

6.2.3 *Analysis of Topical Hierarchy*

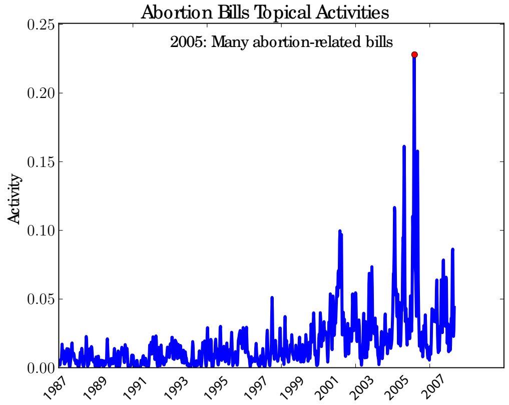

Figure 4: Activities for abortion legislation community

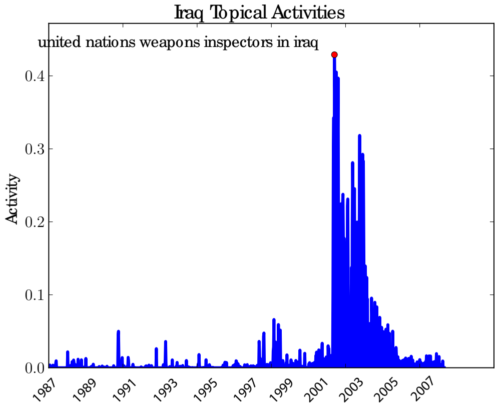

Figure 5: Activities for Iraqi weapons inspection community

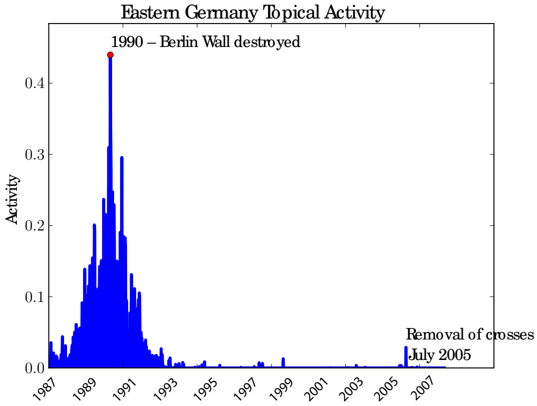

Figure 6: Activities for East Germany community

Fig. 3 depicts a topical hierarchy constructed using the aforementioned complete-link agglomerative clustering method. Broadly speaking, the left-most hierarchies correspond with international news. To the right of this, we have more U.S.-centric articles. In particular, the green clusters coincide with news local to New York City. This topical hierarchy provides an easily interpretable global view of the topics expressed in the corpus.

7. CONCLUSION

In this paper, we introduce a novel community detection method, TEAM, for temporal graph sequences. A promising application of the method is dynamic topic modeling on large corpora, as evidenced by our analysis of topical hierarchies in a corpus of New York Times articles.

Further evaluation of the method on other datasets with ground truth is needed. In particular, it would be useful to compare TEAM with other methods in terms of *(a)* evaluation on ground-truth data, and *(b)* speed and algorithmic complexity. The aforementioned quantities could be used to evaluate any potential tradeoffs when including temporal information for the purpose of community detection.

Next, for datasets for which ground truth is not available, user-studies on phrase representations of the extracted topical clusters for qualities such as completeness, purity, and phraseness (described further in [19]) would help further evaluate the benefit of our method as a tool for information retrieval.

In addition, further exploration is needed regarding the time-space tradeoff in construction of spatio-temporal hierarchical clusters. In this paper, we only consider agglomerative construction of hierarchies on the spatial components of bottom-level clusters.

Finally, when performing phrase extraction, we currently do not use the term-community activities in any way for phrase ranking. Incorporating such information could help improve the rankings of phrases more salient to the particular topic(s) under consideration.

8. ACKNOWLEDGMENTS

We thank Rok Sosic and Jure Leskovec for helpful discussions and suggestions over the course of this project.

9. REFERENCES

[1] B. W. Bader, R. A. Harshman, and T. G. Kolda. Temporal analysis of semantic graphs using asalsan. In *Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on*, pages 33–42. IEEE, 2007.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *the Journal of Machine Learning Research*, 3:993–1022, 2003.

[3] J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *The Journal of Machine Learning Research*, 12:2121–2159, 2011.

[4] L. Gauvin, A. Panisson, and C. Cattuto. Detecting the community structure and activity patterns of

temporal networks: a non-negative tensor factorization approach. *PLOS ONE*, 9(1):e86028, 2014.

[5] F. Guo, S. Hanneke, W. Fu, and E. P. Xing. Recovering temporally rewiring networks: A model-based approach. In *Proceedings of the 24th International Conference on Machine Learning*, pages 321–328. ACM, 2007.

[6] J.-D. J. Han, N. Bertin, T. Hao, D. S. Goldberg, G. F. Berriz, L. V. Zhang, D. Dupuy, A. J. Walhout, M. E. Cusick, F. P. Roth, et al. Evidence for dynamically organized modularity in the yeast protein–protein interaction network. *Nature*, 430(6995):88–93, 2004.

[7] S. Hanneke, W. Fu, E. P. Xing, et al. Discrete temporal models of social networks. *Electronic Journal of Statistics*, 4:585–605, 2010.

[8] B. Klimt and Y. Yang. The enron corpus: A new dataset for email classification research. In *Machine Learning: ECML 2004*, pages 217–226. Springer, 2004.

[9] M. Kolar, L. Song, A. Ahmed, E. P. Xing, et al. Estimating time-varying networks. *The Annals of Applied Statistics*, 4(1):94–123, 2010.

[10] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009.

[11] A. Lakhina, K. Papagiannaki, M. Crovella, C. Diot, E. D. Kolaczyk, and N. Taft. *Structural analysis of network traffic flows*, volume 32. ACM, 2004.

[12] S. Lattanzi and D. Sivakumar. Affiliation networks. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing*, pages 427–434. ACM, 2009.

[13] M. Ley. The dblp computer science bibliography: Evolution, research issues, perspectives. In *String Processing and Information Retrieval*, pages 1–10. Springer, 2002.

[14] C.-J. Lin. Projected gradient methods for nonnegative matrix factorization. *Neural computation*, 19(10):2756–2779, 2007.

[15] E. Loper and S. Bird. Nltk: The natural language toolkit. In *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics - Volume 1*, ETMTNLP '02, pages 63–70, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.

[16] R. K. Pan and J. Saramäki. Path lengths, correlations, and centrality in temporal networks. *Physical Review E*, 84(1):016105, 2011.

[17] E. Sandhaus. The new york times annotated corpus ldc2008t19, 2008.

[18] C. Wang, D. Blei, and D. Heckerman. Continuous time dynamic topic models. *arXiv preprint arXiv:1206.3298*, 2012.

[19] C. Wang, M. Danilevsky, N. Desai, Y. Zhang, P. Nguyen, T. Taula, and J. Han. A phrase mining framework for recursive construction of a topical hierarchy. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 437–445. ACM, 2013.

[20] X. Wang and A. McCallum. Topics over time: a non-markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 424–433. ACM, 2006.

[21] J. Yang and J. Leskovec. Overlapping community detection at scale: a nonnegative matrix factorization approach. In *Proceedings of the sixth ACM International Conference on Web Search and Data Mining*, pages 587–596. ACM, 2013.

[22] E. Zheleva, H. Sharara, and L. Getoor. Co-evolution of social and affiliation networks. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1007–1016. ACM, 2009.