Spatial Compactness meets Topical Consistency: Jointly modeling Links and Content for Community Detection

Mrinmaya Sachan Avinava Dubey Shashank Srivastava Eric P. Xing Eduard Hovy
School of Computer Science
Carnegie Mellon University
{mrinmays, akdubey, shashans, epxing, hovy}@cs.cmu.edu

ABSTRACT

In this paper, we address the problem of discovering topically meaningful, yet compact (densely connected) communities in a social network. Assuming the social network to be an integer-weighted graph (where the weights can be intuitively defined as the number of common friends, followers, documents exchanged, etc.), we transform the social network to a more efficient representation. In this new representation, each user is a bag of her one-hop neighbors. We propose a mixed-membership model to identify compact communities using this transformation. Next, we augment the representation and the model to incorporate user-content information imposing topical consistency in the communities. In our model a user can belong to multiple communities and a community can participate in multiple topics. This allows us to discover community memberships as well as community and user interests. Our method outperforms other well-known baselines on two real-world social networks. Finally, we also provide a fast, parallel approximation of the same.

Categories and Subject Descriptors

H.2.8 [**Information Systems**]: Data Mining; G.3 [**Probability and Statistics**]: [Probabilistic Algorithms]

Keywords

Community Detection, Social Networks, Graphical Models

1. INTRODUCTION

Social Networks allow users to stay in touch with friends, relatives and other acquaintances wherever they are in the world. At the same time, they unite users with common interests and/or beliefs through various groups and pages. These groups allow users to share ideas, pictures, posts, activities, events, etc. with other users. This makes them an invaluable source of heterogeneous data that can be exploited to discover relationships among groups of people. An important problem associated with discovering relationships among users in a social network is the automated discovery of communities. A community is a collection of users as a group such that there is a higher degree of 'similarity' among users in the group as against people across groups.

The notion of 'similarity' between users is subjective and has been addressed differently in previous works. Preliminary paradigms treated communities as densely connected components in the network. They postulated that real networks are not random, but display inhomogeneities leading to a high level of order and organization with high concentrations of edges within special groups of vertices (communities), and low concentrations between these groups [4]. This is often referred to as the notion of "compactness" in community structure. The notion of compactness is important as it reflects a higher level of interaction and social vicinity among members of a community and allows the interpretation of the global organization of the social network as the coexistence of their local structural sub-units (communities) associated with more highly interconnected parts. Such an interpretation is also desirable for explaining local cliques that people find themselves in their daily lives, for example in people's personal relationships, in corporate organizations and in scientific research activities.

Using this interpretation of communities, methods like cut-based graph partitioning[6], local agglomerative/divisive clustering[4], centrality based[7] and Clique percolation methods (CPM)[16] have been suggested to optimize different notions of compactness. While these works gained rapid popularity, a drawback in most of these models (with exceptions) is that they involve a hard partitioning of nodes, and do not allow users to have memberships in multiple communities. However, today, it is understood that users in a social network rarely belong to only one community: for instance the same individual usually has family, friend, and colleague affiliations in different social circles.

Further, in the context of social networks, communities have become synonymous to interest groups providing people interested in similar topics common forums. In fact, these ideas have now found explicit existence and dedicated nomenclature in Google+ "circles" or Facebook "smart lists". Finally, early works that only considered the underlying graph structure were also handicapped as they did not account for node properties such as user interests which can be easily gauged from the text associated with the network.

On the other end of the spectrum, there have been purely user-content based works to cluster users into communities like CUT[24]. In fact, neither information alone is satisfactory in determining the community memberships: the link

WSDM'14, February 24–28, 2014, New York, New York, USA.

http://dx.doi.org/10.1145/2556195.2556219.

information is often sparse and noisy and results in a poor partition of networks. On the other hand, the irrelevant content attributes significantly mislead the process of community detection. It is therefore important to combine link and content information for community detection in networks. At the same time it is also important to retain the notion of compactness as we wish to exploit the inhomogeneous structure of social networks and discover individuals contained in communities that are not just topically similar but are closely knit and share communications.

Several mixed-membership models for community discovery have been proposed to incorporate structural and/or user-interest information in social networks. A significant reason for the success of these models is that they enable us to incorporate domain knowledge by modeling a complex set of factors responsible for generating the data as latent variables. Inferring the posterior often allows the discovery of unseen structures and intricate relations between entities. For example, social network users are often modeled as having distributed membership over a set of communities or a set of topics (user-interests), and a community itself as a distribution over topics (community-interests).

Broadly, all of these models fall under one of two modeling paradigms. The first paradigm (MMSB[1], Link-PLSA-LDA[13], Topic-Link LDA[11] and RTM[3]) draws a vector of community memberships for the two candidate entities (documents or users) and typically makes a Bernoulli draw for a binary link between them using a notion of similarity between the community membership vectors. These methods directly model the adjacency matrix leading to a $\Omega(U^2)$ latent variables, where U is the number of users (nodes) in the network. This can be too large for real-world social networks. The second paradigm (CART[17] and various TURCM models[19]) implicitly assumes a link between the sender and the recipients for every post in the social network and extends the author-topic model[18] to model community memberships for the sender-recipient pair for every message. This significantly increases the number of parameters to be estimated to $\Omega(U^2C)$ where C is the number of communities. This too, can be large for real-world social networks.

Note that CUT[24] purely models content and MMSB[1] purely models linkage. CART[17] and TURCM models[19] claim to combine both but do not model compactness. [1]

Further, while the first paradigm assumes that users that are interconnected share similar interests, the second paradigm posits that users that actively talk about similar topics are connected with each other. Neither is always true in the real world. In real world social networks, there are users who are members of communities but do not actively contribute in them. Also, there are members who may be vocal about the same topics but may not be connected at all. As we will later see in our evaluations on real-world datasets, previously suggested latent variable models following the two paradigms do not conform to the notion of compactness in social networks. This often leads to cases when the individuals contained in a community may not share much communication and, as such, may not actually reflect a community in the 'traditional' sense. Finally, all the aforementioned latent variable models work on binary (link or no-link) networks.

In this paper, we propose a framework to use integer-valued weights in the networks. We believe that this is intuitive in a real-world setting where the strength of interaction between users can be measured as the number of common friends or followers, number/length of documents exchanged, number of comments or likes on same thread, etc. We represent the integer-weighted network efficiently as a bag of users; each user is herself represented as a bag of her neighbors. We use this representation to present a mixed membership model that can discover compact communities. Next, we extend both the representation and the model to incorporate the content information leading to compact yet topically consistent communities. We believe that with our approach, it would be possible to "zoom" into the network and uncover not only the communities but also the interests of the communities and its members. Hence, it would provide a tool to identify and interpret local organizations in large networks that are not only topologically compact (based on graph structure) but also semantically aligned (based on topics discovered). We show improvements over previous methods in terms of held-out perplexity, several compactness metrics and on a link prediction task.

As another contribution, we also provide a faster, efficient parallel implementation of the method and illustrate its speed-up over the sequential counterpart.

2. NOTATION

Let a (text associated) social network (TSN) be defined as a network where every edge is associated with a set of text documents. Let $\mathbf{U}$ be the set of users (nodes) in the TSN. Let $\mathbf{N_u}$ be the set of neighbors of user (sender) $u \in \mathbf{U}$. For a given user $u \in \mathbf{U}$ and its neighbor $r \in \mathbf{N_u}$, let S_{ur} be the strength of interaction between u and r and $\mathbf{D_{ur}}$ be the set of documents sent by u to r. Then, $\mathbf{D_u} = \cup_{r \in \mathbf{N_u}} \mathbf{D_{ur}}$ is the set of all documents authored by u. Let $\mathbf{W_d}$ be the multi-set of words in a given document d. We note that a document can be sent to multiple users (recipients) at the same time. Let $\mathbf{R_d}$ be the set of recipients for a given document d. Let the vocabulary set be $\mathbf{V}$. Also, let the cardinality of all these sets (in boldface) be represented by their corresponding capitalized symbols. Overall, let Δ_u be the (weighted) degree of node u where $\Delta_u = \sum_{r \in \mathbf{N_u}} S_{ur}$. We experimented with various definitions of S_{ur} as discussed before and found that $S_{ur} = D_{ur}$ i.e. setting strength of interaction as the number of documents exchanged between the users works best. Assume this setting henceforth.

3. NET-LDA: THE NETWORK MODEL

The idea of transforming the input representation for a better task-specific performance is not new[9]. A well-known example of this idea is the bag-of-words representation for a document, in which each document is seen as an exchangeable collection of words. This representation has proven to be effective in various NLP tasks such as topic modeling. We propose an alternate representation to model the link structure of a social network that explicitly takes into account the entire neighborhood of every user. By doing so, we can better model the inhomogeneity of structure in social networks and lead to more compact communities.

[1]Also note that other bayesian models like RTM[3], Link-PLSA-LDA[13] and Topic-Link LDA[11] have also been proposed for document networks (citation networks and co-authorship networks) but they are not directly extendable to social networks in their full generality.

As the name suggests, Net-LDA closely follows the modeling scheme in LDA [2] to model the network structure. To model neighborhoods in the underlying graph, we translate the network to a document corpus and argue that an LDA model on the newly constructed corpus discovers communities in the social graph just as LDA discovers topics in a document corpus. Each user in the network is represented by a document in this alternate representation and the list of all the users interacting with it forms the words in the document (replicated as many times as the integer weight of the edge connecting them). Formally, let every user u be characterized by a network interaction footprint (NIF), a document representing all its interactions with other users in the network: $NIF(u) = \cup_{r \in \mathbf{N_u}} \{r, r, \ldots S_{ur} \ times\}$ and $\cup$ represents the multi-set union (i.e. each interacting user $r \in \mathbf{N_u}$ is represented S_{ur} times in the NIF document). For example, in a dummy network where user A sends two documents to user B, three documents to user C (and no more documents to other users), the network interaction footprint of A (document representing A) is $[B, B, C, C, C]$. Note that because of this transformation, we only need $\theta(\sum_u \Delta_u)$ space to represent the graph (instead of $\Omega(U^2)$).

The network interaction footprints of the users are represented as random mixtures over latent community variables. Each community is in turn defined as a distributions over the interacting users r. This idea is analogous to LDA where the network interaction footprint is a document, interactions are words in that document and communities are latent topics. Just as LDA clusters similar words into one topic, this extrapolation would cluster users with similar neighborhoods in one community. This can also be seen as soft clustering (e.g. Soft K-Means) on the adjacency matrix of the graph. This makes the communities detected by Net-LDA "compact". Technically, this occurs due to the sparsity induced by the Dirichlet prior. The sparsity through the user-community Dirichlet prior penalizes the model for using many topics for a document. This ensures that all the neighbors of a node have slightly higher odds to belong to the same community. This, in turn, makes the community structure spatially "compact". On the other hand, the sparsity induced by the community-neighbor Dirichlet ensures that the user (document) is assigned to only a few communities (topics). This conforms to real world scenarios where users are typically members of a small number of communities. Note that this model has $\theta(\sum_u \Delta_u)$ latent variables, which is much smaller than U^2 in models following paradigm 1 (e.g. MMSB) for real-world (sparse) networks. Further, this model has $\theta(UC)$ parameters which can be efficiently estimated and is a reduction over $\Omega(U^2)$ parameters as in models following Paradigm 2.

4. SN-LDA:THE SOCIAL NETWORK MODEL

In the social network literature, previous works[8] have shown success in combining local (node-based) features with neighborhood features for various graph mining tasks. In order to account for the spatial as well as topic-coherence of communities discovered, we represent the TSN as a collection of users, where each user is represented as a bag of one hop neighbors and a bag of documents authored by the user.

We further augment Net-LDA with content information. To achieve this, we associate every user with a multinomial distribution π over communities. This distribution gives the membership of the user in every community. We define a topic to be a semantic theme represented as a multinomial distribution ϕ over words. We associate topics with the communities and hence model communities as a multinomial distribution θ over the topics of interest amongst users in the community. This can help us determine key topics prevailing in a community and can be useful in locating/searching communities on given themes and interests. Just as in Net-LDA, we represent the user as a distribution π over the community space and communities as a distribution η over the space of interacting users. The plate notation is given in Figure 1 and generative story below. Here $Dir_X(\alpha)$ denotes a X-dimensional symmetric Dirichlet with scalar parameter α and $Mult(.)$ denotes the discrete multinomial. The sender, neighbor (recipient) and words are observed. The communities and topics are latent.

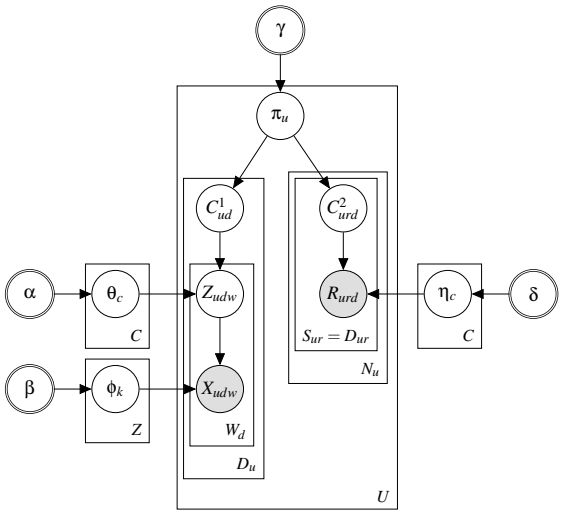

Figure 1: Plate diagram of SN-LDA

1. For each topic index, $1 \leq k \leq Z$, sample a V dimensional multinomial, $\vec{\phi}_k \sim Dir_V(\beta)$.

2. For each community index, $1 \leq c \leq C$:

   (a) Sample a U dimensional multinomial $\vec{\eta}_c \sim Dir_U(\delta)$.

   (b) Sample a Z dimensional multinomial $\vec{\theta}_c \sim Dir_Z(\alpha)$.

3. For each user $u \in \mathbf{U}$, sample a C dimensional multinomial, $\vec{\pi}_u \sim Dir_C(\gamma)$.

4. For each user $u \in \mathbf{U}$:

   (a) For each document $d \in \mathbf{D_u}$ sent by sender u:

      i. Draw community assignment $C^1_{ud} \sim Mult(\vec{\pi}_u)$.

      ii. For every word $w \in \mathbf{W_d}$:

         - Draw topic assignment $Z_{udw} \sim Mult(\vec{\theta}_{C^1_{ud}})$.
         - Draw word $X_{udw} \sim Mult(\vec{\phi}_{Z_{udw}})$.

   (b) For each neighbor $r \in \mathbf{N_u}$:

      i. For each document $d \in \mathbf{D_{ur}}$:

         - Draw community assignment $C^2_{urd} \sim Mult(\vec{\pi}_u)$.
         - Draw neighbor $R_{urd} \sim Mult(\vec{\eta}_{C^2_{urd}})$.

We hypothesize that while a document is assigned to a particular community, the recipients of the document might still belong to different communities. This necessitates the community variables for the document c^1 and the recipients c^2 to be separate. To test this hypothesis, we design another graphical model (we call it Naiive SN-LDA) where the community variable is only drawn once for each document. The generative story of Naiive SN-LDA can be obtained by replacing 4(b) with 4(a)-iii in SN-LDA as follows:

4(a)-iii. For each recipient $r \in \mathbf{R_d}$:

- Draw recipient (neighbor) $R_{urd} \sim Mult(\vec{\eta}_{C^1_{ud}})$.

We will see in our evaluations that Naiive SN-LDA doesn't do well in practice. This substantiates our hypothesis and justifies the aforementioned modeling choice.

4.1 Parameter Estimation

We sample the topic and community assignments from the conditional for the variable given the observations and other assignments using a Gibbs sampling procedure. Let C^1_{ud}, C^2_{urd}, Z_{udw}, R_{urd} and W_{udw} be the community, topic, recipient and word assignments for a given user, document and word, as appropriate. Let $\#^{B_1 \dots B_v A}_{b_1 \dots b_v a, -i}$ be the number of times $a, (1 \le a \le A)$ is generated along with the combination of variables $b_1 \dots b_v$ in the model, $(1 \le b_i \le B_i, 1 \le i \le v)$ excluding the instance i. Also, $\#_{cz,ud}$ be the number of times topic z is assigned to words in community c for user u in document d. The Gibbs sampling equations are given below.

$$P(C^2_{urd} = c^* | \boldsymbol{C^2_{-urd}}, -) \propto \left(\#^{UC^2}_{uc^*,-urd} + \#^{UC^1}_{uc^*} + \gamma\right) \frac{\#^{C^2R}_{c^*r,-urd} + \delta}{\Sigma_{r'} \#^{C^2R}_{c^*r',-urd} + U\delta} \quad (1)$$

$$P(C^1_{ud} = c^* | \boldsymbol{C^1_{-ud}}, -) \propto \left(\#^{UC^1}_{uc^*,-ud} + \#^{UC^2}_{uc^*} + \gamma\right) \times \frac{\prod_{z \in \mathbf{Z}} \prod_{i=0}^{\#_{c^*z,ud}} (\#^{C^1Z}_{c^*z,-ud} + \alpha + i)}{\prod_{i=0}^{W_d - 1} (\Sigma_{z'} \#^{C^1Z}_{c^*z',-ud} + Z\alpha + i)} \quad (2)$$

$$P(Z_{udw} = z^* | \boldsymbol{Z_{-udw}}, \boldsymbol{C^1_{ud} = c^*}, \boldsymbol{C^1_{-ud}}, -) \propto \frac{\#^{CZ}_{c^*z^*,-udw} + \alpha}{\Sigma_{z'} \#^{CZ}_{c^*z',-udw} + Z\alpha} \times \frac{\#^{ZW}_{z^*w,-udw} + \beta}{\Sigma_{w'} \#^{ZW}_{z^*w',-udw} + V\beta} \quad (3)$$

A single iteration of the sampler performs $\mathcal{O}(DC + WZ)$ computations to model the texts (Let D be the total number of documents and W be the total number of words in the TSN) and $\mathcal{O}(C \sum_u \sum_{r \in \mathbf{N_u}} S_{ur})$ computations to model the network. Hence, the sampler has a worst time complexity of $\mathcal{O}[(DC + WZ) + C \sum_u \sum_{r \in \mathbf{N_r}} S_{ur}]$. The complexity could be significantly high even for moderate size TSNs. Note that this is a longstanding drawback of latent variable models.

5. PARALLEL SN-LDA

The Gibbs Sampling procedure utilized by our model makes multiple passes at the data and hence, does not scale to real-world data sizes. An obvious solution to address this issue is to distribute the learning over multiple processors. Next, we provide a parallel implementation of SN-LDA that has theoretical guarantees of convergence and also does well in practice.

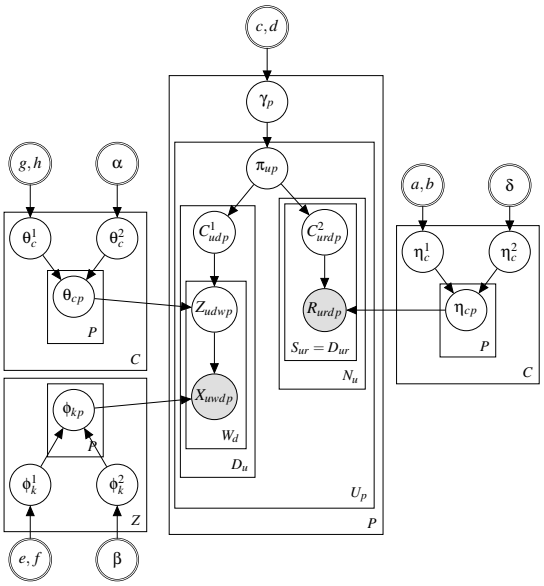

Figure 2: Plate diagram of Parallel SN-LDA

The approach for the parallel variant (inspired from Distributed LDA[14]) is to split the data (social network) roughly equally into P processors. Each processor is roughly (randomly) assigned U/P users (along with their network interaction footprints and all the documents authored by the users). All the sufficient statistics in the previous Gibbs sampler ($\#^{B_1 \dots B_v A}_{b_1 \dots b_v a, -i} \forall a, b_j, A, B_j, i$) are also distributed with each processor keeping a local copy. Moreover, the parallel version also keeps a copy of the four distributions θ, ϕ, π and η on each processor. Appropriate priors are introduced to bind these distributions globally. For instance, a global hierarchy of word-topic distributions ϕ^2_k and a topic dependent strength parameter ϕ^1_k is placed over the local word topic distributions ϕ_{kp}. Likewise, θ, π *and* η are also distributed. The model on each processor essentially is an SN-LDA model. Figure 2 shows the Bayesian model reflecting this structure. The generative process for the model is:

$\eta^1_c \sim Gamma(a, b)$ $\quad$ $\eta^2_c \sim Dir(\delta)$ $\quad$ $\eta_{cp} \sim Dir(\eta^1_c \eta^2_c)$
$\theta^1_c \sim Gamma(g, h)$ $\quad$ $\theta^2_c \sim Dir(\alpha)$ $\quad$ $\theta_{cp} \sim Dir(\theta^1_c \theta^2_c)$
$\phi^1_k \sim Gamma(e, f)$ $\quad$ $\phi^2_k \sim Dir(\beta)$ $\quad$ $\phi_{kp} \sim Dir(\phi^1_k \phi^2_k)$
$\gamma_p \sim Gamma(c, d)$ $\quad$ $\pi_{up} \sim Dir(\gamma_p)$
$C^1_{udp} \sim \pi_{up}$ $\quad$ $Z_{udwp} \sim \theta_{C^1_{udp}}$ $\quad$ $X_{udwp} \sim \phi_{Z_{udwp}}$
$C^2_{urdp} \sim \pi_{up}$ $\quad$ $R_{urdp} \sim \eta_{C^2_{urdp}}$

Pseudocode of Parallel SN-LDA is given in Algorithm 1. Processors concurrently perform Gibbs sampling on the local data partition followed by a global update of the counts. Local variables C^1_{udp}, C^2_{urdp} and Z_{udwp} are inferred locally and the global variables γ_p, θ^1_c, θ^2_c, η^1_c, η^2_c, ϕ^1_k and ϕ^2_k are then inferred globally after the local processing finishes on all processors. Here, we employ an auxiliary variable method [5] to sample both the local and global variables. Please refer to the appendix for the Gibbs sampling equations.

Algorithm 1: Parallel SN-LDA Implementation

```
Initialize: All Local and Global Variables;
while !(Termination Condition) do
    for each processor p in parallel do
        Sample c_1, c_2, z locally using the local data partition
        Sample γ_p locally;
    end
    Synchronize;
    Sample global variables θ_c^1, θ_c^2, η_c^1, η_c^2, φ_k^1 and φ_k^2;
    Broadcast global variables θ_c^1, θ_c^2, η_c^1, η_c^2, φ_k^1 and φ_k^2;
end
```

The average case time complexity of one iteration of the Gibbs sampler (both local and global updates) is $\mathcal{O}[(X + M)/P + M]$ where $X = DC + WZ + C\sum_u \sum_{r \in \mathbf{N_u}} S_{ur}$ and $M = ZW + CU + CZ$. This is an improvement over the sequential implementation when $P \geq \mathcal{O}(\frac{X+M}{X-M})$. Note that this is an average case analysis and assumes that all the processes run at the same speed and the data is uniformly (and roughly equally) divided across machines. Our evaluations show that the technique works well in practice.

6. EVALUATION

6.1 Datasets

In our experiments, we perform a comparative evaluation of Net-LDA and SN-LDA against baseline models (CUT, TUCM, TURCM-1, TURCM-2, Full TURCM, CART and MMSB) on two real-world datasets: first, is the publicly available Enron Email corpus[2] and secondly, a subset of the collection of tweets crawled from Twitter[3] over a 7 month period (June 1 - Dec 31, 2009). The Enron dataset contains email exchanges from about 150 employees, mostly senior management. On the other hand, Twitter is a social networking and micro blogging service where users communicate by short text messages (up to 140 characters) called 'tweets'. Follower relationships impose an underlying graph structure. The Twitter dataset used has 5405 nodes, 13214 edges and 23043 posts. The posts were preprocessed and stemmed appropriately. We choose these two datasets for the diversity and challenges they bring along with them. While Twitter imposes a restriction on the length of posts, the number of followers of a user can run into millions. In a social network, such nodes (users) are sometimes called 'star nodes'. This is a case when the graph is dense but the associated content is much smaller. On the other hand, while the Enron dataset has fewer nodes, emails can be arbitrarily long; case of a sparse graph but rich content. Scaling a technique that integrates both content and link to such diverse social networks is an important challenge.

6.2 Experimental Design

As we discussed before, our definition of communities lays emphasis on two criteria: how tightly users in a community are inter-connected and how strongly users in a community share interests. Hence, we firstly give a qualitative evaluation of the communities and try to argue how topics and links combine to produce communities effectively. We evaluate the strength of inter-connections in the community structure by computing various compactness metrics like normalized-cut[20] and fuzzy modularity [10] of the community structure discovered. Normalized-cut measures the degree of disassociation between groups by computing the cut cost as a fraction of the total edge connections to all the nodes in the graph. On the other hand, modularity [15] was specifically designed as a measure of compactness for community structure. It assumes that a good division of the network is one in which the number of edges between groups is smaller than expected. Since we perform a fuzzy partition of the network, we will instead use a fuzzy variant called Fuzzy Modularity Q_f introduced in [10]. Along with these two measures, we will also use other community quality evaluation measures from the graph clustering literature such as Average link $J_{al}(C)$ and Rouben's measure $J_{rm}(C)$[23]. Next, we pick the task of link prediction and show improvements over baseline models. Finally, we demonstrate speed-up of the parallel variant of SN-LDA over its sequential counterpart in terms of various compactness measures, perplexity and link-prediction accuracy.

A key step in such parametric approaches is choosing the hyper-parameter values. A common heuristic is to pick the values that minimize held-out perplexity on a validation set. A quick investigation on the two datasets showed that setting the number of topics to 15 and number of communities to something between 10 to 12 roughly leads to optimality. Hence, we use the setting Z=15, C=12 in our experiments. In line with other topic modeling works, we set $\alpha = \frac{1}{Z}, \beta = \frac{50}{V}, \gamma = \frac{1}{C}$ *and* $\delta = \frac{10}{U}$ in our SN-LDA implementation. For Parallel SN-LDA, guided by the set-up in previous works[14] we set $a = \frac{U \times (P-1)}{PC}, b = 1, c = 2, d = 0.1, e = \frac{W \times (P-1)}{PZ}, f = 1, g = \frac{W \times (P-1)}{PC}, h = 1, \alpha = \frac{2}{C}, \beta = \frac{2}{Z}, \gamma = 0.1$ *and* $\delta = \frac{2}{C}$. Gibbs sampling was carried out by starting with a random assignment to all the latent variables and using the corresponding update equations to compute fresh values over a large burn-in period (first 1000 iterations). When this distribution stabilizes, sufficient number of samples were taken at regular intervals (every 5th iteration for next 4000 iterations) to avoid correlation.

6.3 Results

6.3.1 Qualitative Analysis

From our first analysis, we intend to qualitatively corroborate our intuition that communities are formed when users with similar interests aggregate together. We first give a view of top words for a few sample topics discovered by SN-LDA on the Enron corpus in Table 1. As many as 10 out of the 15 topics can be recognized as popular and sensible topics of discussion in Enron like power, gas, etc. Similar visualization is possible for the twitter datasets where we obtain topics like Internet, Stock Markets, Web, News, etc.

Next, we illustrate intuitively the utility of such a probabilistic model. For instance, we can compute the distribution over topics for a particular user showing the things that she is primarily interested in (See Figure 3a). [12] showed how one can discover social roles of people by associating words with users through their community memberships and the corresponding community-topic distribution. In Table 2 we give top words for a few roles using the Enron corpus. From these words we can see that social roles are nothing but work profiles for people working in Enron, like management, en-

[2]http://www.cs.cmu.edu/ enron

[3]http://snap.stanford.edu/data/twitter7.html

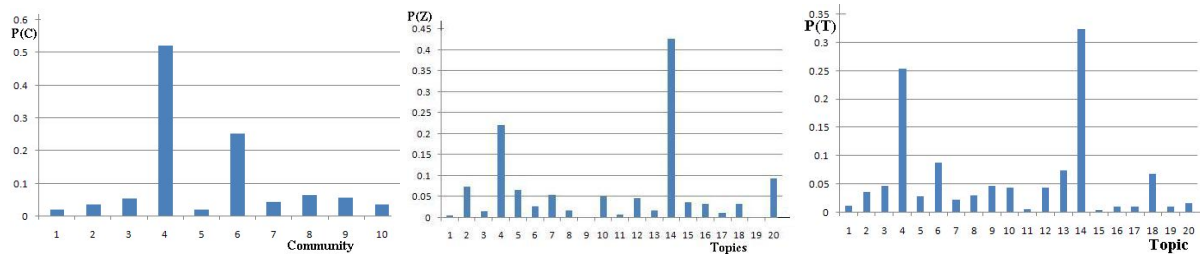

Figure 3: (a) Community Proportions for a given user, (b) Topic Proportions for a given Community and (c) Topic Proportions for a given user for Twitter dataset

Email	**Web**	**Stock Market**	**Gas Prices**	**California Power**
message	mailto	pdx	enron	edison
subject	aol	nymex	gas	power
fw	hotmail	enronxgate	prices	energy
original	net	company	book	ect
mail	http	corp	dynergy	california
Positions	**Office**	**Power Crisis**	**Gas Trading**	**Gas Production**
analyst	backup	gas	city	enron
associate	plan	california	hou	energy
capacity	seat	power	gas	paso
california	business	ect	trading	ngcorp
company	work	energy	california	el

Table 1: (Manually labeled) Topics for SN-LDA on Enron

Management	**Engineering**	**Analyst**
contract	mailto	pdx
agreement	dynergy	california
meeting	electric	cash
corporation	psa	database
budget	calpx	meeting

Table 2: Social Roles discovered by SN-LDA on Enron

gineering and analyst. These social roles can be confirmed against their true roles in Enron. For example, the roles of Sally Beck (Chief Operating Officer), Vincent Kaminski (Head of Quantitative Modeling Group) and Louise Kitchen (President of Enron Online) can be conveniently recognized as management and the role of William Williams III (Senior Analyst) can be recognized as an analyst. Interestingly, the social roles/departmental structures also have high degree of overlap with the community structure discovered.

In Figure 3a, we illustrate for the Twitter dataset that a particular user (user 93) has a membership in community 4 to a high degree in our SN-LDA model. Besides, the model also suggests that the user participates in community 6 to some extent. This analysis gives us an insight to the extent of participation of a user in various social groups. This probabilistic notion of membership has clear advantages in modeling user tastes and preferences to the hard clustering approach taken in many previous approaches for community discovery. Further, topical peaks for a community indicate the dominant topics for that particular community. For example, after looking at the topic proportions for community 4 (see Figure 3b), which is the primary community for user 93, it was found that topics 14 (Stock Markets) is the dominant topic in this community. Also topic 4 (Internet) is the dominant topic in community 6. Now that we know how to estimate community memberships and topical interest of

	Enron Corpus			Twitter Corpus		
	$N-Cut$	J_{al}	J_{rm}	$N-Cut$	J_{al}	J_{rm}
Net-LDA	6.280	1.230	0.5753	7.432	1.189	0.698
SN-LDA	**6.083**	**1.616**	**1.134**	**6.953**	**1.457**	**0.928**
TUCM	10.563	0.672	0.413	9.112	0.657	0.442
TURCM-1	9.584	0.695	0.441	9.938	0.623	0.425
TURCM-2	8.235	0.701	0.452	8.887	0.729	0.461
Full TURCM	8.137	0.921	0.563	9.115	0.696	0.451
CART	11.375	0.694	0.481	10.976	0.605	0.410
CUT	10.122	0.705	0.456	10.785	0.627	0.422
MMSB	7.980	1.266	0.646	8.463	1.198	0.606
Naiive SN-LDA	13.184	0.296	0.280	10.847	0.418	0.416

Table 3: Network Quality Measures on Enron and Twitter

communities, we can also determine the topical interests of the users. Figure 3c shows the distribution over topics for the Twitter user (user 93). It shows that this user is primarily interested in topic 14 (Stock Markets) and also likes topics 4 (Internet). Recall that topics 14 and 4 were strong interests of the communities (community 4 and 6) the user had a high membership in). This analysis is useful in finding individual user's interests and tastes. This kind of analysis supports our hypothesis that users tend to communicate frequently over certain topics (based on interests) and form communities based on those interests. This also illustrates the power of SN-LDA over the plain Net-LDA model.

6.3.2 *Community Quality Analysis*

Next, we present our first quantitative results for evaluating the goodness of communities obtained on the compactness metrics previously defined and compare Net-LDA and SN-LDA against state-of-the-art methods. Tables 3 and 4 show the comparison on both datasets. To establish consistency of the method for various parameter values, table 4 shows the fuzzy modularity comparison for different numbers of communities (Z fixed to 15). Both NetLDA and SN-LDA show much better scores than the baselines, suggesting that our models have detected more spatially compact communities. Also, as expected SN-LDA shows an improvement over Net-LDA due to its additional power of accounting for user interests while discovering communities. It is interesting to note that Net-LDA shows an improvement over MMSB, another model for community detection using link structure in graphs, which suggests that our modeling scheme for networks models compactness in a better manner for integer-valued graphs. Naiive SN-LDA does not perform well. This substantiates our modeling choice for drawing separate community labels for content and link modeling.

To reiterate our argument, we visualize the community assignments for CART and our two methods on the Enron dataset in figure 4. We assign each user to his most probable community and rearrange so that all the users assigned to

	Enron Corpus					Twitter Corpus				
	6	**8**	**10**	**12**	**14**	**6**	**8**	**10**	**12**	**14**
Net-LDA	0.619	**0.542**	**0.463**	0.340	0.421	0.413	0.442	0.462	0.471	0.463
SN-LDA	**0.634**	0.534	0.427	**0.427**	**0.436**	**0.438**	**0.491**	**0.482**	**0.484**	**0.485**
TUCM	0.148	0.243	0.291	0.287	0.246	0.167	0.263	0.321	0.313	0.262
TURCM-1	0.198	0.271	0.339	0.331	0.283	0.168	0.261	0.309	0.287	0.241
TURCM-2	0.203	0.278	0.346	0.337	0.289	0.166	0.265	0.324	0.309	0.261
Full TURCM	0.215	0.294	0.363	0.350	0.299	0.171	0.272	0.332	0.316	0.267
CART	0.152	0.249	0.302	0.294	0.255	0.157	0.227	0.243	0.235	0.196
CUT	0.133	0.231	0.266	0.278	0.227	0.159	0.244	0.299	0.285	0.237
MMSB	0.2601	0.3313	0.3745	0.375	0.362	0.226	0.274	0.289	0.291	0.284
Naiive SN-LDA	0.114	0.119	0.125	0.127	0.128	0.158	0.175	0.187	0.192	0.200

Table 4: Fuzzy Modularity Comparisons on Enron and Twitter Corpora as C is varied, Z=15

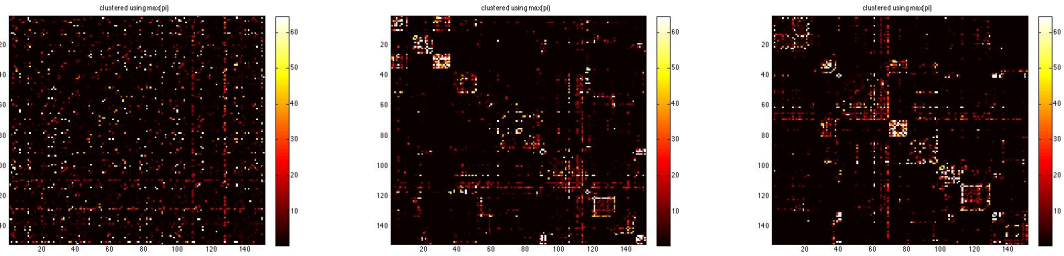

Figure 4: Visualizing the community assignments for (a) CART, (b) Net-LDA (c)SN-LDA on Enron Dataset

a particular community are stacked together. After this operation, the adjacency (strength) matrix is plotted for each model. We can observe that Net-LDA and SN-LDA roughly show a block diagonal structure reaffirming our argument that these methods detect compacter communities. Previous models (like CART), on the other hand, do not show a characteristic compactness structure.

6.3.3 *Link prediction*

Next, we illustrate performance on a supervised link prediction task. Link prediction is closely related to communities. In fact, community information is often included in similarity-based link prediction methods[21] with an implicit assumption that two nodes in the same community are more likely to be connected. However, we believe that this assumption would fail if the communities were not spatially compact. Hence, we provide performance on the link prediction task as another evaluation of our models. For this, we made a 60-40 split of all documents. We learnt our model parameters from the 60% training split and trained SVM to predict links (if there is a document exchange in Enron) between user pairs on the test-split, using user community-membership profiles as the feature vectors. Table 5 shows that both NetLDA and SN-LDA outperform the baselines on both datasets further confirming our hypothesis.

6.3.4 *Parallel SN-LDA*

As our final evaluation, we illustrate the speed-up of Parallel SN-LDA over its sequential counterpart. Herein, we are interested in two aspects of performance: the quality of the models learned measured by the compactness metrics, performance on the link prediction task and perplexity of held out data, and the time taken to learn the models.

To begin with, we compute held out perplexity. Perplexity is a measure used to evaluate language models. We compute perplexity using the harmonic mean method proposed in

	Enron Corpus			Twitter Corpus		
	Prec.	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**
Net-LDA	0.708	0.467	0.563	0.648	0.547	0.593
SN-LDA	0.656	0.527	**0.584**	0.677	0.559	**0.612**
TUCM	0.583	0.403	0.476	0.542	0.428	0.478
TURCM-1	0.528	0.329	0.405	0.526	0.401	0.455
TURCM-2	0.512	0.367	0.427	0.582	0.449	0.507
Full TURCM	0.504	0.412	0.453	0.558	0.450	0.498
CART	0.565	0.332	0.418	0.538	0.388	0.451
CUT	0.513	0.304	0.382	0.515	0.353	0.419
MMSB	0.590	0.482	0.531	0.547	0.474	0.508
Naiive SN-LDA	0.512	0.286	0.367	0.505	0.326	0.396

Table 5: Link Prediction Results on Enron and Twitter

[22]. However, unlike topic models where data is just the set of words, here we want to evaluate how well our models do on learning the linkage structure as well as the text. Hence, we further compute the perplexity of observing links and text separately. The link and text perplexity can be computed using S samples from independent chains of the posterior:

$$p(\mathbf{W_{test}}|.) = \sum_{u\in\mathbf{U}}\sum_{d\in\mathbf{D_u}} log\frac{1}{S}\left(\sum_s\sum_c \pi_{c|u}\prod_{w\in d}\sum_z \theta_{z|c}\phi_{w|z}\right)$$

$$p(\mathbf{R_{test}}|.) = \sum_{u\in\mathbf{U}}\sum_{r\in\mathbf{N_u}}\sum_{d\in\mathbf{D_{ur}}}\sum_{v\in\mathbf{V}} log\frac{1}{S}\left(\sum_s\sum_c \pi_{c|u}\eta_{r|c}\right)$$

$$\pi_{c|u} = \frac{\#_{uc}^{UC^1} + \#_{uc}^{UC^2} + \gamma}{\sum_c\left(\#_{uc}^{UC^1} + \#_{uc}^{UC^2} + \gamma\right)}, \theta_{z|c} = \frac{\#_{cz}^{CZ} + \alpha}{\sum_z(\#_{cz}^{CZ} + \alpha)}$$

$$\phi_{w|z} = \frac{\#_{zw}^{ZW} + \beta}{\sum_w(\#_{zw}^{ZW} + \beta)}, \eta_{r|c} = \frac{\#_{cr}^{CR} + \delta}{\sum_r(\#_{cr}^{CR} + \delta)}$$

$$Ppx(\mathbf{W}) = exp\left(\frac{-log\ p(\mathbf{W_{test}}|.)}{W_{test}}\right), Ppx(\mathbf{R}) = exp\left(\frac{-log\ p(\mathbf{R_{test}}|.)}{R_{test}}\right)$$

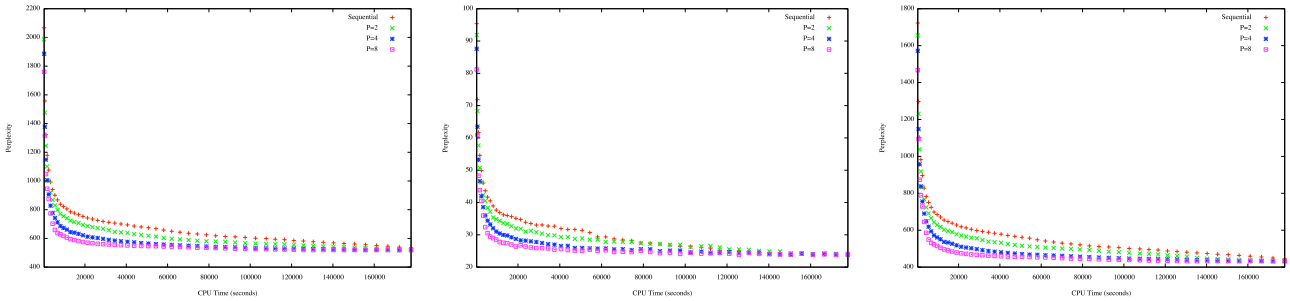

Figure 5: Perplexity Comparisons: Parallel SN-LDA and SN-LDA (a) Text, (b) Link, (c) Overall Perplexity on Enron

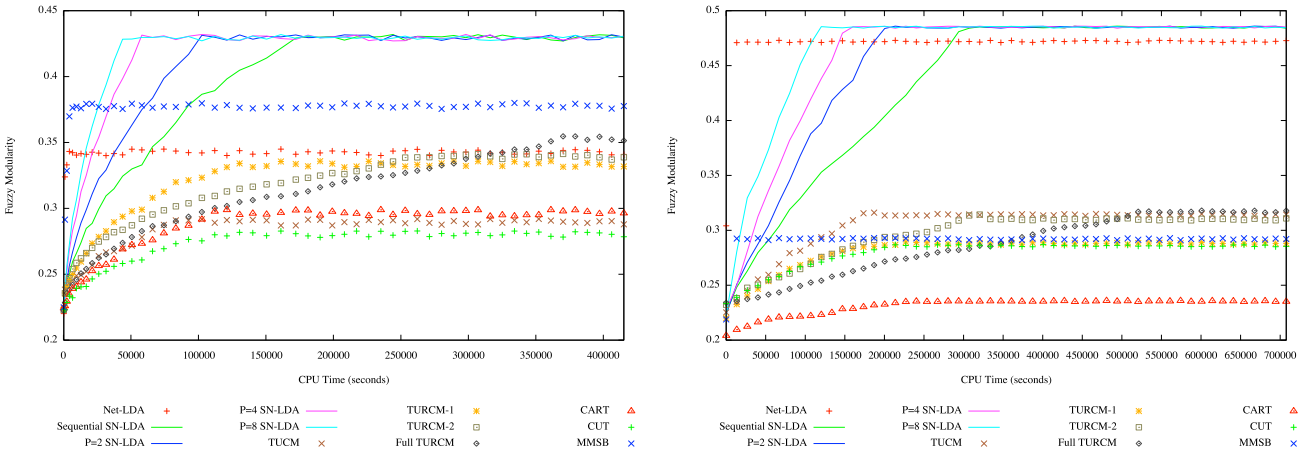

Figure 6: Speedup & Comparison of Fuzzy Modularity on (a) Enron, (b) Twitter

Figure 5 plots held out perplexity against runtime for parallel SN-LDA model (when number of processors is 2,4 and 8) and SN-LDA for the Enron dataset. Herein, it is easy to observe that Parallel SN-LDA shows a greater drop in both held-out perplexities as well as overall perplexities with time implying faster convergence. Both text and link perplexities fall monotonically. This implies that the model's predictive ability on both text and link improves with time. The drop becomes more and more steep as the number of processors increase. Please note that in the limit, the perplexity of the parallel algorithms matches the sequential algorithm. This pattern is independent of choice of Z and C. Similar pattern is seen on the Twitter dataset. In fact, the parallel model also improves with time on various compactness metrics at a greater rate than the sequential version as the number of processors is increased. Figure 6 shows the improvement on both datasets. A similar pattern was also seen in terms of link prediction in Figure 7. Also note that in our experiments, Parallel SN-LDA with just 4 processors converges faster that all the baselines with the exception of MMSB.

	100	**500**	**2500**	**12500**
P=2	0.613	0.593	0.599	0.592
P=4	0.607	0.594	0.593	0.568
P=8	0.602	0.593	0.575	0.564

Table 6: Efficiency Speed-ups by Parallel SN-LDA on Twitter datasets of various sizes

Finally, we compute the speed-up efficiency: fractional decease in training (cpu) time over the number of processors used ($\frac{T_1}{pT_p}$). All simulations have been done on the same architecture and platform. In all models, training is assumed to finish when the held-out likelihoods converge (change in likelihood over consecutive iterations falls below 0.1% of the likelihood in the previous iteration). We chose twitter datasets of various sizes (100, 500, 2500 and 12500 nodes, respectively) and compute the speed-ups with respect to the size of datasets for various number of processors used in Table 6. The consistency of speed-ups across data sizes makes us believe that it should be possible to scale up our technique to datasets of larger size with more processors.

7. CONCLUSION

We began by positing that communities are formed by users who communicate on topics of mutual interest, are connected to each other in the social graph and share frequent personal communication. We described a new representation for social networks and used it to propose a mixed-membership model which incorporates both text and link information for community detection in social networks that conforms to the notion of 'compactness'. When compared against existing methods, our experiments show significant improvements in terms of various compactness metrics and on the link prediction task. We also provide a fast, efficient parallel approximation of the model that scales well in terms of both memory and time. We believe that the scheme of modeling network information as described in Net-LDA can

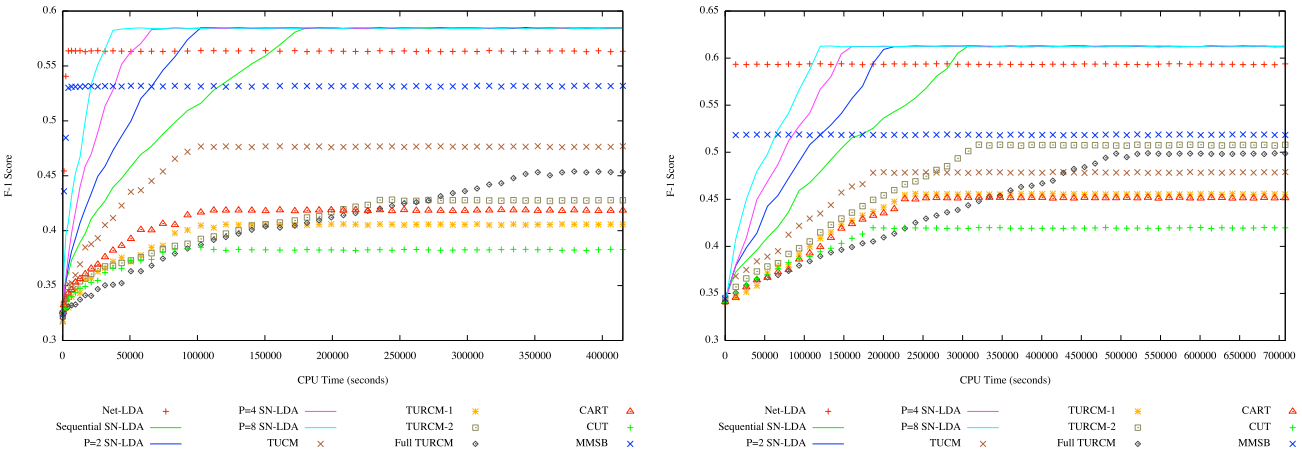

Figure 7: Speedup & Comparison of Link Prediction on (a) Enron, (b) Twitter

be employed to improve various latent variable models previously described in the literature.

APPENDIX

Gibbs Sampling for Parallel SN-LDA:

For parallel SN-LDA the posterior of Z_p, C_p^1, C_p^2 given the global variables $\gamma_p, \theta_c^1, \theta_c^2, \eta_c^1, \eta_c^2, \phi_k^1, \phi_k^2$ has the same form as the posterior of Z, C^1, C^2 given $\gamma, \beta, \alpha, \delta$ for SN-LDA.

$$P(\mathbf{Z}_{\ldots\mathbf{p}}, \mathbf{C^1}_{\ldots\mathbf{p}}, \mathbf{C^2}_{\ldots\mathbf{p}} | \gamma_p, \theta_c^1, \theta_c^2, \eta_c^1, \eta_c^2, \phi_k^1, \phi_k^2) \propto$$
$$\prod_{u \in \mathbf{U_p}} \left[\frac{\Gamma(C\gamma_p)}{\Gamma(C\gamma_p + \#_{u.p}^{UC^1P} + \#_{u.p}^{UC^2P})} \prod_c \frac{\Gamma(\gamma_p + \#_{ucp}^{UC^1P} + \#_{ucp}^{UC^2P})}{\Gamma(\gamma_p)} \right]$$
$$\prod_c \left[\frac{\Gamma(\theta_c^1)}{\Gamma(\#_{c.p}^{C^1ZP} + \theta_c^1)} \prod_z \frac{\Gamma(\#_{czp}^{C^1ZP} + \theta_c^1\theta_{cz}^2)}{\Gamma(\theta_c^1\theta_{cz}^2)} \right]$$
$$\prod_c \left[\frac{\Gamma(\eta_c^1)}{\Gamma(\#_{c.p}^{C^2RP} + \eta_c^1)} \prod_r \frac{\Gamma(\#_{crp}^{C^2RP} + \eta_c^1\eta_{cr}^2)}{\Gamma(\eta_c^1\eta_{cr}^2)} \right]$$
$$\prod_z \left[\frac{\Gamma(\phi_z^1)}{\Gamma(\#_{z.p}^{ZWP} + \phi_z^1)} \prod_w \frac{\Gamma(\#_{zwp}^{ZWP} + \phi_z^1\phi_{zw}^2)}{\Gamma(\phi_z^1\phi_{zw}^2)} \right]$$

Consequently, the sampling equation for the local variables are similar to that of SN-LDA given in equation 1-3 with γ replaced by γ_p, α replaced by $\theta_c^1\theta_c^2$ and β replaced by $\phi_k^1\phi_k^2$ and the counts replaced by counts for the given processor.

The auxiliary variable method used to sample the global variables $\gamma_p, \theta_c^1, \theta_c^2, \eta_c^1, \eta_c^2, \phi_k^1$ *and* ϕ_k^2 here is explained in detail in [5] and [14]. We use the following expression:

$$\frac{\Gamma(a)}{\Gamma(a+n)} = \frac{1}{\Gamma(n)} \int_0^1 t^{a-1}(1-t)^{n-1} dt \quad (4)$$
$$\frac{\Gamma(a+n)}{\Gamma(a)} = \sum_{s=0}^{n} S(n,s)(a)^s \quad (5)$$

where, S is the sterling number of the first kind.

Sampling γ_p:

$$P(\gamma_p|-) \propto \prod_{u \in \mathbf{U_p}} \left[\frac{\Gamma(C\gamma_p)}{\Gamma(C\gamma_p + \#_{u.p}^{UC^1P} + \#_{u.p}^{UC^2P})} \prod_c \frac{\Gamma(\gamma_p + \#_{ucp}^{UC^1P} + \#_{ucp}^{UC^2P})}{\Gamma(\gamma_p)} \right] \gamma_p^{c-1} \exp(-d\gamma_p)$$

Using equations 4 and 5 :

$$P(\gamma_p, t, s|-) \propto \prod_{u \in \mathbf{U_p}} \left[t_u^{C\gamma_p - 1}(1-t_u)^{\#_{u.p}^{UC^1P} + \#_{u.p}^{UC^2P}} \right] \times \prod_{u \in \mathbf{U_p}} \prod_c \left[S(\#_{ucp}^{UC^1P} + \#_{ucp}^{UC^2P}, s_{ucp}) \gamma_p^{s_{ucp}} \right] \gamma_p^{c-1} \exp(-d\gamma_p)$$

This leads to simultaneous sampling equations for γ_p, t & s:

$$P(\gamma_p|t,s,-) \propto \prod_{u \in \mathbf{U_p}} t_u^{C\gamma_p} \left(\prod_{u \in \mathbf{U_p}} \prod_c \gamma_p^{s_{ucp}} \right) \gamma_p^{c-1} \exp(-d\gamma_p)$$
$$= Gamma\left(c + \sum_{u \in \mathbf{U_p}} \sum_c S_{ucp}; d - C \sum_{u \in \mathbf{U_p}} t_u \right) (*)$$

$$P(t_u|\gamma_p, s-) \propto t_u^{C\gamma_p - 1}(1-t_u)^{\#_{u.p}^{UC^1P} + \#_{u.p}^{UC^2P}}$$
$$= Beta\left(C\gamma_p; \#_{u.p}^{UC^1P} + \#_{u.p}^{UC^2P} \right) \quad (*)$$

$$P(s_{ucp}|\gamma_p, t, -) \propto S(\#_{ucp}^{UC^1P} + \#_{ucp}^{UC^2P}, s_{ucp}) \gamma_p^{s_{ucp}}$$
$$= Antoniak\left(\#_{ucp}^{UC^1P} + \#_{ucp}^{UC^2P}, \gamma_p \right) \quad (*)$$

Note that both auxiliary variables can be sampled locally.

Sampling η_c^1 *and* η_c^2:

$$P(\eta_c^1, \eta_c^2|-) \propto \prod_c \left[\frac{\Gamma(\eta_c^1)}{\Gamma(\#_{c.p}^{C^2RP} + \eta_c^1)} \prod_r \frac{\Gamma(\#_{crp}^{C^2RP} + \eta_c^1\eta_{cr}^2)}{\Gamma(\eta_c^1\eta_{cr}^2)} \right] \left[\prod_r {\eta_{cr}^2}^{\delta-1} \right] {\eta_c^1}^{a-1} \exp(-b\eta_c^1)$$

Using the same argument but different auxiliary variables:

$$P(\eta_c^1, \eta_c^2, t, s|-) \propto {\eta_c^1}^{a-1} \left[\prod_p t_{cp}^{\eta_c^1 - 1}(1-t_{cp})^{\#_{c.p}^{C^2RP} - 1} \right] \times \left[\prod_{p,r} S(\#_{crp}^{C^2RP}, s_{crp}) (\eta_c^1\eta_{cr}^2)^{s_{rcp}} \right] \left[\prod_{c,r} {\eta_{cr}^2}^{\delta-1} \right] \exp(-b\eta_c^1) \quad (*)$$

$$P(\eta_c^1|-) \propto Gamma\left[a + \sum_{p,r} s_{crp}; b - \sum_{p} \log(t_{cp})\right] \quad (*)$$

$$P(\eta_c^2|-) \propto Dirichlet\left[\delta + \sum_{p} s_{crp}\right] \quad (*)$$

$$P(t_{cp}|-) \propto Beta\left[\eta_c^1; \#_{c.p}^{C^2RP}\right] \quad (*)$$

$$P(s_{crp}|-) \propto Antoniak\left[\#_{crp}^{C^2RP}; \eta_c^1\eta_{cr}^2\right] \quad (*)$$

Sampling for θ_c^1, θ_c^2, ϕ_k^1 and ϕ_k^2 is similar to η_c^1 and η_c^2.

Acknowledgments

Thanks to Qirong Ho, Kriti Puniyani, Keerthiram Murugesan and the anonymous reviewers for their valuable comments and suggestions to improve the quality of the paper.

A. REFERENCES

[1] Edoardo M. Airoldi, David M. Blei, Stephen E. Fienberg, and Eric P. Xing. Mixed membership stochastic blockmodels. *J. Mach. Learn. Res.*, 9:1981–2014, June 2008.

[2] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, March 2003.

[3] Jonathan Chang. Relational topic models for document networks. In *Proc. of Conf. on AI and Statistics (AISTATS)*, 2009.

[4] Aaron Clauset, M. E. J. Newman, and Cristopher Moore. Finding community structure in very large networks. *Phys. Rev. E*, 70(6):066111, 2004.

[5] Michael D. Escobar and Mike West. Bayesian density estimation and inference using mixtures. *Journal of the American Statistical Association*, 90:577–588, 1994.

[6] Gary William Flake, Steve Lawrence, C. Lee Giles, and Frans M. Coetzee. Self-organization and identification of web communities. *IEEE Computer*, 35:66–71, 2002.

[7] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.

[8] Keith Henderson, Brian Gallagher, Lei Li, Leman Akoglu, Tina Eliassi-Rad, Hanghang Tong, and Christos Faloutsos. It's who you know: graph mining using recursive structural features. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '11, pages 663–671, New York, NY, USA, 2011. ACM.

[9] Qirong Ho, Junming Yin, and Eric P. Xing. On triangular versus edge representations — towards scalable modeling of networks. In Peter L. Bartlett, Fernando C. N. Pereira, Christopher J. C. Burges, LÃl'on Bottou, and Kilian Q. Weinberger, editors, *NIPS*, pages 2141–2149, 2012.

[10] Jian Liu. Fuzzy modularity and fuzzy community structure in networks. *The European Physical Journal B*, 77:547–557, 2010.

[11] Yan Liu, Alexandru Niculescu-Mizil, and Wojciech Gryc. Topic-link lda: joint models of topic and author community. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 665–672, New York, NY, USA, 2009. ACM.

[12] Andrew McCallum, Andrés Corrada-Emmanuel, and Xuerui Wang. Topic and role discovery in social networks. In *Proceedings of the 19th international joint conference on Artificial intelligence*, IJCAI'05, pages 786–791, San Francisco, CA, USA, 2005. Morgan Kaufmann Publishers Inc.

[13] Ramesh M. Nallapati, Amr Ahmed, Eric P. Xing, and William W. Cohen. Joint latent topic models for text and citations. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '08, pages 542–550, New York, NY, USA, 2008. ACM.

[14] David Newman, Arthur Asuncion, Padhraic Smyth, and Max Welling. Distributed algorithms for topic models. *J. Mach. Learn. Res.*, 10:1801–1828, December 2009.

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

[16] Gergely Palla, Imre Derényi, Illés Farkas, and Tamás Vicsek. Uncovering the overlapping community structure of complex networks in nature and society, 2005.

[17] Nishith Pathak, Colin DeLong, Arindam Banerjee, and Kendrick Erickson. Social topics models for community extraction. In *Proceedings of the 2nd SNA-KDD Workshop*, 2008.

[18] Michal Rosen-Zvi, Thomas Griffiths, Mark Steyvers, and Padhraic Smyth. The author-topic model for authors and documents. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, UAI '04, pages 487–494, Arlington, Virginia, United States, 2004. AUAI Press.

[19] Mrinmaya Sachan, Danish Contractor, Tanveer A. Faruquie, and L. Venkata Subramaniam. Using content and interactions for discovering communities in social networks. In *Proceedings of the 21st international conference on World Wide Web*, WWW '12, pages 331–340. ACM, 2012.

[20] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):888–905, August 2000.

[21] Sucheta Soundarajan and John Hopcroft. Using community information to improve the precision of link prediction methods. In *Proceedings of the 21st international conference companion on World Wide Web*, WWW '12 Companion, pages 607–608, New York, NY, USA, 2012. ACM.

[22] Hanna M. Wallach, Iain Murray, Ruslan Salakhutdinov, and David Mimno. Evaluation methods for topic models. In *ICML*, 2009.

[23] M.-S. Yang. A survey of fuzzy clustering. *Mathematical and Computer Modelling*, 18(11):1 – 16, 1993.

[24] Ding Zhou, Eren Manavoglu, Jia Li, C. Lee Giles, and Hongyuan Zha. Probabilistic models for discovering e-communities. In *Proceedings of the International Conference on World Wide Web*, 2006.