Scalable Modeling of Conversational-role based Self-presentation Characteristics in Large Online Forums

Abhimanu Kumar
School of Computer Science
Carnegie Mellon University
abhimank@cs.cmu.edu

Shriphani Palakodety
School of Computer Science
Carnegie Mellon University
shriphanip@gmail.com

Chong Wang
School of Computer Science
Carnegie Mellon University
chongw@cs.cmu.edu

Carolyn P. Rose
School of Computer Science
Carnegie Mellon University
cprose@cs.cmu.edu

Eric P. Xing
School of Computer Science
Carnegie Mellon University
epxing@cs.cmu.edu

Miaomiao Wen
School of Computer Science
Carnegie Mellon University
mwen@cs.cmu.edu

ABSTRACT

Online discussion forums are complex webs of overlapping sub-communities (macrolevel structure, across threads) in which users enact different roles depending on which subcommunity they are participating in within a particular time point (microlevel structure, within threads). This sub-network structure is implicit in massive collections of threads. To uncover this structure, we develop a scalable algorithm based on stochastic variational inference and leverage topic models (LDA) along with mixed membership stochastic block (MMSB) models. We evaluate our model on three large-scale datasets, Cancer-ThreadStarter (22K users and 14.4K threads), Cancer-NameMention(15.1K users and 12.4K threads) and StackOverFlow (1.19 million users and 4.55 million threads). Qualitatively, we demonstrate that our model can provide useful explanations of microlevel and macrolevel user presentation characteristics in different communities using the topics discovered from posts. Quantitatively, we show that our model does better than MMSB and LDA in predicting user reply structure within threads. In addition, we demonstrate via synthetic data experiments that the proposed active sub-network discovery model is stable and recovers the original parameters of the experimental setup with high probability.

1. INTRODUCTION

Online forums are a microcosm of communities where users' presentation characteristics vary across different regions of the forum. Users participate in a discussion or group activity by posting on a related thread. During his stay on a forum, a user participates in many different discussions and posts on multiple threads. The thread level presentation characteristics of a user are different than the global presentation characteristics. A participating user gears his responses to suit specific discussions on different threads. These thread based interactions give rise to active sub-networks, within the global network of users, that characterize the dynamics of

interaction. Overlaying differential changes in user interaction characteristics across these sub-networks provides insights into users' macroscopic (forum-wide) as well as microscopic (thread specific) participation behavior.

Analysing online social networks and user forums have been approached using various perspectives such as network [15, 14] , probabilistic graphical models [1], combined network & text [7, 12]. However none of these have taken into account the dynamics of sub-networks and the related thread-based framework within which forum discussions take place. Whereas active sub-network modelling has been very useful to the research in computational biology in recent years where it's been used to model sub-networks of gene interactions [3, 11], very few approaches using active sub-network have been proposed to model online user interactions. Taking into account sub-network interaction dynamics is important to correctly model the user participant behavior. For example, users post their responses on discussion threads after reading through responses of other users in the threads. The users possibly post multiple times on the thread as a form of reply to other posts in the thread. For analysing such interactions it becomes imperative that the structure of the conversation must also be taken into account besides the user interaction network and the text posted. This enables us to gain deeper insights into user behavior in the online community that was not possible earlier.

One of the main challenges of this work has been the ability to model active sub-networks in a large forum with millions of users and threads. A social network spanning around millions of users and threads would be an ideal case to demonstrate the effectiveness of sub-network modelling. To efficiently scale our model, we derive a scalable inference based on stochastic variational inference (SVI) with sub-sampling [9] that has the capacity to deal with such massive scale data and parameter space. The scalability of the SVI with sub-sampling is further boosted by employing Poisson distribution to model edge weights of the network. A Poisson based scheme need not model zero edges ([10]), where as MMSB style approaches [1] must explicitly model them. A further set of parallelization in inner optimization loops of local variational parameters pushes the learning speed even more. This work is to date the largest modelling of any social graph that also takes user contents into account.

Contributions.

- This work provides novel insights into how users' self- representational characteristics vary depending on the discussion they are in. This is achieved via active sub-network modelling.

- Our model outperforms LDA and MMSB in link prediction across three datasets demonstrating the leverage it gains by combining the two along with discussion structures in modelling sub-networks.

- It is highly scalable and is able to achieve convergence in matter of hours for users and threads that are an order of a million despite the time complexity of $O(\text{users} \times \text{users} \times \text{threads})$.

- Stability is another aspect of the proposed new approach and is demonstrated by the model recovering back its parameters in synthetic experiments.

2. USER ROLE MODELLING

Online forums have a specific discussion structure that provides a lot of context to all the interactions occurring among the users. Here we describe a typical forum discussion scenario.

2.1 Discussion structure in online forums

When two users interact in a thread or through a post they play certain conversational roles and project their specific identity. It is valuable to know what conversational roles each plays (which topic or community they each belong to) in that interaction. When a user u is representing community c out of all the communities that he is part of and participates in a discussion, he tailors his post content accordingly to suit the explicit or implicit community and discussion norms. Knowing the style of community specific text content provides a lot of information about that community in general. It also provides information about what role user u plays when he is in community c and engaged in a specific discussion thread t. In online forums multi-user interactions occur a lot i.e. in a thread a user can post by addressing another specific user but he is also addressing other users in the thread explicitly or implicitly (via either gearing his replies to address other users' concerns into consideration or addressing them directly in the post). Modeling this phenomenon would bring the model closer to realities of online discussions. This can be modelled by aggregating users posts across a thread, though not across the whole of the forum. We will elaborate on this more in the generative story of our model.

2.2 Graphical model & generative story

Based on the description above our graphical model is designed as shown in Figure 1. In this model we aggregate the posts of a given user in a given thread t into one document which has token set $N_{t,p}$. This helps us incorporate the knowledge that a user's post is influenced by the posts of other users present on the thread, assuming that he reads at least some of them.

The generative process for the model is as follows:

- For each user p,
    - Draw a K dimensional mixed membership vector $\vec{\pi}_p \sim$ Dirichlet(α).
- for each topic pair g and h,
    - Draw $B(g, h) \sim Gamma(\kappa, \eta)$; where κ, η are parameters of the gamma distribution.
- For each pair of users (p, q) and each thread t,
    - Draw membership indicator for the initiator, $\vec{z}_{(p \rightarrow q, t)} \sim$ Multinomial(π_p).
    - Draw membership indicator for the receiver, $\vec{z}_{(p \leftarrow q, t)} \sim$ Multinomial(π_q).
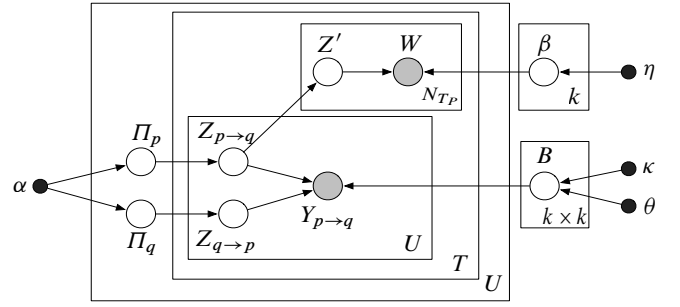

Figure 1: The proposed approach models active sub-network of users in the forum. U si the total number of user in all of the forum, T si the number of threads. N_{T_p} is the total number of tokens user p posted in thread t.

- Sample the value of their interaction, $Y(p, q, t) \sim$ Poisson$(\vec{z}^{\top}_{(p \rightarrow q, t)} B \, \vec{z}_{(p \leftarrow q, t)})$.

- For each user $p \in t$,
    - Form the set $\delta_{t,p}$ that contains all the users that p interacts to on thread t,
        * For each word $w \in N_{t,p}$,
        * Draw $z'_{t,p,w}$ from $Dirichlet(\sum_{\forall q \in \delta_{t,p}} z_{(t,p \rightarrow q)})$.
        * Draw $w \sim \phi(w|\beta, z'_{t,p,w})$.

The use of Poisson distribution for $Y(p, q, t) \sim$ Poisson$(\vec{z}^{\top}_{(p \rightarrow q, t)} B \, \vec{z}_{(p \leftarrow q, t)})$ (the network edge between the user's p and q) besides modelling non-binary edge strength enables the model to ignore non-edges between users $(Y_{t,p,q})$ and thus achieve faster convergence [10]. In MMSB style community block models, non-edges are to be modelled explicitly.

$$\log L = \log P(Y, W, Z_{\leftarrow}, Z_{\rightarrow}, \Pi, B, \beta | \alpha, \eta, \theta, \alpha). \quad (1)$$

The log-likelihood of the model described in 2.2 is given above and derived in detail in the appendix 14.

$$q = \prod_p q(\Pi_q | \gamma_p) \prod_t \left[\prod_{p,q} q(Z_{t,p \rightarrow q}, Z_{t,p \leftarrow q} | \phi_{t,p,q}) \right.$$
$$\left. \cdot \prod_{p \in t} \prod_{i=1}^{N_{T_p}} q(Z'_{t,p,i} | \chi_{t,p,i}) \right]$$
$$\cdot \prod_{g,h} q(B_{g,h} | \nu_{g,h} \lambda_{g,h}) \prod_k q(\beta k | \tau_k). \quad (2)$$

We use variational approximation to maximize log-likelihood. Equation 2 above is the approximation of the log-likelihood and we use structured mean field [18] to maximize parameters of q. The local variational parameters, ϕ (MMSB parameters) and χ (LDA) parameters, are maximized using equations 3 and 4 where $\Delta_{\phi'_{t,p,g,h}}$ and $\Delta_{\chi'_{t,p,g,h}}$ are defined by equations 19 and 20 respectively.

$$\phi_{t,p,g,h} \propto e^{\Delta_{\phi'_{t,p,g,h}}}. \quad (3)$$

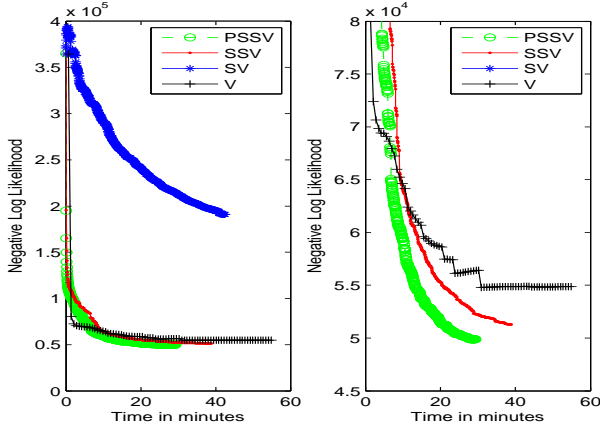$$\chi_{t,p,i,k} \propto e^{\Delta_{\chi'_{t,p,g,h}}}. \quad (4)$$

Figure 2: The log-likelihood vs incremental speed optimization routine. The right hand plot is a zoomed in version of the left. PSSV (Parallel Sub-sampled Stochastic Variational), SSV(Sub-sampled Stochastic Variational), SV(Stochastic Variational) and V(Variational). Each addition of optimization increases the speed by several orders of magnitude. The final PSSV is 4 times faster than (V)ariational and achieves better log-likelihood too.

The traditional variational updates for global parameters γ, ν, λ (MMSB) and τ (LDA) are defined using equations 22, 23, 24 and 25 respectively (details are in the appendix).

Terminology.

There is a difference to be made between community-topic, word topic and user roles. Community topic is the π vector that we get from the model (figure 1) that decides the membership proportion of a user in different latent communities. Word topic is the β vector of word topic proportions from the LDA component of the model, figure 1. There is a one to one correspondence between β and π vectors as seen in figure 1. β helps us in identifying what contents users are generally interested in in a given latent community. A user role is a specific configuration of π. It can be just the case that a role 'r' might be the π vector where r-th coordinate is 1 and all else are 0 out of the total K coordinates, i.e. it predominantly relates to that r-th latent community.

3. SCALABLE ESTIMATION

The global update equations in previous sections are computationally very expensive and slow as we need to sum over all the updated local variables. U users with T threads and vocabulary size V leads to $O(U^2T + UVT)$ local variables. Traditional sampling or variational estimation techniques would be quite slow for such a model. In order to obtain faster convergence we make use of stochastic variational approximation along with sub-sampling and parallelization.

The updates in case of SVI with sub-sampling follow a two step procedure. Step one computes a local update for the global variables based on the sub-sampled updated local variables. The local updates $(\gamma', \nu', \lambda'$ and $\tau')$ for the global variables $(\gamma, \nu, \lambda$ and $\tau)$ are

---

Input : $Y, W, P, \alpha, \theta, \kappa, \eta$
Initialize : $\gamma \leftarrow \gamma_0, \tau \leftarrow \tau_0, \nu \leftarrow \nu_0, \lambda \leftarrow \lambda_0$
while *not converged* **do**
  for c processors **in parallel do**
    pick a set of threads T **for** *each* $t \in T$ **do**
      pick a node $p, \forall q \in neighborhood\ \delta_{t,p}$
      while $\phi\ \&\ \chi$ *not converged* **do**
        get new $\phi_{t,p\rightarrow q}, \phi_{t,p\leftarrow q}, \phi_{t,q\rightarrow p}, \phi_{t,q\leftarrow p}$
        and $\chi_{t,p,i} \forall i \in N_{t,p}$
        iterate between ϕ and χ using equations 3 and 4
      end
    end
  end
  aggregate ϕ and χ obtained from different processors.
  get local update $\gamma', \tau', \nu', \lambda'$ via stochastic approximation of equations 22,25,23,24.
  get global updates of $\gamma, \tau, \nu, \lambda$; e.g.
  $\gamma^{t+1} = (1 - step)\gamma^t + (step)\gamma'$
  Similarly globally update τ, ν, λ as above using equation 9.
end

Algorithm 1: PSSV: Parallel Sub-sampling based Stochastic Variational inference for the proposed model

$$\gamma'_{p,k} = \alpha_k + \frac{NT}{2|S_p|}\sum_{q \in S_p}\sum_h \phi_{t,p,q,k,h} + \frac{NT}{2|S_p|}\sum_{q \in S_p}\sum_g \phi_{t,q,p,g,k}.$$
(5)

$$\nu'_{g,h} = \nu^t_{g,h} + \rho_\nu \frac{NT}{2|S_p|}\sum_{q \in S_p}\frac{dL}{\partial \nu_{g,h}}.$$
(6)

$$\lambda'_{g,h} = \frac{\left(\sum_t\sum_{p,q}\phi_{t,p,q,g,h}y_{t,p,q} + \kappa_{g,h}\right)}{\left(\left(\sum_t\sum_{p,q}\phi_{t,p,q,g,h}\right) + \frac{1}{\theta_{g,h}}\right)\nu_{g,h}}.$$
(7)

$$\tau'_{p,v} = \nu_v + \frac{NT}{2|S_p|}\left(\sum_{w_{t,p,i}=v}^{N_{t,p}}\chi_{t,p,i,k}\right).$$
(8)

where S_p is a set of neighborhood edges of user p, and N and T are total number of edges and threads respectively in the network. The set S_p is chosen amongst the neighbors of p by sampling equal no. zero and non-zero edges.

In step two of the sub-sampled SVI the final update of global variable is computed by the weighted average of the local updates of the global variable and the variables value in the previous iteration:

$$\mu^{t+1} = (1 - \xi^t)\mu^t + \xi^t\mu'.$$
(9)

where μ represents any global variable from $\lambda, \nu, \gamma, \tau$. ξ^t is chosen appropriately using SGD literature and is decreasing. ξ^t is standard stochastic gradient descent rate at iteration t, also expressed as $\xi^t = \frac{1}{(t+\zeta)^\rho}$ [6]. ζ and ρ are set as 1024 and 0.5 respectively for all our experiments in the paper, and t is the iteration number.

We achieve further speed by parallelizing the text (χ) and network (ϕ) local variational updates. This is achievable as the dependency between phi and χ parameters (defined in equations 20 and 19) allows us to parallelize their variational updates. Algorithm 1 describes the parallelized SVI with sub-sampling updates for the local parameters. Figure 2 shows a plot of how the final (p)arallel (s)ub-sampling based (s)tochastic (v)ariational (PSSV) inference is faster

	users	threads	posts	edges
TS	22,095	14,416	1,109,125	287,808
UM	15,111	12,440	381,199	177,336
SO	1,135,996	4,552,367	9,230,127	9,185,650

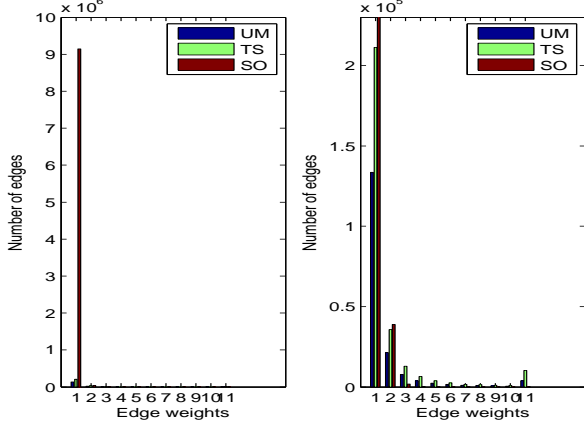Table 1: Dataset statistics. SO mostly has edges with weight one.

Figure 3: Distribution of different edge weights over the 3 datasets. SO predominantly consists of edges with weight one. Right hand plot is a scaled version of the left. The label '11' contains edge weights of 11 and above.

than each of its individual components. SO dataset described in section 4 is used as the data for this experiment. The number of parallel cores used in the PSSV scheme is four whereas its one for the rest of the three. The amount of sub-sampled forum threads is 400 and the total number of threads is 14,416. All the schemes in the graph start with the same initialization values of the hyper-parameters. PSSV is atleast twice as fast as the nearest scheme besides obtaining the best log-likelihood of all the four at the point of convergence. The SV (stochastic variational) samples one thread at a time and therefore takes some time in the beginning to start minimizing the objective value (negative log likelihood). The objective value increases in the first few iterations for SV. The number of iterations to be done by SV is very large but each iterations takes the smallest time of all four. The V (variational) scheme takes the least number of iterations to converge though its iterations are the most time consuming as it has to go through all the 14,416 threads in every iteration.

System details.
The machine used in all the experiments in this paper is "Intel(R) Xeon(R) CPU E5-2450 0 @ 2.10GHz" 16 corewith 8GBs of RAM per core. The operating system is Linux 2.6.32 x86_64.

4. DATASETS

We analyse three real world datasets corresponding to two different forums: 1) Cancer-ThreadStarter, 2) Cancer-UserName, and 3) Stack Overflow. To test the stability of the model we use a synthetically generated dataset. The Cancer forum [1] is a self-help community where users who either have cancer, are concerned they
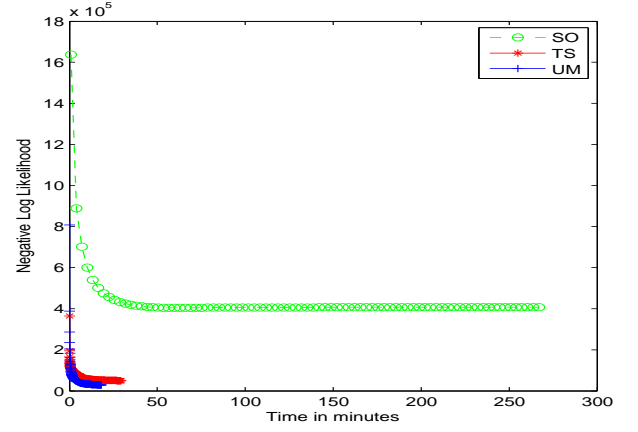

Figure 4: The log-likelihood over heldout set for the fully tuned model on the 3 datsets

may have cancer, or care for others who have cancer, come to discuss their concerns and get advice and support. StackOverflow is an online forum for question answering primarily related to computer science. We use the latest dump of Stack Overflow [2]. In each of these datasets a user posts multiple times in a thread and all these posts are aggregated into one bigger posts per thread as defined in section 2.2. Number of times a user u replies to user v in thread T is the edge weight of edge $u \rightarrow v$ in thread T. Table 3 gives the distributions of edges, posts, users and threads in the three datasets used.

4.1 Cancer-ThreadStarter (TS)

In the Cancer forum, the conversations happen in a structured way where users post their responses on a thread by thread basis. Every thread has a thread starter that posts the first message and starts the conversation. We construct a graph from each thread by drawing a link from each participant on the thread to the participant who started the thread This graph has 22,095 users and 14,416 Threads.

4.2 Cancer-Username Mention (UM)

Users call each other by their usernames (or handle assigned to them in the forum) while posting in many cases. We create a graph where in an edge between user u and user v in thread t means that user u calls user v by username in thread t. This graph has 15,111 users and 12,440 threads.

4.3 Stack Overflow (SO)

In Stack Overflow users ask questions and then other users reply with their answers. We obtain the ThreadStarter graph from this structure. This dataset has have 1,135,996 users and 4,552,367 threads.

4.4 Synthetic data

We generate a synthetic dataset using the generative process defined in section 2.2. We have 1000 users and 100 threads. The number of posts and edges vary depending on the choice of priors α and η

	α	ω	θ	κ	η	K
TS	0.05	1e-4	2.5∼1.5	2.5∼1.5	0.05	10
UM	0.05	1e-3	2.0∼1.0	2.0∼1.0	0.05	10
SO	0.05	1e-2	1.0∼0.5	1.0∼0.5	0.05	20

Table 2: Tuned values for the parameters. θ and κ are matrices and $a \sim b$ assigned to them means diagonal values are a and non-diagonals are b. K is the number of topics

5. EXPERIMENTAL SETUP AND EVALU-ATION

We divide each dataset into three subsets: 1) the training set, 2) the heldout set, and 3) the test set. We learn our model on training set and tune our priors ($\alpha, \eta, \kappa, \theta$ etc.) on heldout set. The split is done over the edges where 80% of the edges are in training and rest 20% are divided amongst heldout and test equally. For the two cancer datasets we only predict non-zero edge weights whereas for the Stack Overflow we predict zero as well as non-zero edge weights. Graph 3 shows the distribution of edge weights in cancer and Stack Overflow dataset. We chose Stack Overflow to predict zero weights since it has large number of edges with very low weights, predominantly weight one. Predicting zero as well as non-zero edge weights demonstrates that the model is versatile and can predict a wide range of edge-weights. In addition to 20% of the total non-zero edges we randomly sample equal number of zero edges from the graph for the SO held and test set. The optimization objective for learning is defined in equation 18.

A link prediction task is incorporated to demonstrate the model's effectiveness. It is a standard task in the area of graph clustering and social networks in particular. Researchers have used it in the past to demonstrate their model's learning ability [12]. The link prediction task works as an important validation of our model. If the proposed model performs better than its individual parts then it can be safely concluded that it extracts important patterns from each of its building blocks. Moreover it adds validity to the qualitative analysis of the results.

Link-prediction.

We predict the edge-weight of the edges present in the test set. The predicted edge, $\hat{Y}_{t,u,v}$, between users u and v in thread t is defined as

$$\hat{Y}_{t,u,v} = \pi_u^T B \pi_v. \qquad (10)$$

$$B = \nu. * \lambda \qquad (11)$$

and the prediction error is the $rmse$, defined as given predicted edge $\hat{Y}_{t,u,v}$ and the actual edge $Y_{t,u,v}$,

$$rmse = \sqrt{\sum (\hat{Y}_{t,u,v} - Y_{t,u,v})^2}. \qquad (12)$$

The summation is over the edges in the test (or heldout) set. The block matrix B described in equation 11 is well defined for both MMSB and the proposed model. Hence the prediction is obtained for the active network modelling without LDA (just MMSB component) and with LDA. We created an artificial weighted Identity matrix for LDA $\hat{B} = m * I$. It is a diagonal matrix with all element values m. For every user u and every thread t the topics discovered over the posts of u in t is used as the vector π_u in equation 10 for prediction. A diagonal B is desirable in block models as it provides
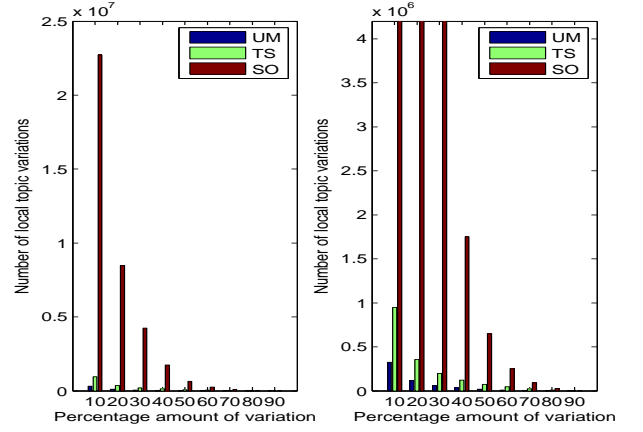

Figure 5: Number of local variations in topic proportion on a per user per thread level. The axis is percentage variation (from 10 to 90 percent). The right hand plot is a scaled in version of the left.

clean separation among the clusters obtained [1]. The value of m is tuned over heldout set. We define a basic baseline that always predicts the average weight ($\bar{Y}$) all the edges, zero (Stack Overflow) or non-zero (Cancer), in the heldout or test set.

$$rmse_{baseline} = \sqrt{\sum (\bar{Y} - Y_{t,u,v})^2}. \qquad (13)$$

Parameter tuning.

We tune our parameters $\eta, \kappa, \theta, \alpha$, and K (number of community-topics) over the held set. ω, the parameter to balance the contribution of the text side to the network side is tuned over the heldout set. It is used in the local variational update of ϕ(equation 19). Equation 19 contains a summation term over all the tokens $\sum_{i=1}^{N_{T_p}} \chi_i$ in the per user per thread document and if not balanced by ω will dominate the rest of the terms. The constant ϵ used in equation 19 and 20 is a smoothing constant and is fixed at a low value. The six quantities, $\alpha, \omega, \theta, kappa, \eta$ and K are tuned in that sequence. α is tuned first keeping rest constant then ω and so on where each next to be tuned parameter uses values of already tuned parameters. Table 4.4 shows the final values of all the parameters.

Figure 4 shows plot of tuned log-likelihood over the 3 datasets against time. UM being the smallest of the two takes the least amount of time.

6. RESULTS

Link prediction.

Table 3 shows the link prediction results on heldout and test set for the for the four prediction model.

The proposed approach to model thread level conversational roles outperforms all of the other models. LDA performs poorer than MMSB since LDA does not explicitly model network information.

Cancer dataset.

Figure 5 shows the number of times the global role of a user is different from the thread level role that he plays. It is interesting to see that the variation between global and thread level role assignment is high among all the datasets. A model that ignores this local
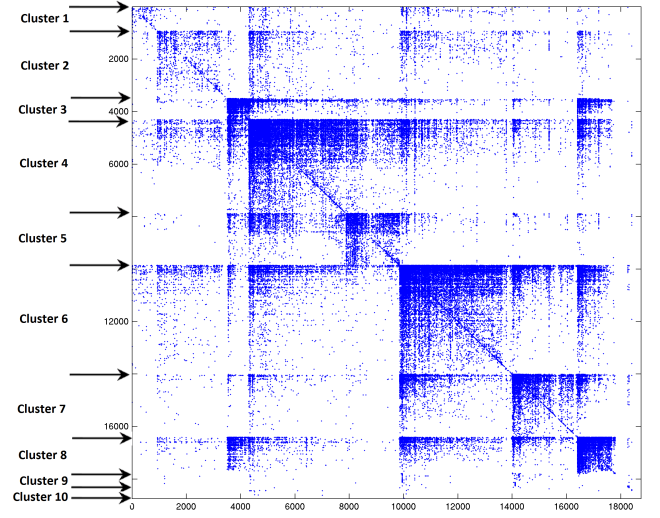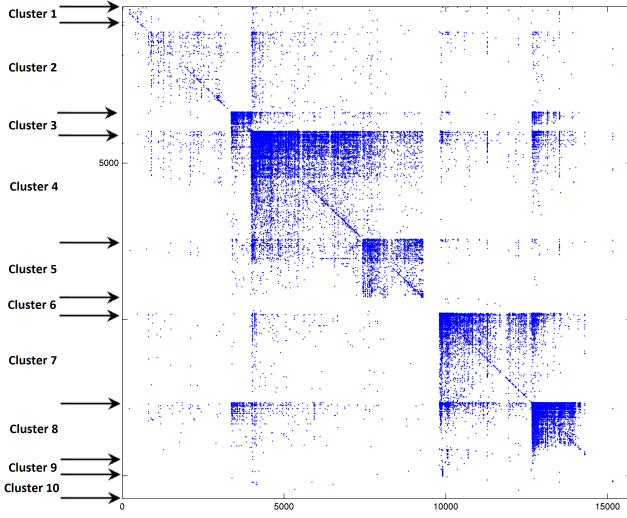
Figure 6: Adjacency matrix of users sorted by clusters. Left side is clustered by MMSB and right side is clustered by our model using user's dominant role as cluster index over TS dataset. Our model is able to correctly cluster 3K additional users that MMSB doesnt assign any dominant cluster (or role) and discovers a new role (Cluster-6).

	UM held	UM test	TS held	TS test	SO held	SO test
Our Model	**1.303**	**1.292**	**2.782**	**2.748**	**0.348**	**0.361**
MMSB	1.450	1.502	2.984	2.881	0.421	0.434
LDA	1.793	1.731	3.725	3.762	0.466	0.479
Baseline	1.886	1.983	4.504	4.417	0.502	0.509

Table 3: Link prediction results over the 3 datsets

vs global dynamics tends to lose a lot of information. Figure 6 shows the plot of the user by user adjacency matrix for TS dataset. The users are sorted based on the community-topic cluster (roles) assigned by the respective models (our model and MMSB model). The number of community-topics are 10 and every user is assigned the dominant community-topic, π, (role) that they have more than 50% of chance of lying in. A user is discarded if he doesn't have the said dominant role. Our model is unable to assign a clear role to 3.3K users and the MMSB approximately to 6.3K users out of 22K. Based on the topics assigned, users are sorted and their adjacency matrix shows clean clustering along the block diagonals. As seen in the figure, the combined model is able to effectively find the primary roles (dominant topic) for the extra 3K users that the MMSB model was unable to provide for. Besides a new role (Role 6) that is not accounted for by MMSB is discovered by the proposed model (figure 6). Users that predominantly have role 6 on a global scale tend to vary their roles often on a thread level, i.e. their topic probabilities change quite often. The average change in topic probabilities per role per user-thread pair across the 10 roles discovered in TS is 30.6%; for role 6 it is 41.5% (highest of all the roles). This means that this role is very dynamic and an active sub-network modelling helps here as it captures the dynamism of this entity. From figure 6 cluster of roles 4, 5, 6, 7 and 8 are the largest. Table 4 corresponds to top 15 words corresponding to these roles. Role 4, role 7 and role 8 are related to discussion regarding cancer where as role 5 is related to conversations regarding spiritual
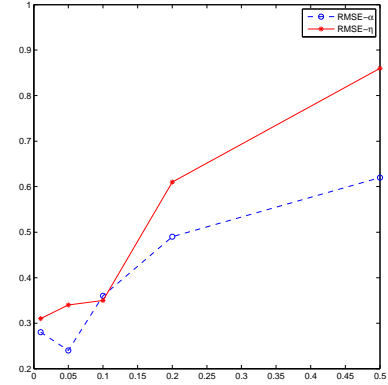

Figure 8: RMSE vs α and η for the synthetic dataset. The X-axis is α or η values and the Y-axis is the RMSE of the recovered π.

and family matters. But role 6 does not seem to be related to any specific type of conversation. It is free flowing and has lots of non specific words which tells us that there is a cornucopia of discussions happening in this role with no specific matter at hand. This fact is also verified by looking at the raw Cancer forum data. Users who are predominantly in this role tend to post across many discussion threads and variety of conversation topics. This role is detected by our model due to the fact that it takes into account the dynamics of such a role.

Stack Overflow.

The optimal topic number for SO dataset is 20 community-topics as noted in table 4.4 and the number of users are 1.13 million. It is difficult to visualize the user-user adjacency matrix of this size. The 20 topic set is divided into four sets with size 5 each. Topics 1 to 5 form set one, topics 6 to 10 form set two and so on. Every user's role is visualized by projecting user's π over a pentagon as shown in
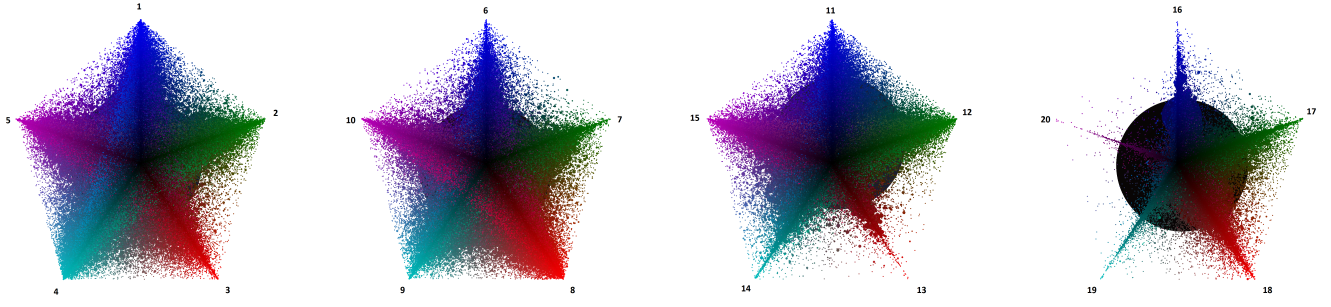
Figure 7: The 20 roles assigned to users in the stack overflow dataset. The numbers at the vertex are the role numbers. Due to the large number of roles we visualize them 5 at a time with first 5 first then second 5 and so on. We can see that the roles are separated cleanly and clustered around the pentagon corners.

Topic 4	Topic 5	Topic 6	Topic 7	Topic 8
side	same	their	surgeon	radiat
test	life	live	everi	anoth
took	tell	happi	found	doctor
away	mani	mayb	down	problem
left	famili	sorri	alway	pleas
support	prayer	best	while	person
doesn	though	check	home	kind
seem	ladi	news	bodi	soon
move	until	question	these	each
almost	wish	dure	bone	hard
scan	someon	deal	mean	might
idea	under	case	came	medic
studi	felt	mind	posit	herceptin
guess	where	seem	drug	share
diseas	nurs	haven	send	free

Table 4: top 15 words for topics corresponding to top 5 biggest role in TS.

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6
public	code	function	that	name	array
valu	should	also	view	method	more
chang	your	then	thread	time	what
when	user	object	into	document	system
event	properti	creat	control	line	about
would	blockquot	follow	current	element	sure
list	just	form	link	differ	post
databas	field	implement	oper	would	each
there	class	valu	question	defin	imag
issu	overrid	server	length	file	class
path	main	veri	creat	result	paramet
display	result	string	each	where	applic
like	size	start	result	more	just
order	import	java	blog	know	local
save	project	android	featur	browser	specif

Table 5: top 15 words for word topics corresponding to first 6 role clusters in SO.

figure 7. The projection uses both position and color to show values of community-topic π for each user. Every user u is displayed as a circle (vertex) v_u in the figure where the size of the circle is the node degree of v_u and position of v_u is equal to a convex combination of the five pentagon corner coordinates (x, y) that are weighted by the elements of π_u. Hence circles v_u at the pentagon's corners represent π's that have a dominating community in the 5 community-topics chosen, while circles on the lines connecting the corners represent π's with mixed-membership in at least 2 communities (as only a partial π vector is used in each sub-graph). All other circles represent π's with mixed-membership in ≥ 3 communities. Each circle v_u's color is also a π-weighted convex combination of the RGB values of 5 colors: blue, green, red, cyan and purple. This color coding helps distinguish between vertices with 2 versus 3 or more communities. We observe a big black circle at the back ground of every plot. This circle represents the user with id 22656 [3] that has the highest node-degree of 25,220 in the SO dataset. This user has the highest all time reputation on stack overflow and tends to take part in myriads of question answering threads. Hence he is rightly picked up by the model to be in the middle of all the roles.

Figure 7 has a clean clustering where the nodes are clustered

---

[3] http://stackoverflow.com/users/22656/jon-skeet

around the pentagon vertices. This indicates that the model is able to find primary roles for most of the users in the forum. Though table 5 shows that there is significant amount of variation with respect to the global role of a user at a thread level. Modeling this variation helps our model in getting better clusters as compared to simple MMSB; this fact is apparent from the link prediction task too, table 3. We get an rmse of 0.348 on heldout and 0.361 on test set which is better than all the other 3 approaches.

Table 5 shows the top 15 words corresponding to first 6 roles discovered in SO. While roles 1, 2, 3 are pertinent to discussions regarding database conversations, general coding, android and J2EE, role 4 relates to online blogs and browsing apis. Role 5 and role 6 are related to server/client side browser based coding and general coding respectively.

Do demonstrate the effectiveness of our model on this dataset, we take the example of user 20860. User id 20860 is globally assigned role 1 as the dominant role but he also takes part in coding related discussions. For example,

- *Because the join() method is in the string class, instead of the list class? I agree it looks funny.*

- *The simplest solution is to use shell_exec() to run the mysql client with the SQL script as input. This might run a little slower because it has to fork, but you can write the code in a*

couple of minutes and then get back to working on something useful. Writing a PHP script to run any SQL script could take you weeks....

But in most of the cases he visits general software or coding questions that are specifically related to databases and this fact is picked up by our model and it assigns him predominantly (>0.8) a database role even though he is active contributor to general software and coding discussions. MMSB on the other hand assins him 30% database (role 1) 30% general coding (role 2) and rest is distributed among the remaining 18 roles.

The model picks up other similar cases for which it is able to successfully distinguish (compared to MMSB) between user's global and local roles even though they are dynamic in nature.

Synthetic dataset.

Figure 8 gives the rmse of the model for the recovery of community topic π over the synthetic dataset. From the graph, higher values of the parameters make it harder to recover the pi values. For this experiment we fix topic number at 5 and vary α or η by fixing the other at 0.01. The other priors such as κ, θ, ω etc. are fixed at the values used to generate the dataset. It is apparent that the rmse is more sensitive towards α values and recovers them well compared to η. The results are averaged over 20 random runs of the experiment for the given values of α and η. The rmse achieved for lower values of priors α and η is very promising as it means that the confidence interval of the estimate is very high for sparse priors for this model given sufficient data.

7. RELATED WORK

White et al.[17] proposed a mixed-membership model that obtained membership probabilities for discussion-forum users for each statistic (in- and out-degrees, initiation rate and reciprocity) in various profiles and clustered the users into "extreme profiles" for user role-identification and clustering based on roles in online communities,. Ho et al. [7] presented TopicBlock that combines text and network data for building a taxonomy for a corpus. The LDA model and MMSB models were combined by Nallapati et al. [12] using the Pairwise-Link-LDA and Link-PLSA-LDA models where documents are assigned membership probabilities into bins obtained by topic-models. Sachan et al. [13] provide a model for community discovery that combines network edges with hashtags and other heterogeneous data and use it to discover communities in twitter and Enron email dataset.

For simultaneously modeling topics in bilingual-corpora, Smet et al. [16] proposed the Bi-LDA model that generates topics from the target languages for paired documents in these very languages. The end-goal of their approach is to classify any document into one of the obtained set of topics. For modeling the behavioral aspects of entities and discovering communities in social networks, several game-theoretic approaches have been proposed (Chen et al. [2], Yadati and Narayanam [19]). Zhu et al. [20] combine MMSB and text for link prediction and scale it to 44K links.

Ho et al. [8] provide unique triangulated sampling schemes for scaling mixed membership stochastic block models [1] to the order of hundreds of thousands of users. Prem et al. [6] use stochastic variational inference coupled with sub-sampling techniques to scale MMSB like models to hundreds of thousands of users.

None of the works above address the sub-network dynamics of thread based discussion in online forums. Our work is unique in this context and tries to bring user role modelling in online social networks closer to the ground realities of online forum interactions. Active sub-network modelling has been used recently to model gene interaction networks [11]. They combine gene expression data with network topology to provide bio-molecular sub-networks, though their approach is not scalable as they use simple EM for their inference. We leverage the scalable aspects of SVI [9] to combine MMSB (network topology) with LDA (post contents) in a specific graphical structure (thread structure in the forum) to obtain a highly scalable active sub-network discovery model.

Matrix factorization and spectral learning based approaches are some of the other popular schemes for modelling user networks and content. In recent past both approaches have been made scalable to a million order node size graph [5, 4]. But these methods are unable to incorporate the rich structure that a probabilistic modelling based method can take into account. For example modelling active sub-networks besides incorporating content as well as network graph will be very hard to achieve in matrix factorization or spectral clustering paradigm.

8. DISCUSSION AND FUTURE WORK

The proposed model relies on the fact that forum users have dynamic role assignments in online discussions and leveraging this fact helps to increase prediction performance as well as understand the forum activity. The model performs very well in its prediction tasks. It outperforms all the other methods over all the datsets by a huge margin. The model is scalable and is able to run on social network dataset of unprecedented content size. There is no past research work that scales forum contents to more than one million user and around 10 million posts.

The idea that active subnetwork is useful in modelling online forums is demonstrated qualitatively and quantitatively. Quantitatively it provides better prediction performance and qualitatively it captures the dynamics of user roles in forums. This dynamism can help us find new cluster roles that may have been missed by state of the art clustering approaches, as we observed for TS dataset. From the synthetic experiments it is observed that the model recovers its parameters with high likelihood with sparse priors. This works to its advantage for scalable learning as big data sets tend to be sparse.

The scalability aspects of the inference scheme proposed here are worth noting. Besides the multi-core and stochastic sub-sampling components of the proposed inference, the use of Poisson to model the edge weights has enabled us to ignore zero-edges if need be. This reduces the amount of work needed for learning the network parameters. The learned network is at par with the state of the art inference schemes as demonstrated in the prediction tasks.

One aspect to explore in future is to combine multiple types of links in network. For example in many online forums users explicitly friend other users, follow other users or are members of same forum related sub-groups as other users. All these relations can be modelled as a graph. It is worth finding out how important is modelling active sub-network in such a case. It is possible that various types of links might reinforce each other in learning the parameters and thus will obviate the need to model a computationally costly sub-network aspect. As we saw in figure 8 that sparsity helps, but how sparser can we get before we start getting poor results needs some exploration.

As we have seen, figure 8, that the model recovers the community-topic parameters with very high likelihood for lower values of model priors α and τ. If this is a general attribute of active sub-network models then it can be leveraged for sparse learning. Moreover, although in case of large online forums modelling active sub-networks is computationally challenging and costly, the sparsity aspects of active sub-networks might help reduce the computation costs.

9. 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] Wei Chen, Zhenming Liu, Xiaorui Sun, and Yajun Wang. A game-theoretic framework to identify overlapping communities in social networks. *Data Min. Knowl. Discov.*, 21(2):224–240, September 2010.

[3] Raamesh Deshpande, Shikha Sharma, Catherine M. Verfaillie, Wei-Shou Hu, and Chad L. Myers. A scalable approach for discovering conserved active subnetworks across species. *PLoS Computational Biology*, 6(12), 2010.

[4] Inderjit Dhillon, Yuqiang Guan, and Brian Kulis. A fast kernel-based multilevel algorithm for graph clustering. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, KDD '05, pages 629–634, New York, NY, USA, 2005. ACM.

[5] Rainer Gemulla, Erik Nijkamp, Peter J. Haas, and Yannis Sismanis. Large-scale matrix factorization with distributed stochastic gradient descent. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '11, pages 69–77, New York, NY, USA, 2011. ACM.

[6] Prem Gopalan, David M. Mimno, Sean Gerrish, Michael J. Freedman, and David M. Blei. Scalable inference of overlapping communities. In Peter L. Bartlett, Fernando C. N. Pereira, Christopher J. C. Burges, Léon Bottou, and Kilian Q. Weinberger, editors, *NIPS*, pages 2258–2266, 2012.

[7] Qirong Ho, Jacob Eisenstein, and Eric P. Xing. Document hierarchies from text and links. In *Proceedings of the 21st international conference on World Wide Web*, WWW '12, pages 739–748, New York, NY, USA, 2012. ACM.

[8] 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, Léon Bottou, and Kilian Q. Weinberger, editors, *NIPS*, pages 2141–2149, 2012.

[9] Matthew D. Hoffman, David M. Blei, Chong Wang, and John Paisley. Stochastic variational inference. *J. Mach. Learn. Res.*, 14(1):1303–1347, May 2013.

[10] Brian Karrer and M. E. J. Newman. Stochastic blockmodels and community structure in networks. *CoRR*, abs/1008.3926, 2010.

[11] Ilana Lichtenstein, Michael Charleston, Tiberio Caetano, Jennifer Gamble, and Mathew Vadas. Active Subnetwork Recovery with a Mechanism-Dependent Scoring Function; with application to Angiogenesis and Organogenesis studies. *BMC Bioinformatics*, 14(1):59+, 2013.

[12] 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.

[13] 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, New York, NY, USA, 2012. ACM.

[14] Jianbo Shi. Learning segmentation by random walks. In *In Advances in Neural Information Processing*, pages 470–477. MIT Press, 2000.

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

[16] Wim De Smet, Jie Tang, and Marie-Francine Moens. Knowledge transfer across multilingual corpora via latent topics. In *Proceedings of the 15th Pacific-Asia conference on Advances in knowledge discovery and data mining - Volume Part I*, PAKDD'11, pages 549–560, Berlin, Heidelberg, 2011. Springer-Verlag.

[17] Arthur White, Jeffrey Chan, Conor Hayes, and Brendan Murphy. Mixed membership models for exploring user roles in online fora, 2012.

[18] Eric P. Xing, Michael I. Jordan, and Stuart J. Russell. A generalized mean field algorithm for variational inference in exponential families. *CoRR*, abs/1212.2512, 2012.

[19] Narahari Yadati and Ramasuri Narayanam. Game theoretic models for social network analysis. In *Proceedings of the 20th international conference companion on World wide web*, WWW '11, pages 291–292, New York, NY, USA, 2011. ACM.

[20] Y. Zhu, X. Yan, L. Getoor, and C. Moore. Scalable Text and Link Analysis with Mixed-Topic Link Models. *ArXiv e-prints*, March 2013.

APPENDIX

The log-likelihood of the model:

$$\log L = \log P(Y, W, Z_\leftarrow, Z_\rightarrow, \Pi, B, \beta | \alpha, \eta, \theta, \alpha)$$

$$= \sum_t \Bigg[\sum_{p,q} \log P(Y_{t,p,q} | Z_{t,p\rightarrow q}, Z_{t,p\leftarrow q}, B)$$

$$+ \sum_{p,q} \log P(Z_{t,p\rightarrow q} | \Pi_q)$$

$$+ \sum_{p,q} \log P(Z_{t,p\leftarrow q} | \Pi_q) \Bigg] + \sum_p \log P(\Pi_p | \alpha)$$

$$+ \Bigg[\sum_{t=1}^{T} \sum_{p \in t} \sum_{i=1}^{N_{T_p}} \log P(w_{t,p,i} | Z'_{t,p,i}, \beta)$$

$$+ \sum_{t=1}^{T} \sum_{p \in t} \sum_{i=1}^{N_{T_p}} \log P(Z'_{t,p,i} | \bar{Z}_{t,p\rightarrow q}) \Bigg]$$

$$+ \sum_k \log P(\beta_k | \eta) + \sum_{g,h} \log P(B_{g,h} | \kappa, \theta).$$

$$(14)$$

The data likelihood for the model in figure 1

$$P(Y, R_p | \alpha, \beta, \kappa, \eta) =$$

$$\int_\Phi \int_\Pi \sum_z P(Y, R_p, z_{p\rightarrow q}, z_{p\leftarrow q}, \Phi, \Pi | \alpha, \beta, \kappa, \eta)$$

$$= \int_\Phi \int_\Pi \sum_z \Bigg[\prod_{p,q} \prod_t P(Y_{pq}^t | z_{p\rightarrow q}^t, z_{p\leftarrow q}^t, B) \cdot P(z_{p\rightarrow q}^t | \Pi_p)$$

$$\cdot P(z_{p\leftarrow q}^t | \Pi_q) \cdot \Bigg(\prod_p P(\Pi_p | \alpha) \prod_t \prod_p P(R_p^t | z_{p\rightarrow q}^t, \Phi)$$

$$\cdot \prod_k P(\Phi_k | \beta) \Bigg) \cdot \prod_{g,h} P(B_{gh} | \eta, \kappa) \Bigg].$$

$$(15)$$

The complete log likelihood of the model is:

$$
\begin{aligned}
\log P(&Y, W, z_\rightarrow, z_\leftarrow, \Phi, \Pi, B | \kappa, \eta, \beta, \alpha) \\
&= \sum_t \sum_{p,q} \log P(Y_{pq}^t | z_{p\rightarrow q}^t, z_{p\leftarrow q}^t, B) \\
&\quad + \sum_t \sum_{p,q} (\log P(z_{p\rightarrow q}^t | \Pi_p) + \log P(z_{p\leftarrow q}^t | \Pi_p)) \\
&\quad + \sum_p \log P(\Pi_p | \alpha) + \sum_t \sum_p \sum_{w \in R_p^t} \log P(w | z_{p\rightarrow}, \Phi) \\
&\quad + \sum_k \log P(\Phi_k | \beta) + \sum_{gh} \log P(B_{gh} | \eta, \kappa). \quad (16)
\end{aligned}
$$

The mean field variational approximation for the posterior is

$$
\begin{aligned}
q(z, &\Phi, \Pi, B | \Delta_{z_\rightarrow}, \Delta_\Phi, \Delta_B, \Delta_{z_\leftarrow}, \Delta_{B_\kappa}) = \\
&\prod_t \prod_{p,q} \left(q_1(z_{p\rightarrow q}^t | \Delta_{z_{p\rightarrow q}}) + q_1(z_{p\leftarrow q}^t | \Delta_{z_{p\leftarrow q}}) \right) \\
&\cdot \prod_p q_4(\Pi_p | \Delta_{\Pi_p}) \prod_k q_3(\Phi_k | \Delta_{\Phi_k}) \prod_{g,h} q(B_{g,h} | \Delta_{B_\eta}, \Delta_{B_\kappa}). \quad (17)
\end{aligned}
$$

The lower bound for the data log-likelihood from jensen's inequality is:

$$
\begin{aligned}
L_\Delta &= E_q \left[\log P(Y, W, z_\rightarrow, z_\leftarrow, \Phi, \Pi, B | \kappa, \eta, \beta, \alpha) - \log q \right] \\
&= E_q \left[\sum_t \sum_{p,q} \log \left(B_{g,h}^{Y_{p,q}^t} \frac{e^{-B_{gh}}}{Y_{pq}^t!} \right) + \sum_t \sum_{pq} \log \left(\prod_k (\pi_{p,k}^{z_{p\rightarrow q}=k}) \right) \right. \\
&\quad + \sum_t \sum_{p,q} \log \left(\prod_k (\pi_{q,k})^{z_{p\leftarrow q}=k} \right) \\
&\quad + \sum_t \sum_p \sum_{w \in R_p^t} \log \left(\prod_{u \in V} (\bar{z}^T \phi_u)^{w=u} \right) \\
&\quad + \sum_p \log \left(\prod_k (\Pi_{p,k})^{\alpha_k - 1} \cdot \frac{\Gamma(\sum \alpha_k)}{\prod_k \Gamma(\alpha_k)} \right) \\
&\quad + \sum_k \log \left(\prod_{u \in V} (\phi_{k,u})^{\beta_k - 1} \cdot \frac{\Gamma(\sum \beta_k)}{\prod_k \Gamma(\beta_k)} \right) \\
&\quad + \sum_{g,h} \log \left(B_{g,h}^{\kappa - 1} / \eta^\kappa \Gamma(\kappa) \cdot \exp(-B_{g,h}/\eta) \right) \Big] \\
&\quad - E_q \left[\sum_t \sum_{p,q} \log \left(\prod_k (\Delta_{z_{p\rightarrow q},k})^{z_{p\rightarrow q}=k} \right) \right. \\
&\quad + \sum_t \sum_{p,q} \log \left(\prod_k (\Delta_{z_{p\leftarrow q},k})^{z_{p\leftarrow q}=k} \right) \\
&\quad + \sum \log \left(\prod_k (\Pi_{p,k})^{\Delta_{\pi_{pk}}-1} \frac{\Gamma(\Delta_{\Pi_p})}{\prod_{k=1} \Gamma(\Delta_{\Pi_{p,k}})} \right) \\
&\quad + \sum_k \log \left(\prod_{u \in v} (\Phi_{k,u})^{\Delta_{\Phi_{ku}}-1)} \frac{\Gamma(\Delta_{\Phi_k})}{\prod_{u \in v} \Gamma(\Delta_{\Phi_{k,u}})} \right) \\
&\quad + \sum_{g,h} \log \left(\frac{B_{g,h}^{\Delta_{\kappa=1}}}{\Delta_\eta^{\Delta_\kappa} \Gamma(\Delta_\kappa)} \exp(-B_{g,h}/\Delta_\eta) \right) \Big]. \quad (18)
\end{aligned}
$$

Δ_ϕ used in the update of ϕ in equation 3. The parameter ω is used here to balance out the contribution from the text side to the network.

$$
\begin{aligned}
\Delta_{\phi'_{t,p,g,h}} &= y_{t,p,q}(\log \lambda_{g,h} + \Psi(\nu_{g,h})) - \nu_{g,h}\lambda_{g,h} - \log(y_{t,p,q}!) \\
&\quad + \Psi(\gamma_{p,g}) - \Psi\left(\sum_g \gamma_{p,g} \right) \\
&\quad + \Psi(\gamma_{q,h}) - \Psi\left(\sum_h \gamma_{q,h} \right) \\
&\quad + \omega \sum_{i=1}^{N_{T_P}} \chi_{t,p,i,g} \left[\ln \frac{\epsilon}{\delta_{p,t}} - \frac{1}{\delta_{t,p}} + \ln \left(1 + \frac{\epsilon}{\delta_{p,t}} \right) \cdot \frac{1}{\delta_{t,p}} \right]. \quad (19)
\end{aligned}
$$

Δ_χ used in equation 4 for χ update

$$
\begin{aligned}
\Delta_{\chi'_{t,p,g,h}} &= \left[\Psi(\tau_{k,w_{t,p,i}}) - \Psi\left(\sum_{w_{t,p,i}} \tau_{k,w_{t,p,i}} \right) \right] \\
&\quad + \ln(\frac{\epsilon}{\delta p,t}) \frac{1 - \sum_{q,h} \phi_{t,p,q,k,j}}{\delta_{t,p}} \\
&\quad + \frac{\sum_{q,h} \phi_{t,p,q,k,h}}{\delta_{t,p}} \ln(1 + \frac{\epsilon}{\delta_{t,p}}). \quad (20)
\end{aligned}
$$

Partial derivative of ν

$$
\begin{aligned}
\frac{dL}{d\nu_{g,h}} &= \sum_t \sum_{p,q} \phi_{t,p,q,g,h}(y_{t,p,q}\Psi'(\nu_{g,h}) - \lambda_{g,h}) \\
&\quad + (\kappa_{g,h} - \nu_{g,h})\Psi'(\nu_{g,h}) + 1 - \frac{\lambda_{g,h}}{\theta_{g,h}}. \quad (21)
\end{aligned}
$$

The traditional variational updates for the global parameters

$$
\gamma_{p,k} = \alpha_k + \sum_t \sum_q \sum_h \phi_{t,p,q,k,h} + \sum_t \sum_q \sum_g \phi_{t,q,p,g,k}. \quad (22)
$$

$$
\nu_{g,h}^{t+1} = \nu_{g,h}^t + \rho_\nu \frac{dL}{d\nu_{g,h}}. \quad (23)
$$

$$
\lambda_{g,h} = \frac{\left(\sum_t \sum_{p,q} \phi_{t,p,q,g,h} y_{t,p,q} + \kappa_{g,h} \right)}{\left(\left(\sum_t \sum_{p,q} \phi_{t,p,q,g,h} \right) + \frac{1}{\theta_{g,h}} \right) \nu_{g,h}}. \quad (24)
$$

$$
\tau_{p,v} = \nu_v + \sum_t \sum_{p \in t} \left(\sum_{w_{t,p,i}=v}^{N_{t,p}} \chi_{t,p,i,k} \right). \quad (25)
$$

where ρ_ν is ν's gradient ascent update step-size using its partial derivative $\frac{dL}{d\nu_{g,h}}$ define in equation 21.