FRec: A Novel Framework of Recommending Users and Communities in Social Media

Lei Li† Wei Peng‡ Saurabh Kataria‡ Tong Sun‡ Tao Li†

†School of Computing and Information Sciences
Florida International University
Miami, FL 33199

‡Xerox Innovation Group
Xerox Corporation
Webster, NY 14580

ABSTRACT

In this paper, we propose a framework of recommending users and communities in social media. Given a user's profile, our framework is capable of recommending influential users and topic-cohesive interactive communities that are most relevant to the given user. In our framework, we present a generative topic model to discover user-oriented and community-oriented topics simultaneously, which enables us to capture the exact topic interests of users, as well as the focuses of communities. Extensive evaluation on a data set obtained from Twitter has demonstrated the effectiveness of our proposed framework compared with other probabilistic topic model based recommendation methods.

Categories and Subject Descriptors: H.3.3[Information Search and Retrieval]: Information filtering

Keywords: User Recommendation; Community Recommendation; Social Media; Topic Modeling

1. INTRODUCTION

In social media, a community is often formed by a collection of users with social connections as well as similar topic preferences. Taking online marketing campaign as an example, marketers not only target individuals with certain interest, but also hope the marketing messages could be cascaded to more audience sharing similar interests. In such a scenario, one critical issue of utilizing social media data is how to precisely identify users' personal interest and the interest of communities where these users are connected to or frequently interact with. Thus it is very important to capture both user-oriented and community-oriented topics.

Automated discovery of topics and communities has received widespread attention in academia and has been addressed differently in previous works. A common approach is to use generative Bayesian models to capture the correlations among users, communities and topics. However, prior approaches cannot make a distinction between user-oriented and community-oriented topics. Taking a query "campaign + economy" as an example, the task is to identify users and communities that are interested in US presidential campaign and also often discuss the topic of economy related to the campaign. "campaign" is discussed by a lot of people as it is relevant to the presidential selection, whereas " economy" often appears in users' general posts and may not be related to "campaign". In this case,

- if we only consider user-oriented topics, the recommended users identified to be interested in the query are not necessarily connected to the communities focusing on the query-related topics. Targeting these users will not guarantee the marketing messages to further cascade in the social network. In addition, the extreme versatility of users interest, informal writing, and spam in the social network make it difficult to infer communities interests with reasonable perplexity.

- if we only consider community-oriented topics based on posts by all the users in the communities, the fine-grained topic interest of each individual user is difficult to model due to the coarse community-oriented topic structure. Also, detecting topics in an indiscriminate way will result in a lot of noise since all the user-generated content will contribute to the community topics. Therefore, we cannot identify the source from which "economy" is originated.

The advantage of modeling user-oriented and community-oriented topics simultaneously is that it could identify high-quality community topics by sampling the topic for each word from either the community topic-word distribution or the user topic-word distribution. Thus the noises induced by a wide variety of user interests that could contaminate the community topics can be naturally mitigated.

In our work, we identify the latent relationships among social objects, i.e. users and communities, by distinguishing a user's interest from interests of communities. We propose a generative topic model to capture both types of interests as topics in a parameter universe with a mechanism that identifies the association of interests to either a given user or a given community. Our proposed model makes use of the communities derived from the social links of users to avoid the expensive computation of combining the community discovering process with the topic modeling process. We further provide a novel recommendation framework, named `FRec`, based upon the derived relationships, which is able to recommend topic-related influential users and topic-cohesive interactive communities for a given user's profile.

The contribution of our work is three-fold: (1) A modeling approach to distinguishing community v.s. user interests

CIKM'13, Oct. 27–Nov. 1, 2013, San Francisco, CA, USA.

http://dx.doi.org/10.1145/2505515.2505645.

(cf. §3) by using a Bernoulli variable to control the distribution from which a word is drawn; (2) A principled framework (cf. §4) which is capable of recommending topic-related influential users and topic-cohesive interactive communities given a user profile; and (3) Extensive evaluation (cf. §5) on a Twitter dataset related to "presidential campaigns" that demonstrates the effectiveness of the framework.

The rest of the paper is organized as follows. §2 presents a brief summary of prior work relevant to community-based topic models and recommender systems. §3 discusses the proposed topic modeling approach, and §4 describes the recommendation strategy. Empirical evaluation of our method is reported in §5. Finally §6 concludes the paper.

2. RELATED WORKS

Recommendation in social media, e.g., user and community recommendation, has been well studied in previous research works. In this section, we highlight the ones that are most relevant to our work.

2.1 User Recommendation

User recommendation, often referred to as friendship recommendation or link prediction, focuses on recommending users to a target user based on diverse criteria. From a network perspective, user recommendation refers to finding missing edges in a user network. Typical approaches to solving this problem often utilize the network structure and node connections, e.g., proximity measures that are based on network topological features [8], supervised learning methods [1], relational learning methods [10], etc.

In social media, the content generated by users, e.g., user relationships or posts, is a valuable information source to model users' preference. Recently, several methods have been proposed to resolve user recommendation in social media by employing latent Dirichlet allocation (LDA) alike topic models [9]. These efforts, however, only consider interest similarity, and ignore the interactiveness of users, which is essential for expanding social network. In our work, we try to recommend users with influence abilities, given the fact that these users can help enrich the interactions among users. In addition, our model can distinguish users' personal interests from the topics discussed within communities.

2.2 Community Recommendation

Automated community discovery has been well studied by researchers. One direction in community discovery involves using the social linking structure among users to identify communities, e.g., min-cut based partitioning, centrality-based and Clique percolation methods [5, 11]. However, they did not take into account the content generated by users in social network, which might result in the irrationality of the identified communities. For example, two users in a community are reasonably connected through several links, but they may have no common topic interest at all.

Another direction in community discovery is to incorporate content analysis into the discovery process. Probabilistic models are often employed to capture the topics being discussed by users and within communities [13, 14, 15, 17], which assume all the content generated by a user will contribute to the community detection. In reality, however, an online user often posts his/her personal information, e.g., moods and activities, which might not be related to any community. Comparatively, our model distinguishes community-oriented topics from users' personal topics within the content, which is more reasonable in modeling the topic interests of users.

Given the detected communities, a further step for online community management is community recommendation. [4] proposes a collaborative filtering method for personalized community recommendation, by considering multiple types of co-occurrences in social data, e.g., semantic and user information. [3] uses association rule mining to discover associations between sets of communities that are shared across many users, and LDA [2] to model user-community co-occurrences via latent aspects. Both works performed experimental evaluation on Orkut data set.

3. USER-COMMUNITY-TOPIC MODEL

In this section, we first discuss two basic topic models used for tracking topic interests of online users or online communities. Based on the discussion, we propose User-Community-Topic model to resolve the issues in the two basic models. We then describe how to learn the hyper-parameters using Gibbs sampling.

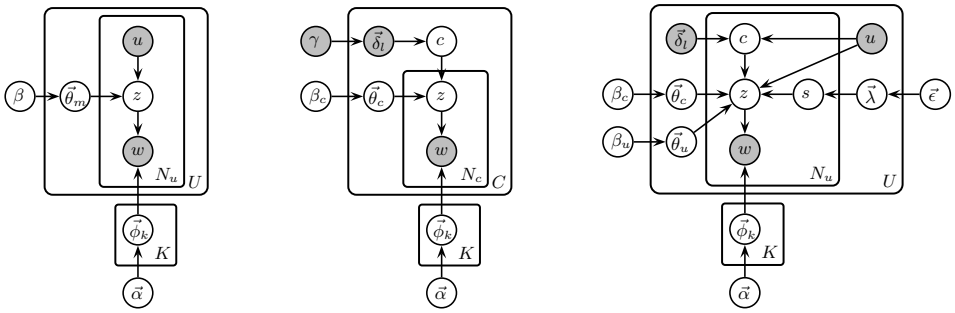
3.1 Discussion on Topic Models

Fig. 1(a) shows the graphical model for what we refer to as the "user-topic model" (UT). UT aims to capture the correlation between users and topics. The generation of a document (containing all the posts of a user) is considered as a mixture of topics. Each topic corresponds to a multinomial distribution over the vocabulary. Based on the learned posterior probability, each user's preference of using words and involvement in topics can be discovered. However, in most cases, users might have diverse interests over topics. By using UT model, the obtained posterior probability of a user over a specific topic might be affected by the general topic interests of this user. In addition, users in social media often share common interests over topics, which cannot be captured in UT model.

Another model is called "community-topic model" (CT) (as shown in Fig. 1(b)), where the generation of a document is affected by both the topic factors and the community factors in a hierarchical manner. In CT model, we treat all the posts within a community as a document. The difference from UT is the community factor c, by which topics within a document would be affected. One major problem of the CT model is that user posts in a community could include various topics, rendering the community document highly inconsistent. Sampling for all the words in a community document would result in uncontrolled generalization error for inference due to the noisy feature of social media data. In addition, there is no way to capture a specific user's interests using CT model, since no user factor is involved.

3.2 The Proposed Model

Our goal is to model the relations among users, topics and communities within the environment of social media. Taking Twitter as an example, we have the tweets posted by users and the follower-followee relations of users; however, we do not have the explicit community membership of users. We perform community discovery on the users' friendship network, and allow a user to belong to multiple communities by using soft clustering based methods. To achieve this, we employ the algorithm introduced in [16] to obtain the community memberships of users. We therefore assume there



(a) UT Model. (b) CT Model. (c) UCT Model.

Figure 1: Plate diagram for three topic models.

is a community factor c that captures the user-community memberships with respect to user u. Also, within each community, users might discuss different topics, and hence we have a topic factor z that characterizes the topic-community relations. For topic mixture and term mixture, we give them Dirichlet priors; for community mixture, we use the distribution derived by analyzing the follower-followee relations. In this way, we are not concerned with the relations between the community and the user, but focus more on the relations between the community and the topic (i.e., $p(z|c)$), and the relations between the user and the topic (i.e., $p(z|u)$). Table 1 lists the notations used in our model.

We denote our proposed topic model as "user-community-topic model" (UCT in Fig. 1(c)). We add a latent Bernoulli variable s (a binary factor) to indicate whether a word is related to a user itself or to a community. In particular, s takes value 0 if the word w is generated via the user-topic route, value 1 if the word is generated from the community-topic route. The variable s in our model acts as a switch: if $s = 0$, words are sampled from a user-specific multinomial $\vec{\theta}_u$, whereas if $s = 1$, words are sampled from a community-specific multinomial $\vec{\theta}_c$ (with different symmetric Dirichlet priors parameterized by β_u and β_c). s is sampled from a document-specific Bernoulli distribution $\vec{\lambda}$, which in turn has a prior ϵ. The joint probability of the UCT model can be written as:

$$\begin{aligned} &p(w, z, u, c, s, \phi_k, \theta_u, \theta_c, \delta_l, \lambda | \vec{\alpha}, \vec{\beta}_c, \vec{\beta}_u, \vec{\epsilon}) \\ &\quad = p(w|z, \phi_k) p(z|u, c, s, \theta_u, \theta_c) p(c|u, \delta_l) \\ &\quad \cdot p(s|\lambda) p(\lambda|\vec{\epsilon}) p(\phi_k|\vec{\alpha}) p(\theta_u|\vec{\beta}_u) p(\theta_c|\vec{\beta}_c), \end{aligned}$$

where $p(z|u, c, s, \theta_u, \theta_c) = p(z|u, s = 0, \theta_u)$ (where $s = 0$), and $p(z|u, c, s, \theta_u, \theta_c) = p(z|c, s = 1, \theta_c)$ (where $s = 0$). Here $p(z|u, s = 0, \theta_u)$ is the probability of a user-specific topic, whereas $p(z|c, s = 1, \theta_c)$ is the probability of a community-specific topic. Given the graphical model described in Fig. 1(c), the generative scheme is shown in Alg. 1.

3.2.1 Gibbs Updates

To estimate the model, we use the collapsed Gibbs sampling [6]. For our UCT model, we are interested in the latent user-topic portions $\vec{\theta}_u$, the latent community-topic portions $\vec{\theta}_c$, the topic-word distributions $\vec{\phi}_k$ and the topic index assignments for each word z_i. Also in the learning process, the value of s will be generated based on a Bernoulli distribution and be updated through the Gibbs sampling for each word. $\vec{\theta}_u$, $\vec{\theta}_c$ and $\vec{\phi}_k$ can be calculated using just the topic index assignments z_i, i.e., $\mathbf{z}$ is a sufficient statistic for the three distributions. Therefore, we can integrate out the multinomial parameters and simply sample z_i and s_i.

Table 1: Notations for quantities in the model.

	Descriptions
U	the user set in the community data.
V	the dictionary of texts in the community data.
L	the number of communities predefined.
N_u	the term set of texts posted by user u.
$\vec{\alpha}$	Dirichlet prior hyperparameter (known) on the term distribution.
$\vec{\beta}$	Dirichlet prior hyperparameter (known) on the mixture topic distribution.
$\vec{\gamma}$	Prior hyperparameter (known) on the mixture community distribution.
$\vec{\epsilon}$	Prior hyperparameter on the binary mixture.
$\vec{\phi}_k$	$p(t\|z = k)$, the mixture component of topic k.
$\vec{\theta}_m$	$p(z\|u = m)$, the topic mixture proportion for user m.
$\vec{\delta}_l$	$p(u\|c = l)$, the user proportion for community l. (observed)
$\vec{\lambda}$	binary mixture for word generation.
c	the community mixture.
u	mixture indicator that chooses a user from a community.
z	mixture indicator that chooses the topic for the term from a user.
w	term indicator for the word from a user.
s	binary factor for word generation.

Algorithm 1 Generative scheme of UCT model.

```
for each topic z ∈ (1, ··· , K) do
    Sample φ_k ~ Dir(·|α⃗)
end for
for each user u ∈ (1, ··· , U), do
    Sample λ_u ~ Beta(·|ε⃗)
    for each word w ∈ (1, ··· , N_u), do
        Sample s ~ Bern(·|λ⃗_u)
        Choose a community assignment c_u ~ Mult(·|δ⃗_l)
        if(s==0): then
            Choose a topic assignment z ~ Mult(·|θ⃗_u)
        else
            Choose a topic assignment z ~ Mult(·|θ⃗_c)
        end if
        Choose a term w ~ Mult(·|φ⃗_k, z)
    end for
end for
```

The collapsed Gibbs sampler needs to compute the probability of a topic z being assigned to a word w_i, given all other topic assignments to all other words, with respect to a specific value of s (0 or 1). Similarly, it needs to calculate the probability of s being assigned to a word w_i, given all other s assignments to all other words. Let $\mathbf{z}_{-i}$ denote all topic allocation except for z_i and $\mathbf{s}_{-i}$ represent

Table 2: Gibbs updates for UCT model.

$$p(s_i=1|\mathbf{s_{-i}},w,z,u,c) \propto \frac{p(s_i=1,\mathbf{s_{-i}},w,z,u,c)}{p(\mathbf{s_{-i}},w,z,u,c)} \propto p(s_i=1|z_i,c_i)=p(z_i|s_i=1,c_i)\cdot p(s_i=1|u_i) \propto \frac{n_{z_i,c_i,s_i=1}+\beta_c(z_i)}{\sum_{z_i} n_{z_i,c_i,s_i=1}+\sum_{z_i}\beta_c(z_i)-1}\cdot\frac{n_{s_i=1,u_i=u}+\epsilon_{s=1}}{\sum_{s_i} n_{s_i=1,u_i=u}+\epsilon_{s=0}+\epsilon_{s=1}-1}.$$

$$p(s_i=0|\mathbf{s_{-i}},w,z,u,c) \propto \frac{p(s_i=0,\mathbf{s_{-i}},w,z,u,c)}{p(\mathbf{s_{-i}},w,z,u,c)} \propto p(s_i=0|z_i,u_i)=p(z_i|s_i=0,u_i)\cdot p(s_i=0|u_i) \propto \frac{n_{z_i,u_i,s_i=0}+\beta_u(z_i)}{\sum_{z_i} n_{z_i,u_i,s_i=0}+\sum_{z_i}\beta_u(z_i)-1}\cdot\frac{n_{s_i=0,u_i=u}+\epsilon_{s=0}}{\sum_{s_i} n_{s_i=0,u_i=u}+\epsilon_{s=0}+\epsilon_{s=1}-1}.$$

$$p(z_i|\mathbf{z_{-i}},w,s_i=0,u,c) \propto \frac{p(\mathbf{z_i},w,s_i=0,u,c)}{p(\mathbf{z_{-i}},w,s_i=0,u,c)} \propto p(z_i,s_i=0,w_i,u_i,c_i)=p(w_i|z_i)\cdot p(z_i|s_i=0,u_i) \propto \frac{n_{w_i,z_i}+\alpha(w_i)}{\sum_V n_{w_i,z_i}+\sum_V\alpha(w_i)-1}\cdot\frac{n_{z_i,u_i,s_i=0}+\beta_u(z_i)}{\sum_{z_i} n_{z_i,u_i,s_i=0}+\sum_{z_i}\beta_u(z_i)-1}.$$

$$p(z_i|\mathbf{z_{-i}},w,s_i=1,u,c) \propto \frac{p(\mathbf{z_i},w,s_i=1,u,c)}{p(\mathbf{z_{-i}},w,s_i=1,u,c)} \propto p(z_i,s_i=1,w_i,u_i,c_i)=p(w_i|z_i)\cdot p(z_i|s_i=1,c_i) \propto \frac{n_{w_i,z_i}+\alpha(w_i)}{\sum_V n_{w_i,z_i}+\sum_V\alpha(w_i)-1}\cdot\frac{n_{z_i,c_i,s_i=1}+\beta_c(z_i)}{\sum_{z_i} n_{z_i,c_i,s_i=1}+\sum_{z_i}\beta_c(z_i)-1}.$$

all s assignments except for s_i. The probabilities that we need to update include: (1) $p(s_i=0|\mathbf{s_{-i}},w_i,z_i,u_i,c_i)$, (2) $p(s_i=1|\mathbf{s_{-i}},w_i,z_i,u_i,c_i)$, (3) $p(z_i|\mathbf{z_{-i}},w_i,s_i=0,u_i,c_i)$, and (4) $p(z_i|\mathbf{z_{-i}},w_i,s_i=1,u_i,c_i)$. The derivations of the updates for these probabilities are described in Table 2.

We analyze the computational complexity of Gibbs sampling in the proposed UCT model. As discussed above, in Gibbs sampling, we need to compute the posterior probability $p(z_i|\mathbf{z_{-i}},w_i,s_i,u_i,c_i)$ for user-word pairs $(U\times V)$ and community-word pairs $(C\times V)$, where V is the total number of words. Each $p(z_i|\mathbf{z_{-i}},w_i,s_i,u_i,c_i)$ consists of K topics, and requires a constant number of operations, resulting in $O(V\cdot K\cdot U)$, assuming $U>>C$, for a single sampling.

4. RECOMMENDATION STRATEGIES

In our work, we try to recommend a list of users with relevant topic interests and cohesive discussions. The target user can select some of the recommended users as friends, and then start to involve the discussion among these users. Our recommendation framework, `FRec`, provides various recommendation mechanisms based on our user-community-topic model. We also consider the user influence with respect to a topic. For each topic in the topic list, we can use the derived probabilities $p(u|z)$ as the initialization of the PageRank algorithm [7], and run PageRank on the friendship network to obtain the influence scores of users towards a specific topic z. Then the topic-relevant user influence can be denoted as $R(u|z)$. We setup a threshold (0.01) for $p(u|z)$ to filter out low probabilities.

4.1 User-to-User Recommendation

Given a target user $\hat{u}$, we can rank other users based on $p(u_i|\hat{u})$, and then select top ranked ones as $\hat{u}$'s recommendation. $p(u_i|\hat{u})$ can be calculated using Eq.(2).

$$\begin{aligned} p(u_i|\hat{u}) &= \frac{\sum_z\sum_c p(u_i\hat{u}cz)}{p(\hat{u})} \\ &\propto p(u_i)\sum_z\sum_c p(z|\hat{u})p(z|u_i,s=0)p(z|c,s=1)p(c|u_i)p(c|\hat{u})p(c) \\ &\propto p(u_i)\sum_z\left(p(z|\hat{u})p(z|u_i,s=0)\sum_c p(z|c,s=1)p(c|u_i)p(c|\hat{u})p(c)\right). \end{aligned} \quad (2)$$

Here $p(z|\hat{u})$ is the probability of topics given a test user $\hat{u}$, which can be obtained by extending Gibbs iterations over the test users after the hyper-parameters are learned. Note that in Eq.(2), we consider both user-based topics $(p(z|u_i,s=0))$ and community-based topics $(p(z|c,s=1))$. The user-based topics often include a user's personal interest. To make the recommendation more community-oriented, we can focus on community-based topics by removing the user-based component. The recommendation can be refined as

$$p(u_i|\hat{u}) \propto p(u_i)\sum_z\left(\frac{p(z|\hat{u})}{p(z)}\sum_c p(z|c,s=1)p(c|u_i)p(c|\hat{u})p(c)\right). \quad (3)$$

By integrating the user influence into $p(u_i|\hat{u})$, we can have

$$p(u_i|\hat{u}) \propto p(u_i)\cdot\sum_z\left(\frac{p(z|\hat{u})R(u_i|z)}{p(z)}\sum_c p(z|c,s=1)p(c|u_i)p(c|\hat{u})p(c)\right). \quad (4)$$

In this strategy, the user-to-user relations residing in the friendship network are not considered. In order to make the recommendation more reasonable, we incorporate the neighborhood similarity between u_i and the target user $\hat{u}$ into the recommendation. The neighborhood similarity can be calculated as

$$sim(u_i,\hat{u}) = \frac{|neighborhood(u_i)\cap neighborhood(\hat{u})|}{|neighborhood(u_i)\cup neighborhood(\hat{u})|},$$

where $neighborhood(\cdot)$ denotes all the neighbors of the user. By integrating $sim(u_i,\hat{u})$ into Eq.(4), we have

$$\begin{aligned} \tilde{p}(u_i|\hat{u}) &\propto p(u_i)\cdot sim(u_i,\hat{u}) \\ &\cdot\sum_z\left(\frac{p(z|\hat{u})R(u_i|z)}{p(z)}\sum_c p(z|c,s=1)p(c|u_i)p(c|\hat{u})p(c)\right). \end{aligned} \quad (5)$$

4.2 User-to-Community Recommendation

Given a target user $\hat{u}$, we can also recommend communities to $\hat{u}$ based on the derived correlations among users, topics and communities. Given a community c, we can measure the relevance between $\hat{u}$ and c by

$$\begin{aligned} p(c|\hat{u}) &= \frac{\sum_z p(c,\hat{u},z)}{p(\hat{u})} \propto \sum_z \frac{p(z|\hat{u},s=0)p(z|c,s=1)p(c)}{p(z)} \\ &\propto p(c)\sum_z \frac{p(z|\hat{u},s=0)p(z|c,s=1)}{p(z)}. \end{aligned} \quad (6)$$

A community with more influential users is likely to be more interactive, i.e., it may involve more activities of sharing information and discussing topics. Therefore, we consider the user influence for community recommendation. By integrating the user influence into $p(c|\hat{u})$, we have

$$\tilde{p}(c|\hat{u}) \propto p(c)\sum_z \frac{p(z|\hat{u},s=0)p(z|c,s=1)\cdot\left(\sum_{u_j\in c} R(u_j|z)\right)}{p(z)}. \quad (7)$$

5. EMPIRICAL EVALUATION

5.1 Real-World Data

The data set used in the experiment is a collection of tweets related to "presidential campaigns" between Barack Obama and Mitt Romney, ranging from March 1st, 2012 to May 31st, 2012. We crawled the tweets through Twitter Streaming API by feeding a list of keywords related to the campaign (e.g., campaign, Obama, Romney, economy, etc.) into the API request. We then crawled the follower relationships of each user within the tweets data set. Due to the property of microblogging services, the crawled tweets

might contain a lot of noise, which would hinder the topic modeling. Therefore, we did a series of preprocessing to alleviate the negative impact of noise data, including: (1) removing short tweets (with the word count less than 10); (2) removing tweets with hashtags more than 3; (3) removing tweets whose author has no more than 5 tweets; and (4) removing usernames (starting with "@") and URLs. After preprocessing, the tweets data contain 133,465 users, 5,558,763 mutual-following relationships and 5,079,994 tweets.

5.2 Comparison of Topic Models

For topic modeling, we concatenate the tweets of each user in the data set as a document. We process the tweets data by removing stopwords, tokenizing and stemming using MALLET. We also calculate the TF-IDF score of each word and then select the top ranked 10,000 words as features. After processing, the total number of word tokens in the tweets data is 6,643,278. We compare UCT model with two baselines: (1) CCF (Combinational Collaborative Filtering) [4], which combines the bag-of-users and bag-of-words models to capture the relations among topics, communities and users within the network; and (2) TUCM (Topic User Community Model) [13], which assumes that a user's membership in a community is conditioned on its social relationship, the type of interaction and the shared information with other members. We also include the models shown in Fig. 1(a) (UT) and Fig. 1(b) (CT) in the comparison.

For all the models, we empirically set the number of communities as 500. We set the hyper-parameters to the following values [12]: $\alpha = 0.01$, $\beta = \beta_u = \beta_c = 0.01$ and $\epsilon = 0.3$. We run 200 iterations of Gibbs sampling for training and extend the chain with 100 iterations over the test set.

5.2.1 Perplexity Comparison

We compare the predictive performance of our proposed UCT model with other baselines by computing the perplexity of unseen words in test documents. We calculate the averaged perplexity for 10-fold cross validation on the tweets data set. As is depicted in Fig. 2, the predictive performance

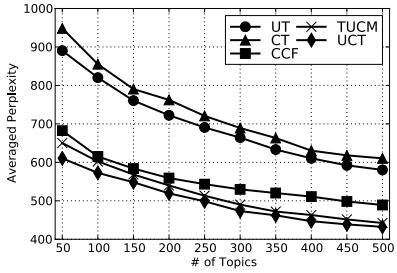
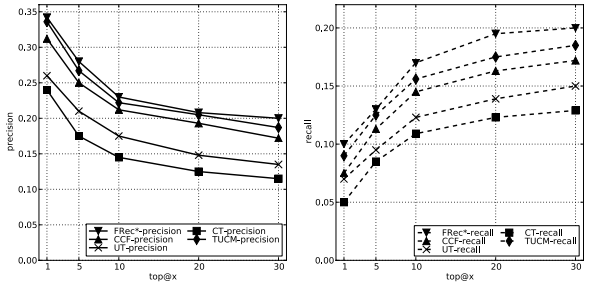


Figure 2: Perplexity evaluation.

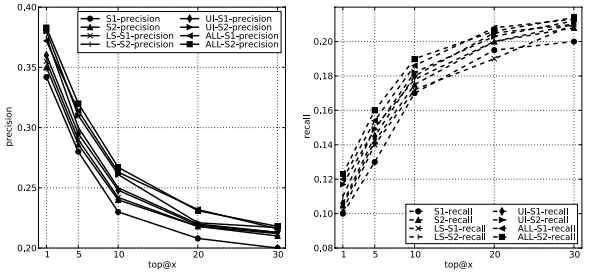
of two basic models (UT and CT) are not comparable with the other three topic models. The reason is straightforward: in both models, only one aspect (either u or c) is considered, which violates the characteristics of the data, since in social media, people post information not only for their own purpose, but also expecting to interact with each other. CCF combines the word factor and the user factor to capture the correlation between users and communities, and TUCM takes into account the type of interactions between users. These two models achieves better predictive performance compared with UT and CT. Our model distinguishes community-oriented topics and user-oriented topics. Such a distinction indeed exists in most real-world scenarios, i.e., a user has his/her personal topic interests, and is also often involved in the discussion within a specific community. In the recommendation experiments, we set the number of topics as 500 for all the models.

5.3 User Profile based Recommendation

To evaluate the user-profile based recommendation, we adopt the leave-one-out or leave-n-out strategy as described in [3]. *precision* and *recall* are used to measure the recommendation effectiveness. *Precision* is calculated at a given cut-off rank, considering only the topmost results recommended by the approach, e.g., top@10. We limit the size of our recommendation list to at most 30.



(a) Comparison of Topic Models.



(b) Effect of Different Components.

Figure 3: Comparison for user recommendation.

5.3.1 Recommending Users

We compare the user recommendation strategy introduced in §4.1 with several topic model based recommendation approaches: (1) UT: The recommendation can be achieved using the strategy similar to Eq.(2), by removing the components related to c; (2) CT: By considering the identified community membership, we can select a list of top ranked users, based on $p(u|c)$, from the community that the target user belongs to; (3) CCF: This method provides user recommendation by calculating the user similarity introduced in [4]; (4) TUCM: The recommendation can be achieved using the strategy similar to Eq.(2).

Our goal is to select a list of users whose topic interests are close to the target user. By removing the user-oriented components from Eq.(2), we can make the recommendation results more community oriented, as defined in Eq.(3). Note that in Eq.(3), we consider the community information of both the target user and the recommended user. To this end, we randomly select 2,000 users from the user repository as the test set and randomly delete a set of links of each test user: (1) $S1$: removing 20% links; and (2) $S2$: re-

moving 2% links. We conduct experiments based on these two setups. Fig. 3 shows the results for these users. For comparisons with topic models and link prediction methods, the experiments use setup $S1$; To evaluate the effect of different components in Eq.(5), we use setup $S1$ and $S2$.

From Fig. 3(a), we observe that our proposed framework FRec achieves the best recommendation performance in terms of *precision* and *recall*. Simply using topics (UT in Fig. 3(a)) cannot guarantee high-quality recommendation results. For example, two users might share similar interests but they do not have connections in the social graph.

In Fig. 3(b), we evaluate how user influence (UI) and users' local similarity (LS) affect the recommendation performance. We compare the basic model of FRec, the model with UI, the model with LS and the model with UI and LS for two different settings $S1$ and $S2$. Based on the comparison, we observer that: (1) User influence component and local similarity component slightly improved the performance of user recommendation. Intuitively, a user will prefer to make friends with influential people, since through these people he/she can reach more friends. Also, a user will be likely to interact with friends-of-friends. (2) The user recommendation has more accurate results if more social links of users are reserved. This is primarily because social links can help identify the underlying communities and then enrich the recommendation model through the user-community relations.

5.3.2 Recommending Communities

For community recommendation, we treat the communities identified from the module of community detection as the ground truth. We randomly sample 2,000 users from the user repository and recommend communities for these users. The comparison includes: (1) FRec: The basic strategy described in Eq.(6); (2) FRec-s1: removing the factor of $p(z|u, s = 0)$ from Eq.(6), i.e., only considering the community-oriented topics for recommendation; (3) FRec-IN: the strategy described in Eq.(7); and (4) FRec-IN-s1: removing the factor of $p(z|u, s = 0)$ from Eq.(7), i.e., considering user influence and the community-oriented topic factor. We also compare FRec with several recommendation approaches, including CCF and TUCM as introduced previously. These two approaches use the inferred probabilities of $p(z|u)$ and $p(z|c)$ for community recommendation. We report the comparison in Fig. 4.

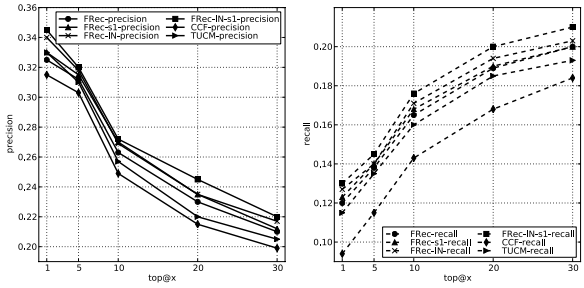


Figure 4: Community recommendation result.

As observed in Fig. 4, the model of FRec-IN-s1 has the best performance against other baselines, which explains that users in social media would like to interact with influential users, and prefer to share information that is often discussed within a community, i.e., by a group of people. The community-oriented topic factor $p(z|c, s = 1)$ has superior power over user-oriented topic factor $p(z|u, s = 0)$ in dominating the results of community recommendation.

6. CONCLUDING REMARKS

We have introduced a generative graphical model, User-Community-Topic model (UCT), for capturing user-oriented topics and community-oriented topics simultaneously in social media data. Based on the model inference, we further proposed a novel recommendation framework, FRec. Given a user's profile, FRec is able to recommend a list of topic-related influential users or a list of topic-cohesive interactive communities. The proposed framework can be easily extended to the case that recommends users and communities based on a set of keywords. In addition, it can be seamlessly integrated into real-life social networks.

ACKNOWLEDGEMENTS

The work is supported in part by a Xerox University Affair Committee (UAC) Award and by US National Science Foundation under grants DBI-0850203, CCF-0939179, HRD-0833093, CNS-1126619, and IIS-1213026.

7. REFERENCES

[1] M. Al Hasan, V. Chaoji, S. Salem, and M. Zaki. Link prediction using supervised learning. In *SDM'06: Workshop on Link Analysis, Counter-terrorism and Security*, 2006.

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

[3] W. Chen, J. Chu, J. Luan, H. Bai, Y. Wang, and E. Chang. Collaborative filtering for orkut communities: discovery of user latent behavior. In *Proc. of WWW*, pages 681–690. ACM, 2009.

[4] W. Chen, D. Zhang, and E. Chang. Combinational collaborative filtering for personalized community recommendation. In *Proc. of SIGKDD*, pages 115–123. ACM, 2008.

[5] S. Fortunato. Community detection in graphs. *Physics Reports*, 486(3):75–174, 2010.

[6] T. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences of the United States of America*, 101(Suppl 1):5228–5235, 2004.

[7] P. Lawrence, B. Sergey, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford University, 1998.

[8] D. Liben-Nowell and J. Kleinberg. The link-prediction problem for social networks. *Journal of the American society for information science and technology*, 58(7):1019–1031, 2007.

[9] M. Pennacchiotti and S. Gurumurthy. Investigating topic models for social media user recommendation. In *Proc. of WWW*, pages 101–102. ACM, 2011.

[10] A. Popescul and L. Ungar. Statistical relational learning for link prediction. In *IJCAI workshop on learning statistical models from relational data*, volume 2003, 2003.

[11] M. Porter, J. Onnela, and P. Mucha. Communities in networks. *Notices of the AMS*, 56(9):1082–1097, 2009.

[12] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. The author-topic model for authors and documents. In *Proc. of UAI*, pages 487–494, 2004.

[13] M. Sachan, D. Contractor, T. Faruquie, and L. Subramaniam. Using content and interactions for discovering communities in social networks. In *Proc. of WWW*, pages 331–340. ACM, 2012.

[14] Z. Xu, Y. Zhang, Y. Wu, and Q. Yang. Modeling user posting behavior on social media. In *Proc. of SIGIR*, pages 545–554. ACM, 2012.

[15] T. Yang, R. Jin, Y. Chi, and S. Zhu. Combining link and content for community detection: a discriminative approach. In *Proc. of SIGKDD*, pages 927–936. ACM, 2009.

[16] K. Yu, S. Yu, and V. Tresp. Soft clustering on graphs. *Advances in Neural Information Processing Systems*, 18:1553–1560, 2006.

[17] D. Zhou, E. Manavoglu, J. Li, C. Giles, and H. Zha. Probabilistic models for discovering e-communities. In *Proc. of WWW*, pages 173–182, 2006.