A Generalized Stochastic Block Model for Recommendation in Social Rating Networks

Mohsen Jamali
School of Computing Science
Simon Fraser University
Burnaby, BC, Canada
mohsen_jamali@cs.sfu.ca

Tianle Huang
School of Computing Science
Simon Fraser University
Burnaby, BC, Canada
tha30@cs.sfu.ca

Martin Ester
School of Computing Science
Simon Fraser University
Burnaby, BC, Canada
ester@cs.sfu.ca

ABSTRACT

The rapidly increasing availability of online social networks and the well-known effect of social influence have motivated research on social-network based recommenders. Social influence and selection together lead to the formation of communities of like-minded and well connected users. Exploiting the clustering of users and items is one of the most important approaches for model-based recommendation. Users may belong to multiple communities or groups, but only a few clustering algorithms allow clusters to overlap. One of these algorithms is the probabilistic EM clustering method, which assumes that data is generated from a mixture of Gaussian models. The mixed membership stochastic block model (MMB) transfers the idea of EM clustering from conventional, non-relational data to social network data. In this paper, we introduce a generalized stochastic blockmodel (GSBM) that models not only the social relations but also the rating behavior. This model learns the mixed group membership assignments for both users and items in an SRN. GSBM can predict the future behavior of users, both the rating of items and creation of links to other users. We performed experiments on two real life datasets from Epinions.com and Flixster.com, demonstrating the accuracy of the proposed GSBM for rating prediction as well as link prediction.

Categories and Subject Descriptors

H.2.8.d [**Information Technology and Systems**]: Database Applications - Data Mining

General Terms

Algorithms, Design, Experimentation, Measurement

Keywords

Social Recommendation, Clustering, Link Prediction, Stochastic Block Model

1. INTRODUCTION

With the rapidly growing amount of information available on the WWW, it becomes necessary to have tools to help users to select the relevant part of online information. To satisfy this need, recommender systems have emerged, e.g. there are popular recommenders for movies[1], books[2], music[3], etc.

Generally two types of recommenders have been investigated: Memory-based and Model-based. Memory based algorithms (collaborative filtering) [9] explore the user-item rating matrix and make recommendations based on the ratings of item i by a set of users whose rating profiles are most similar to that of user u. Model-based approaches [19][13] learn the parameters of a model and store only those parameters. Hence they do not need to explore the rating matrix at prediction time and are very fast after the parameters of the model are learnt. The bottleneck for model-based approaches is the training phase, while in memory-based approaches there is no training, but the prediction (test) phase is slower.

With the advent of online social networks, the social network based approach to recommendation has emerged. This approach assumes, on top of the user-item rating matrix, a social network among users and makes recommendations for a user based on the ratings of the users that have direct or indirect social relations with the given user. A social rating network (SRN) is a social network in which each user expresses ratings on some items besides creating social relations to other users [12].

Exploiting social networks in recommendation works because of the effects of selection and social influence that have been postulated by sociologists for a long time. Selection[16] means that people tend to relate to people with similar attributes. In SRNs, selection implies a tendency of users to form relationships with others who have already shown similar rating behavior. Another fundamental property of social networks is social influence [7], i.e. people tend to behave similarly to their friends. The effect of social influence in SRNs leads to people adopting the rating behavior of their friends, and hence becoming more similar[22]. The increasing availability of online social network data has finally allowed a verification of these sociological models. The results of experiments in [6] and of similar work confirm that a social network provides a source of information that is com-

[1] http://www.netflix.com
[2] http://www.amazon.com
[3] http://www.last.fm

plementary to item ratings and can be exploited to improve the quality of recommendations.

Social influence and selection together lead to the formation of communities of like-minded and well-connected users. Social influence can produce network-wide uniformity, as a new behavior spreads across the links. Selection, on the other hand, tends to drive the network toward smaller clusters of like-minded individuals [10]. The tendency of people to come together and form groups is inherent in the structure of society [2], and studying the ways in which such groups take shape and evolve over time is a theme that runs through large parts of social science research [5]. Users may belong to multiple groups, i.e., when a social actor interacts with different partners, different social contexts may apply and thus the actor may be acting according to the different roles they can possibly play [1]. For example, when a university professor interacts with his student on a social networking website, he belongs to the professors' group, but when interacting with his son on the same social networking website, he will play the role of a father rather than a university professor. Similarly, in a social rating network, the context or group of an item could be different when different users from different groups rate it. For example, a digital camera may belong to the group of advanced cameras when being rated by a professional photographer, and the same camera can be considered in the group of expensive cameras for an amateur photographer.

Exploiting the clustering of users is one of the most important approaches for model-based recommendation [21]. To make a recommendation for a user, a clustering-based approach aggregates the ratings in the cluster of that user. However, only a few clustering algorithms allow clusters to overlap, which is crucial for communities in social rating networks. One of these methods, the probabilistic EM clustering method [3], assumes that data has been generated from a mixture of Gaussian models and learns for every data point a distribution of membership over the groups. Mixture models are not immediately applicable to relational data such as social networks because they assume that the objects are conditionally independent given their cluster assignments [1]. The stochastic blockmodel [20] is an adaptation of mixture modeling to relational data. In that model, each object belongs to one cluster and the relationships between objects are governed by their corresponding pair of clusters. The stochastic blockmodel suffers from the limitation that each object can only belong to one cluster, or in other words, can play a single latent role. The mixed membership stochastic blockmodel (MMB) [1] relaxes the assumption of single-latent role for actors and learns a membership distribution over different groups. Every user is associated with a probability distribution over the groups, i.e. he belongs to each group with a different degree of membership. Mixed membership models, such as latent Dirichlet allocation [4], have re-emerged in recent years as a flexible modeling tool for data where the single cluster assumption is violated [1].

In this paper, we introduce a generalized stochastic blockmodel (GSBM) that models not only the social relations but also the rating behavior. The proposed model learns the mixed group membership assignments for both users and items in an SRN. GSBM fills the gap between cluster-based models and social network based approaches for recommendation. GSBM is capable of predicting both types of user behavior, rating of items and the creation of links to other users.

Basically, GSBM is a generative model that captures the behavior of users in a social rating network, including the creation of social relations and the ratings of items. Every user acts according to different groups he belongs to in each of his actions, whether he is creating a social relation or is rating an item. The items that are being rated by users also belong to different groups based on the users and the context in which they are being rated. The proposed GSBM ia a probabilistic method for finding communities in a social rating network taking into account both types of users behaviors. Every user and every item has a mixed membership assignment latent vector. In every action (creating a social relation, or rating an item), the user is probabilistically considered to be acting as a member of one of the groups. Also every item is considered to belong to a latent group when it is being rated.

The rest of this paper is organized as follows:In section 2, some related work are discussed. The generalized stochastic block model is proposed in section 3. The variational inference method for learning the model parameters is discussed in section 4. Implementation details for GSBM are discussed in section 5. Section 6 presents the design of experiments and the experimental results on two real life data sets. Finally, we conclude the paper and present some directions for future work in section 7.

2. RELATED WORK

In this section we review some related work on recommendation in social networks, both memory based and model based approaches. Then we review clustering based approaches for recommendation. Finally, we briefly discuss the mixed membership stochastic block model (MMB) that is the basis for our proposed model.

TidalTrust[8] performs a modified breadth first search in the trust(social) network to compute a prediction. Basically, it finds raters of the item with the shortest path distance from the source user and aggregates their ratings weighted by the trust between the source user and these raters. To compute the trust value between user u and v who are not directly connected, TidalTrust aggregates the trust value between u's direct neighbors and v weighted by the direct trust values of u and its direct neighbors.

[15] introduces MoleTrust. The ideas used in MoleTrust and TidalTrust are similar, but MoleTrust considers all raters up to a *maximum-depth* given as an input. *maximum-depth* is independent of any specific user and item. Also, to compute the trust value between u and v in MoleTrust, a backward exploration is performed, i.e. the trust value from u to v is the aggregation of trust values between u and users directly trusting v weighted by the direct trust values.

In order to consider enough ratings without suffering from noisy data, [11] proposes a random walk method (TrustWalker) which combines trust-based and item-based recommendation. TrustWalker considers not only ratings of the target item, but also those of similar items. The probability of using the rating of a similar item instead of a rating for the target item increases with increasing length of the walk. This framework contains both trust-based and item-based collaborative filtering recommendations as special cases. The experiments show that their method outperforms existing memory based approaches.

SocialMF [12] is a matrix factorization [13][19] based approach for recommendation in social networks. This work, along with some other works [14] are the first works on model based approaches for recommendation in social networks. In SocialMF, the rating matrix is decomposed into a product of latent feature vectors of users and items. To take the social network into account, SocialMF considers the latent feature of a user to be close to the aggregate of the latent features of his direct neighbors. SocialMF also supports trust propagation between indirect neighbors in the social network.

Since GSBM is a clustering-based model for social rating networks, we briefly review the clustering-based approaches for recommendation. EM clustering has already been discussed in the introduction. A standard model-based recommendation algorithm uses k-means to cluster similar users [21]. Given a set of user rating profiles, the space is partitioned into k groups of users that are close to each other based on a measure of similarity. The discovered user clusters are then applied to the user- based neighborhood formation task, rather than individual profiles as used in the user-based collaborative filtering. To make a recommendation for a user u and target item i, a neighborhood of users belonging to the same group as the user u who also have rated i are considered and their ratings on item i are aggregated to compute the prediction.

The mixed membership stochastic blockmodel (MMB) [1] is a generative model to capture the behavior of users when creating social relations. Every user is associated with a probability distribution over the groups. For every user pair (u, v), the group assignments of u and v are sampled from the probability distribution associated to them. The probabilities of interaction among different groups are defined by a matrix of Bernoulli rates. The link creation from u to v is then sampled from the Bernoulli distribution corresponding to the interaction probability of the groups assigned to the users. MMB only models the social relations and does not model the rating behavior of users. We extend MMB to also capture the rating behavior of users.

3. THE GENERALIZED STOCHASTIC BLOCKMODEL

The effects of social influence and selection in an SRN lead to the formation of groups of users and items. EM clustering methods assume overlapping group assignments among users based on a mixture of Gaussian models, but they cluster the users based only on their attributes and do not consider the social relations. On the other hand, mixed membership stochastic blockmodels (MMB) offer a mixed membership model for relational data, considering only the relational data but not the attributes of users. In this section, we extend MMB and propose a generalized stochastic block model (GSBM) to capture the groups of users and items in SRNs.

In this paper, we assume that a social rating network is given. The set of users in the SRN is denoted by $\mathcal{U}$ where $|\mathcal{U}| = N$ and the set of items is denoted by $\mathcal{I}$ where $|\mathcal{I}| = M$.

The social relations are considered to be directed in this paper, and $T_{u,v}$ is the binary random variable indicating whether there is a social relation from user u to user v. Rating values are assumed to be integer numbers in the range [1,5]. Hence the rating of a user u on an item i can be represented by an indicator vector $\overrightarrow{R}_{u,i}$ in which all elements are set to zero except for the one corresponding to the rating value.

3.1 The proposed GSBM

GSBM assumes K_1 groups among users and K_2 groups among items. Every user u is associated with a latent group assignment vector $\overrightarrow{\Pi}_u$, where $\Pi_{u,t}$ denotes the probability of user u belonging to group t. Similarly, every item i is associated with a latent group membership vector $\overrightarrow{\Delta}_i$. For every user u, the indicator vector $\overrightarrow{z}_{u\to v}$ denotes the group membership of user u when interacting with user v. Also $\overrightarrow{z}_{u\leftarrow v}$ denotes the group membership of user v when being interacted by user u. The probabilities of interactions among different groups of users are defined by a matrix of Bernoulli rates $B_{T_{(K_1\times K_1)}}$, where $B_{T_{l,t}}$ represents the probability of having a social relation between a user from group l and a user from group t.

Indicator vector $\overrightarrow{x}_{u\to i}$ denotes the group membership of user u while rating item i, and $\overrightarrow{x}_{u\leftarrow i}$ denotes the group membership of item i when being rated by user u. These indicator vectors are sampled from the latent group membership vectors $\overrightarrow{\Pi}_u$ and $\overrightarrow{\Delta}_i$. The multinomial distribution of the rating expressed by a user from a particular user group on an item from a specific item group is defined by a matrix $B_{R_{(K_1\times K_2)}}$, where $B_{R_{l,t}}$ is a probability vector representing the probabilities of the multinomial distribution for the rating expressed by a user in group l on an item in group t.

The proposed GSBM assumes the following generative model for social relations and ratings:

- For each user $u \in \mathcal{U}$:
  - Draw a K_1 dimensional mixed membership vector, $\overrightarrow{\Pi}_u \sim Dirichlet(\overrightarrow{\alpha})$
- For each item $i \in \mathcal{I}$:
  - Draw a K_2 dimensional mixed membership vector, $\overrightarrow{\Delta}_i \sim Dirichlet(\overrightarrow{\beta})$
- For each pair of users $(u, v) \in \mathcal{U} \times \mathcal{U}$:
  - Draw membership indicator for the initiator, $\overrightarrow{z}_{u\to v} \sim Multinomial(\overrightarrow{\Pi}_u)$
  - Draw membership indicator for the receiver, $\overrightarrow{z}_{u\leftarrow v} \sim Multinomial(\overrightarrow{\Pi}_v)$
  - Sample the social relation, $T_{u,v} \sim Bernoulli(\overrightarrow{z}^{\top}_{u\to v} B_T \overrightarrow{z}_{u\leftarrow v})$
- For each pair of a user and an item $(u, i) \in \mathcal{U} \times \mathcal{I}$:
  - Draw membership indicator for the user, $\overrightarrow{x}_{u\to i} \sim Multinomial(\overrightarrow{\Pi}_u)$
  - Draw membership indicator for the item, $\overrightarrow{x}_{u\leftarrow i} \sim Multinomial(\overrightarrow{\Delta}_i)$
  - Sample the rating value, $\overrightarrow{R}_{u,i} \sim Multinomial(\overrightarrow{x}^{\top}_{u\to i} B_R \overrightarrow{x}_{u\leftarrow i})$

3.2 Modeling the Sparsity

It is useful to distinguish two sources of non-interaction among users: they may be the results of rarity of interactions in general, or they may be an indication that the pair of relevant groups rarely interact. Thus, the authors of [1] introduced a sparsity parameter $\rho_T \in [0, 1]$ to characterize the source of non-interaction. Instead of sampling $T_{u,v}$ directly from Bernoulli with aforementioned parameter, they down-weight the probability of successful social interaction to $(1 - \rho_T)\overrightarrow{z}^{\top}_{u\to v} B_T \overrightarrow{z}_{u\leftarrow v}$.

For ratings, we introduce two levels of sparsity: A general sparsity parameter ρ_R, and a group level sparsity matrix B_Y whose elements indicate the rarity of ratings by a group of users on a group of items. In this case the process of generating a rating would be as follows:

For every pair of a user and item $(u, i) \in \mathcal{U} \times \mathcal{I}$:

- Draw membership indicator for the initiator, $\vec{x}_{u\rightarrow i} \sim Multinomial(\vec{\Pi}_u)$
- Draw membership indicator for the receiver, $\vec{x}_{u\leftarrow i} \sim Multinomial(\vec{\Delta}_i)$
- Sample the rating decision maker, $Y_{u,i} \sim Bernoulli((1-\rho_R)\vec{x}_{u\rightarrow i} B_Y \vec{x}_{u\leftarrow i})$. Here, B_Y is a $K_1 \times K_2$ matrix indicating the rating probabilities among different groups of users and items. ρ_R is the general rating sparsity parameter. The following step will be skipped if $Y_{u,i} = 0$.
- Sample the rating value, $\vec{R}_{u,i} \sim Multinomial(\vec{x}^{\top}_{u\rightarrow i} B_R \vec{x}_{u\leftarrow i})$

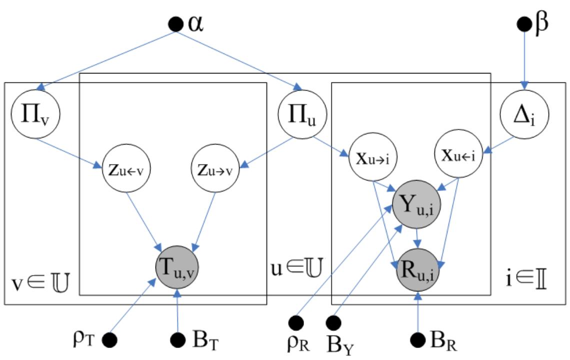


Figure 1: The graphical model underlying GSBM.

Figure 1 illustrates the graphical model underlying the proposed GSBM. Note that the random variable $T_{u,v}$ is a binary variable and is observed for all user pairs (u, v). If the social relation does not exist, then $T_{u,v} = 0$. Also, it should be noted that $Y_{u,i}$ is as an observed binary random variable defined for all pairs (u, i). Basically, $Y_{u,i}$ is the binary version of the observed ratings. If a rating $R_{u,i}$ does not exist, then $Y_{u,i} = 0$, i.e. it is observed as zero.

4. POSTERIOR INFERENCE AND PARAMETER ESTIMATION

As proposed in [1] for the MMB, we use variational-EM to learn the parameters of the GSBM model. The E-step is a variational inference on the latent variables. After learning the variational parameters for the latent variables, we perform the M-step similar to the conventional EM algorithm.

4.1 Variational Inference

The likelihood of the set of observed social relations T is computed as follows:

$$p(T, Z_\rightarrow, Z_\leftarrow | B_T, \vec{\Pi}_{1:N}, \rho_T) = \prod_u \prod_v P(T_{u,v} | \vec{z}_{u\rightarrow v}, \vec{z}_{u\leftarrow v}, B_T, \rho_T) P(\vec{z}_{u\rightarrow v} | \vec{\Pi}_u) P(\vec{z}_{u\leftarrow v} | \vec{\Pi}_v) \tag{1}$$

The likelihood of the set of observed ratings R is computed as following:

$$p(R, Y, X_\rightarrow, X_\leftarrow | B_R, B_Y, \vec{\Pi}_{1:N}, \vec{\Delta}_{1:M}, \rho_R) = \prod_u \prod_i p(\vec{R}_{u,i} | \vec{x}_{u\rightarrow i}, \vec{x}_{u\leftarrow i}, B_R, Y_{u,i}) p(Y_{u,i} | \vec{x}_{u\rightarrow i}, \vec{x}_{u\leftarrow i}, B_Y, \rho_R) \times p(\vec{x}_{u\rightarrow i} | \vec{\Pi}_u) p(\vec{x}_{u\leftarrow i} | \vec{\Delta}_i) \tag{2}$$

Now, the likelihood of the complete observed data is as follows:

$$\begin{aligned} p(T, R, Y, Z_\rightarrow, Z_\leftarrow, X_\rightarrow, X_\leftarrow, \vec{\Pi}_{1:N}, \vec{\Delta}_{1:M} \\ | \alpha, \beta, B_T, B_R, B_Y, \rho_T, \rho_R) = \\ p(T, Z_\rightarrow, Z_\leftarrow | B_T, \vec{\Pi}_{1:N}, \rho_T) \\ p(R, Y, X_\rightarrow, X_\leftarrow | B_R, B_Y \vec{\Pi}_{1:N}, \vec{\Delta}_{1:M}, \rho_R) \\ \times \prod_u p(\vec{\Pi}_u | \vec{\alpha}) \prod_i p(\vec{\Delta}_i | \vec{\beta}) \end{aligned} \tag{3}$$

To learn the values for the latent variables we need to compute the posterior distribution of the latent variables given the observed data. The normalization constant of this posterior distribution is the marginal probability of the data which requires integral over all latent variables. This integral is not solvable in closed form. We use variational inference to approximate this. The main idea behind variational inference is to assume a distribution of the latent variables with free parameters, and then fit those parameters such that the distribution is close in KL-divergence to the true posterior. The variational distribution is simpler than the true posterior since it approximates the posterior by assuming independence among different latent variables.

The log of the marginal probability is bounded using Jensen's inequality[4] as follows:

$$\begin{aligned} \log p(T, R, Y, \alpha, \beta, B_T, B_R, B_Y, \rho_T, \rho_R) \geq \\ \mathbb{E}_q \big[\log p(T, R, Y, Z_\rightarrow, Z_\leftarrow, X_\rightarrow, X_\leftarrow, \vec{\Pi}_{1:N}, \vec{\Delta}_{1:M} \\ | \alpha, \beta, B_T, B_R, B_Y, \rho_T, \rho_R) \big] \\ - \mathbb{E}_q \big[\log q(Z_\rightarrow, Z_\leftarrow, X_\rightarrow, X_\leftarrow, \vec{\Pi}_{1:N}, \vec{\Delta}_{1:M}) \big] \end{aligned} \tag{4}$$

The right hand side of equation 4 is a lower bound for the log likelihood and we denote it by $\mathcal{L}$ in the rest of this paper.

We introduce a distribution q of the latent variables depending on a set of free variational parameters γ, δ, ϕ, and ω. We specify q in a fully-factorized way as follows:

[4]http://en.wikipedia.org/wiki/Jensen's_inequality

$$
\begin{aligned}
&q(\overrightarrow{\Pi}_{1:N}, \overrightarrow{\Delta}_{1:M}, Z_{\rightarrow}, Z_{\leftarrow}, X_{\rightarrow}, X_{\leftarrow} \\
&\qquad |\overrightarrow{\gamma}_{1:N}, \overrightarrow{\delta}_{1:M}, \Phi_{\rightarrow}, \Phi_{\leftarrow}, \Omega_{\rightarrow}, \Omega_{\leftarrow}) = \\
&\prod_u q_1(\overrightarrow{\Pi}_u|\overrightarrow{\gamma}_u) \prod_u \prod_v q_2(\overrightarrow{z}_{u\rightarrow v}|\overrightarrow{\phi}_{u\rightarrow v}) q_3(\overrightarrow{z}_{u\leftarrow v}|\overrightarrow{\phi}_{u\leftarrow v}) \\
&\prod_i q_4(\overrightarrow{\Delta}_i|\overrightarrow{\delta}_i) \prod_u \prod_i q_5(\overrightarrow{x}_{u\rightarrow i}|\overrightarrow{\omega}_{u\rightarrow i}) q_6(\overrightarrow{x}_{u\leftarrow i}|\overrightarrow{\omega}_{u\leftarrow i})
\end{aligned} \tag{5}
$$

In the above equation, q_1 and q_4 are Dirichlet distributions (similar to the distribution in the generative model). q_2, q_3, q_5 and q_6 are multinomial distributions with variational parameters. To optimize the variational parameters, we compute the derivatives of the lower bound for the likelihood with respect to all variational parameters and set them to zero. The learnt variational parameters are as follows[5]:

$$
\begin{aligned}
&\phi_{u\rightarrow v_l} \propto \exp\big(\psi(\gamma_{u,l}) - \psi(\sum_{t=1}^{K_1} \gamma_{u,t})\big) \times \\
&\prod_{t=1}^{K_1} \big[((1-\rho_T)B_{T_{l,t}})^{T_{u,v}\phi_{u\leftarrow v_t}} (1-(1-\rho_T)B_{T_{l,t}})^{(1-T_{u,v})\phi_{u\leftarrow v_t}}\big]
\end{aligned} \tag{6}
$$

$$
\begin{aligned}
&\phi_{u\leftarrow v_l} \propto \exp\big(\psi(\gamma_{v,l}) - \psi(\sum_{t=1}^{K_1} \gamma_{v,t})\big) \times \\
&\prod_{t=1}^{K_1} \big[((1-\rho_T)B_{T_{l,t}})^{T_{u,v}\phi_{u\rightarrow v_t}} (1-(1-\rho_T)B_{T_{l,t}})^{(1-T_{u,v})\phi_{u\rightarrow v_t}}\big]
\end{aligned} \tag{7}
$$

In the above equation, ψ is the digamma function. Also for ω values, we have:

$$
\begin{aligned}
&\omega_{u\rightarrow i_l} \propto \exp\big(\psi(\gamma_{u,l}) - \psi(\sum_{t=1}^{K_1} \gamma_{u,t}\big) \\
&\qquad \times \Big(\prod_{n=1}^{K_2} \big[\prod_{t=1}^{5} B_{R_{l,n,t}}{}^{R_{u,i,t}}\big]^{\omega_{u\leftarrow i_n}}\Big)^{Y_{u,i}} \times \\
&\prod_{n=1}^{K_2} \big[((1-\rho_R)B_{Y_{l,n}})^{Y_{u,i}\omega_{u\leftarrow i_n}} (1-(1-\rho_R)B_{Y_{l,n}})^{(1-Y_{u,i})\omega_{u\leftarrow i_n}}\big]
\end{aligned} \tag{8}
$$

$$
\begin{aligned}
&\omega_{u\leftarrow i_l} \propto \exp\big(\psi(\delta_{i,l}) - \psi(\sum_{t=1}^{K_2} \delta_{i,t}\big) \\
&\qquad \times \Big(\prod_{m=1}^{K_1} \big[\prod_{t=1}^{5} B_{R_{m,l,t}}{}^{R_{u,i,t}}\big]^{\omega_{u\rightarrow i_m}}\Big)^{Y_{u,i}} \\
&\qquad \times \prod_{m=1}^{K_1} \big[((1-\rho_R)B_{Y_{l,m}})^{Y_{u,i}\omega_{u\rightarrow i_m}} \\
&\qquad\qquad (1-(1-\rho_R)B_{Y_{l,m}})^{(1-Y_{u,i})\omega_{u\rightarrow i_m}}\big]
\end{aligned} \tag{9}
$$

The learnt variational parameters for group memberships are as follows:

$$\gamma_{u,l} = \alpha_l + \sum_v \phi_{u\rightarrow v_l} + \sum_v \phi_{v\leftarrow u_l} + \sum_i \omega_{u\rightarrow i_l} \tag{10}$$

$$\delta_{i,l} = \beta_l + \sum_u \omega_{u\leftarrow i_l} \tag{11}$$

[5]For details of the computation of the lower bound and its derivatives, please refer to the detailed appendix available at http://www.cs.sfu.ca/∼sja25/personal/resources/SBM.pdf

Algorithm 1 Variational Inference for Parameters

Input: B_T, B_R, $\overrightarrow{\alpha}$, $\overrightarrow{\beta}$, ρ_T, ρ_R
initialize $\forall u, l : \gamma_{u,l} = \frac{2N}{K_1}$ and $\forall i, s : \delta_{i,s} = \frac{2M}{K_2}$
repeat
  for $u = 1$ to N **do**
    for $v = 1$ to N **do**
      using algorithm 2, compute variational $\phi^{t+1}_{u\rightarrow v}$, $\phi^{t+1}_{u\leftarrow v}$.
      partially update γ^{t+1}_u, γ^{t+1}_v and B^{t+1}_T
    end for
    for $i = 1$ to M **do**
      using algorithm 3, compute variational $\omega^{t+1}_{u\rightarrow i}$, $\omega^{t+1}_{u\leftarrow i}$.
      partially update γ^{t+1}_u, δ^{t+1}_i, B^{t+1}_Y and B^{t+1}_R
    end for
  end for
until convergence

4.2 Parameter Estimation

We compute the empirical Bayes estimates of the model hyper parameters $\{\overrightarrow{\alpha}, \overrightarrow{\beta}, B_T, B_R, \rho_T, \rho_R\}$ in the M-step of the variational expectation-maximization (EM) algorithm.

A closed form solution for the approximate maximum likelihood estimate of $\overrightarrow{\alpha}$ and $\overrightarrow{\beta}$ does not exist [1]. We use a linear time Newton-Raphson method. The approximate maximum likelihood estimator of B_T, B_Y and B_R are:

$$B_{T_{i,j}} = \frac{\sum_u \sum_v T_{u,v}\phi_{u\rightarrow v_i}\phi_{u\leftarrow v_j}}{(1-\rho_T)\sum_u \sum_v \phi_{u\rightarrow v_i}\phi_{u\leftarrow v_j}} \tag{12}$$

$$B_{Y_{m,n}} = \frac{\sum_u \sum_i Y_{u,i}\omega_{u\rightarrow i_m}\omega_{u\leftarrow i_n}}{(1-\rho_R)\sum_u \sum_i \omega_{u\rightarrow i_m}\omega_{u\leftarrow i_n}} \tag{13}$$

$$B_{R_{m,n,l}} = \frac{\sum_u \sum_i Y_{u,i}(\omega_{u\rightarrow i_m}\omega_{u\leftarrow i_n}R_{u,i,l})}{\sum_u \sum_i Y_{u,i}(\omega_{u\rightarrow i_m}\omega_{u\leftarrow i_n})} \tag{14}$$

Also, for the sparsity parameter, we have:

$$1-\rho_T = \frac{\sum_u \sum_v \sum_i \sum_j \phi_{u\rightarrow v_i}\phi_{u\leftarrow v_j}T_{u,v}}{\sum_u \sum_v \sum_i \sum_j \phi_{u\rightarrow v_i}\phi_{u\leftarrow v_j}B_{T_{i,j}}} \tag{15}$$

$$1-\rho_R = \frac{\sum_u \sum_i \sum_m \sum_n \omega_{u\rightarrow i_m}\omega_{u\leftarrow i_n}Y_{u,i}}{\sum_u \sum_i \sum_m \sum_n \omega_{u\rightarrow i_m}\omega_{u\leftarrow i_n}B_{Y_{m,n}}} \tag{16}$$

5. IMPLEMENTATION

The variational-EM algorithm requires a lot of memory and needs to store $NK_1 + 2N^2K_1 + MK_2 + NMK_1 + NMK_2$ variational parameters for the latent variables which is $O(N^2K_1 + NMK_1 + NMK_2)$. To reduce the space complexity, we use the nested variational algorithm introduced in [1]. In each variational cycle, the nested variational algorithm (presented in algorithm 1) only needs to store $NK_1 + 2K_1 + NK_2 + MK_2 + K_1 + K_2$ parameters which is $O(NK_1 + NK_2 + MK_2)$.

Algorithm 1 presents the nested variational inference method for the GSBM model. In this algorithm, γ_u, γ_v, δ_i, the interaction probability matrices B_T, B_y and B_R are partially updated in each iteration. The main idea behind nested variational inference is the scheduling of updates for variational parameters. In the

Algorithm 2 Variational Computation for $\phi_{u\rightarrow v}, \phi_{u\leftarrow v}$

1: **Input:** $\overrightarrow{\gamma_u}, \overrightarrow{\gamma_v}$,
2: initialize $\forall l : \phi^0_{u\rightarrow v_l} = \phi^0_{u\leftarrow v_l} = \frac{1}{K_1}$.
3: **repeat**
4:   **for** $l = 1$ to K_1 **do**
5:     update $\phi^{s+1}_{u\rightarrow v} \propto f_1(\phi^s_{u\leftarrow v}, \overrightarrow{\gamma_u}, B_T)$
6:   **end for**
7:   normalize $\phi^{s+1}_{u\rightarrow v}$ to sum to 1
8:   **for** $l = 1$ to K_1 **do**
9:     update $\phi^{s+1}_{u\leftarrow v} \propto f_2(\phi^s_{u\rightarrow v}, \overrightarrow{\gamma_v}, B_T)$
10:   **end for**
11:   normalize $\phi^{s+1}_{u\leftarrow v}$ to sum to 1
12: **until** convergence

Algorithm 3 Variational Computation for $\omega_{u\rightarrow i}, \omega_{u\leftarrow i}$

1: **Input:** $\overrightarrow{\gamma_u}, \overrightarrow{\delta_i}$,
2: initialize $\forall l : \omega^0_{u\rightarrow v_l} = \frac{1}{K_1}, \forall m : \omega^0_{u\leftarrow i_m} = \frac{1}{K_2}$.
3: **repeat**
4:   **for** $l = 1$ to K_1 **do**
5:     update $\omega^{s+1}_{u\rightarrow i} \propto f_3(\omega^s_{u\leftarrow i}, \overrightarrow{\gamma_u}, B_R)$
6:   **end for**
7:   normalize $\omega^{s+1}_{u\rightarrow i}$ to sum to 1
8:   **for** $m = 1$ to K_2 **do**
9:     update $\omega^{s+1}_{u\leftarrow i} \propto f_4(\omega^s_{u\rightarrow i}, \overrightarrow{\delta_i}, B_R)$
10:   **end for**
11:   normalize $\omega^{s+1}_{u\leftarrow i}$ to sum to 1
12: **until** convergence

nested variational algorithm, variational parameters γ and δ and the interaction probability matrices B_T, B_y and B_R are partially updated in the E-step which leads to a great reduction in space requirement of the algorithm.

Note that functions f_1 and f_2 in lines 5 and 7 of algorithm 2 correspond to equations 7 and 8 respectively. Also, functions f_3 and f_4 in lines 5 and 7 of algorithm 2 correspond to equations 9 and 10 respectively. While being much more space efficient, the nested variational inference leads to increased runtime due to the partial updates, However, algorithm 1 is easily parallizable on a shared memory architecture. It should be noted that variational parameters γ and δ, and the interaction probability matrices B_T, B_Y and B_R are shared by all the processors. The main *for* loop in algorithm 1 (lines 4-13) can be run in parallel since different iterations are independent. Therefore, GSBM gains linear speedup when it runs in parallel. We used *Intel Cilk Plus*[6] to implement the parallel version of our algorithm. Intel Cilk Plus is an extension to C and C++ that offers a quick, easy and reliable way to improve the performance of programs on multi-core processors. We used Cilk to implement the nested variational method presented in algorithm 1 on a 20 core server using 16 GB of the available main memory.

6. EXPERIMENTS

In this section, we present our experimental studies on two real life data sets from Epinions and Flixster. The detailed description and some statistics of these data sets are presented in the following subsection. We split the data set into 80% training and 20% test data, splitting both the set of social relations and the set of ratings expressed by users. We learn a GSBM using the training data. As discussed before, GSBM has two major applications: prediction of user behavior, and discovery of communities. We perform experiments to evaluate the accuracy of GSBM in predicting user behavior, in terms of both rating prediction and link prediction. We make predictions for withheld links/ratings by applying the trained model and compare the predictions to the withheld ground truth. Since there is no ground truth for communities in the social rating networks, we cannot directly evaluate the quality of community discovery. However, since prediction of user behavior is based on the community inference made by GSBM, the quality of predicting user behavior indirectly indicates the quality of community discovery performed by GSBM.

6.1 Data Sets

In this subsection we briefly introduce the data sets used in our experiments: the Flixster[7] data set, and the Epinions[8] data set.

The Flixster data set was fist published by the authors of [12]. Flixster is a social networking service in which users can rate movies. Users can also add some users to their friend list and create a social network. Note that social relations in Flixster are undirected. Epinions is an online product review service where users are allowed create social relations to other users. We used the version of the Epinions dataset[9] published by the authors of [18]. It should be noted that the social relations are directed in Epinions and undirected in Flixster. Since our model considers directed social relations, we represent each undirected social relation in Flixster as two directed social relations.

Since GSBM, like MMB, models the behavior of users at the interaction level, it learns many latent variables. Therefore, the model is slower than existing models. On the other hand, the model is capable of not only performing rating prediction, but also link prediction and community discovery. For the sake of efficiency, we used samples of the Flixster and the Epinions data set in our experiments. Table 1 shows the general statistics of the sampled data sets from Epinions and Flixster. Note that the size of our samples is still fairly large compared to the data set used in [1] (which contained less than 1000 users). It should be noted that convergence of algorithm 1 on each data set takes around 24 hours in the implementation described in section 5.

Statistics	**Flixster**	**Epinions**
Social Relations	41.5K	41.8K
Ratings	55.2K	62.2K
Users	10.6K	11.5K
Items	4.1K	8.8K

Table 1: General statistics of the Flixster and the Epinions data set used in the experiments.

6.2 Experiments on Rating Prediction

GSBM can be used for rating prediction. After learning the parameters of GSBM from the training data, rating prediction can be performed using the learnt latent variables. Given a user u and an item i we compute the predicted rating $\hat{r}_{u,i}$ as follows:

[6] http://software.intel.com/en-us/articles/intel-cilk-plus/

[7] www.flixster.com

[8] www.epinions.com

[9] http://alchemy.cs.washington.edu/data/epinions/

$$\hat{r}_{u,i} = \frac{\sum_{m=1}^{K_1} \sum_{n=1}^{K_2} \left(\gamma_{u,m}\delta_{i,n} \sum_{l=1}^{5} (l \times B_{R_{m,n,l}})\right)}{\left(\sum_{m=1}^{K_1} \gamma_{u,m}\right)\left(\sum_{n=1}^{K_2} \delta_{i,n}\right)} \tag{17}$$

As is the standard in recommender systems, we use RMSE (root mean squared error) to evaluate the accuracy of the recommendations inferred from GSBM comparing them to the withheld ground truth ratings in the test data. We chose the following comparison partners:

- State-of-the-art methods for recommendation in social networks: TrustWalker [11], TidalTrust [8], MoleTrust [15] and SocialMF [12].
- State-of-the-art methods for recommendation using only the user item matrix: user based collaborative filtering method (CF [9]) and matrix factorization based approach (MF [19]).

Model	**Flixster**	**Epinions**
CF	0.913	1.181
MF	0.911	1.175
TrustWalker	0.841	1.079
TidalTrust	0.887	1.109
MoleTrust	0.899	1.104
SocialMF	0.815	1.075
GSBM	0.884	1.092

Table 2: RMSE values of different comparison partners on Flixster and Epinions.

Table 2 and Figures 2 and 3 present the RMSE values for the comparison partners. The proposed GSBM achieves a very good RMSE, clearly outperforming the standard recommendation methods CF and MF and social network based approaches such as MoleTrust and TidalTrust. Compared to the more sophisticated methods for recommendation in social networks, such as SocialMF and TrustWalker, GSBM achieves lower but rather close performance. Note that all the comparison partners do only rating prediction, while GSBM is more comprehensive and also performs link prediction as well as community discovery. A more detailed discussion of the experimental results is presented in section 6.4.

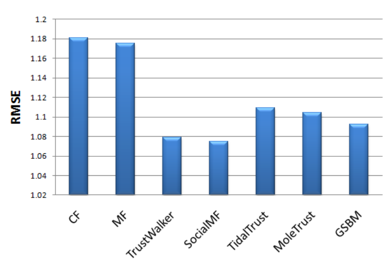


Figure 2: RMSE values for the comparison partners on Epinions.

6.3 Experiments on Link Prediction

The trained model can be employed for the task of link prediction in a social rating network. The probability $p(T_{u,v})$ of a link from user u to user v can be computed as follows:

$$p(T_{u,v}) \propto \frac{\sum_{m=1}^{K_1} \sum_{n=1}^{K_1} \left(\gamma_{u,m}\gamma_{v,n} B_{T_{m,n}}\right)}{\left(\sum_{m=1}^{K_1} \gamma_{u,m}\right)\left(\sum_{n=1}^{K_1} \gamma_{v,n}\right)} \tag{18}$$

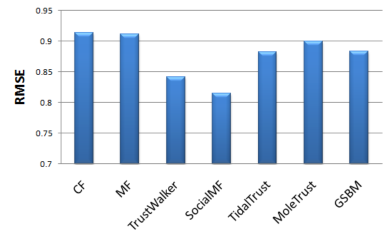
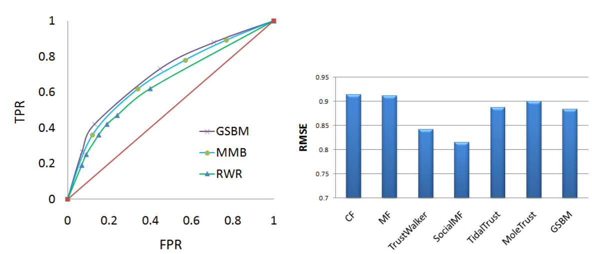


Figure 3: RMSE values for the comparison partners on Flixster.

We threshold the link probability to predict whether or not u creates a link to v. We use the ROC curve, more specifically the area under the ROC curve, to compare our proposed model with existing models. Note that since we need to determine the false positive rate, we randomly add some pairs of users who do not have a social relation in the training data to the test data to have some ground truth for negative results[10]. We consider the mixed membership stochastic blockmodel (MMB) as the baseline method. We use the random walk with restart method(RWR) [17] as another comparison partner. RWR is a state-of-the-art probabilistic method for link prediction in social networks. The probabilistic nature of RWR together with its good performance allows us to generate ROC curves used in our performance evaluation.



(a) Epinions (b) Flixster

Figure 4: ROC curves for link prediction in GSBM.

Figures 6.3 presents the ROC curves for link prediction with GSBM, MMB and RWR on Flixster and Epinions. As shown in the figures, both GSBM and MMB clearly outperform RWR in Flixster and Epinions. Comparing the ROC curves of GSBM and MMB, we observe that GSBM achieves slightly better performance. We believe that this improvement is due to the exploitation of the additional information in the user rating matrix which allows GSBM to take the selection effect into account.

6.4 Discussion

Our experimental results show that GSBM achieves high quality both in rating prediction and link prediction. In rating prediction, GSBM clearly outperforms all comparison partners that only consider the rating matrix (CF and MF) and also outperforms the social recommenders that do not take the rating patterns into account for guiding the traversal of the social network (TidalTrust and MoleTrust). The more sophisticated social recommender methods that use the rating patterns in the search of the social network, however, slightly outperform GSBM. Note

[10]The number of positive and negative test data is the same in our experiments.

that all these comparison partners do only rating prediction, while GSBM covers more tasks and also performs link prediction as well as community discovery. Therefore, GSBM is generally expected to sacrifice some rating prediction accuracy in order to optimize all three tasks simultaneously.

In link prediction, GSBM achieves better quality than all the comparison partners. We believe that this is mainly due to the fact that GSBM can exploit the selection effect in social networks by also considering the rating patterns for the link prediction task. To summarize, our experimental results demonstrate that GSBM achieves high quality in both rating prediction and link prediction while providing a more comprehensive framework than existing methods that perform only one of these tasks.

7. CONCLUSION

With the increasing availability of online social networks and the well-known effect of social influence, modeling the behavior of users in online social rating networks has attracted a lot of attention in the research community. Social influence and selection together lead to the formation of communities of like-minded and well-connected users.

In this paper, we adopted the idea of overlapping and probabilistic group membership presented in the mixed membership stochastic blockmodel (MMB) [1] and extended it to model not only the creation of the social relations, but also the rating behavior for groups of users and items. The proposed GSBM is capable of predicting the future behavior of users, both the social relation creation and the ratings of items. Besides, GSBM can infer communities among users with probabilistic membership assignments.

We performed experiments on two real life data sets from Flixster.com and Epinions.com, employing GSBM for rating prediction and link prediction. Experimental results show that GSBM achieves high quality results in both tasks. Although GSBM does not outperform all comparison partners in rating prediction, it achieves results very close to the state-of-the-art methods while being able to perform multiple tasks (community discovery, rating prediction and link prediction). Due to exploitation of the rating matrix, the link prediction quality of GSBM is better than that of MMB. In other words, GSBM takes the selection effect into account which improves the performance in link prediction.

This paper suggests several directions for future research. Since GSBM learns the latent variables for interactions between all user pairs and all user-item pairs, it has a relatively high runtime complexity. However, GSBM is highly parallelizable, and the speedup is almost linear with respect to the number of available cores. We plan to improve the scalability of GSBM with a more sophisticated parallelization. One of the main features of GSBM is community discovery. However, since we did not have access to a data set with ground truth for the communities, we used predictive models to indirectly evaluate the discovered communities. We are seeking data sets providing such ground truth and will evaluate GSBM directly in terms of its accuracy in community discovery.

8. REFERENCES

[1] E. M. Airoldi, D. M. Blei, S. E. Fienberg, and E. P. Xing. Mixed membership stochastic blockmodels. *Journal of Machine Learning Research*, 9:1981–2014, 2008.

[2] L. Backstrom, D. Huttenlocher, J. Kleinberg, and X. Lan. Group formation in large social networks: membership, growth, and evolution. In *KDD'06*, 2006.

[3] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2 edition, 2007.

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

[5] J. Coleman. *Foundations of Social Theory*. Harvard University Press, 1990.

[6] D. Crandall, D. Cosley, D. Huttenlocher, J. Kleinberg, and S. Suri. Feedback effects between similarity and social influence in online communities. In *KDD 2008*.

[7] N. E. Friedkin. *A Structural Theory of Social Influence*. Cambridge University Press, 1998.

[8] J. Golbeck. *Computing and Applying Trust in Web-based Social Networks*. PhD thesis, University of Maryland College Park, 2005.

[9] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry. Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 35(12), 1992.

[10] P. Holme and M. Newman. Nonequilibrium phase transition in the coevolution of networks and opinions. *Physical Review E*, 74:056108, 2006.

[11] M. Jamali and M. Ester. Trustwalker: A random walk model for combining trust-based and item-based recommendation. In *KDD 2009*.

[12] M. Jamali and M. Ester. A matrix factorization technique with trust propagation for recommendation in social networks. In *RecSys*, pages 135–142, 2010.

[13] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *IEEE Computer*, 42:30–37, 2009.

[14] H. Ma, I. King, and M. R. Lyu. Learning to recommend with social trust ensemble. In *SIGIR 2009*, pages 203–210.

[15] P. Massa and P. Avesani. Trust-aware recommender systems. In *RecSys 2007*, USA.

[16] M. Mcpherson, L. S. Lovin, and J. M. Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(1):415–444, 2001.

[17] J.-Y. Pan, H.-J. Yang, C. Faloutsos, and P. Duygulu. Automatic multimedia cross-modal correlation discovery. In *KDD'04*, pages 653–658. ACM, 2004.

[18] M. Richardson and P. Domingos. Mining knowledge-sharing sites for viral marketing. In *KDD 2002*.

[19] R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. In *NIPS 2008*, volume 20.

[20] T. A. Snijders and K. Nowicki. Estimation and prediction for stochastic blockmodels for graphs with latent block structure. *Journal of Classification*, 14, 1997.

[21] L. Ungar and D. Foster. Clustering methods for collaborative filtering. In *Proceedings of the Workshop on Recommendation Systems*. AAAI Press, Menlo Park California, 1998.

[22] S. Wasserman and K. Faust. *Social Network Analysis*. Cambridge Univ. Press, 1994.