Nonparametric Relational Learning for Social Network Analysis

Zhao Xu
Knowledge Discovery
Fraunhofer IAIS
zhao.xu@web.de

Volker Tresp
Corporate Technology
Siemens AG
volker.tresp@siemens.com

Shipeng Yu
Siemens Medical Solutions
shipeng.yu@siemens.com

Kai Yu
NEC Laboratories America
kyu@sv.nec-labs.com

ABSTRACT

Social networks usually involve rich collections of objects, which are jointly linked into complex relational networks. Social network analysis has gained in importance due to the growing availability of data on novel social networks, e.g. citation networks, Web 2.0 social networks like facebook, and the hyperlinked internet. Recently, the infinite hidden relational model (IHRM) has been developed for the analysis of complex relational domains. The IHRM extends the expressiveness of a relational model by introducing for each object an infinite-dimensional hidden variable as part of a Dirichlet process mixture model. In this paper we discuss how the IHRM can be used to model and analyze social networks. In such an IHRM-based social network model, each edge is associated with a random variable (RV) and the probabilistic dependencies between these RVs are specified by the model based on the relational structure. The hidden variables, one for each object, are able to transport information such that non-local probabilistic dependencies can be obtained. The IHRM provides effective relationship prediction and cluster analysis for social networks. The experimental analysis is performed on two social network applications. The first application is an analysis of the cooperative effect in a recommendation framework where both user properties and item properties are taken into account. The experimental results demonstrate that the IHRM provides good prediction accuracy for user preference on movies and gives interpretable clusters of users and items. In the second experiment we apply the IHRM to Sampson's monastery data, and obtain a grouping of the actors that agrees with results from previous publications. As an additional contribution of this paper, we present a new mean field approximation to inference in the IHRM.

The 2nd SNA-KDD Workshop '08 (SNA-KDD'08), August 24, 2008, Las Vegas, Nevada, USA

Keywords

Statistical Relational Learning, Nonparametric Mixture Models, Dirichlet Process, MCMC Sampling, Variational Approximation

1. INTRODUCTION

Social networks usually consist of rich collections of objects, which are linked into complex relational networks. Statistical relational learning (SRL) [10, 19, 7] is an emerging area of machine learning research which attempts to combine expressive knowledge representation formalisms with statistical approaches to perform probabilistic inference and learning on relational networks. SRL provides effective tools for social network modeling and analysis, such as community discovery and product recommendation. Social networks are graphically represented as a sociogram as illustrated in Figure 1 (top). In this simple relational network, a common task is to make predictions on unknown relationships (friendship) based on known relationships and person profiles (e.g., gender). We can use probabilistic approaches to model the relational network such that the quantities of interest can be inferred with statistical techniques. Figure 1 (bottom) shows a probabilistic model for the sociogram. For each potential edge, a random variable (RV) is introduced that describes the state of the edge. For example, there is a RV associated with the edge between the person 1 and the person 2. The binary variable is YES if the two persons are friends and No otherwise. The edge between the person 1 and Male is also associated with a RV, whose value describes the person's profile. In this example, all variables are binary. To infer the quantities of interest, e.g., whether the person 1 and the person 2 are friends, we need to learn the probabilistic dependencies between the random variables. Here we assume that friendship is conditioned on the profiles (gender) of the involved persons. Thus the probabilistic dependencies can be as shown in Figure 1 (bottom). The directed arcs, for example, the ones between G_1 and $R_{1,2}$ and between G_2 and $R_{1,2}$ specify that the probability that person 1 and the person 2 are friends depends on their respective profiles. Given the probabilistic model, we can learn the parameters and predict the relationships of interest.

In this relational model, the friendship is locally predicted by the profiles of the involved objects: whether a person is a friend of another person is only dependent on the profiles of the two persons. Given that the parameters are fixed, and

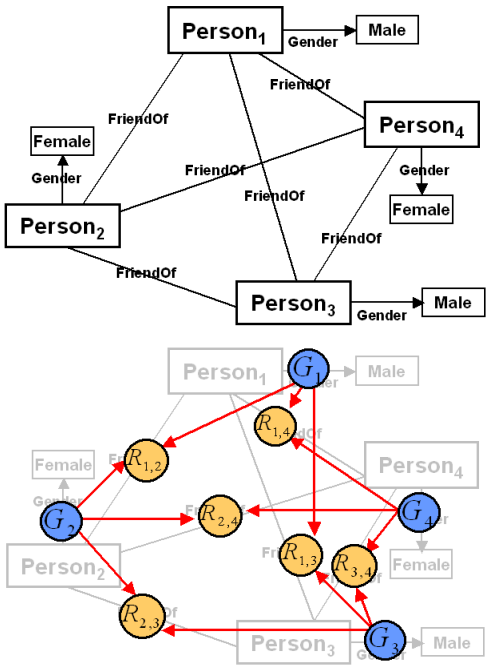

Figure 1: Top: A simple sociogram. Bottom: A probabilistic model for the sociogram. Each edge is associated with a random variable that determines the state of the edge. The directed arcs indicate direct probabilistic dependencies.

given the parent attributes, all friendships are independent of each other such that correlations between friendships, i.e., the collaborative effect, cannot be taken into account. To solve this limitation, structural learning might be involved to obtain non-local dependencies but structural learning in complex relational networks is considered a hard problem [8].

Non-local dependencies can be achieved by introducing for each person a hidden variable as it was proposed in [26]. The state of the hidden variable represents unknown attributes of the person, e.g. the particular habit of making friends of with certain persons. The hidden variable of a person is now the only parent of its profiles and is one of the parents of the friendships in which the person potentially participates. Since the hidden variables are of central importance, this model is referred to as the *hidden relational model* (HRM). A ground Bayesian network of an HRM forms a network of hidden variables over the relational structure. The HRM can be considered a direct generalization of hidden Markov model used in speech recognition or hidden Markov random field used in computer vision [27]. As in those models, information can propagate across the network of hidden variables. The HRM can also be interpreted as a relational mixture model, which clusters all objects in a relational domain. The state of the hidden variable of an object corresponds to its cluster assignment. The HRM clustering can be viewed on as a generalization of co-clustering [12].

In relational domains, different classes of objects generally require a class-specific complexity in the hidden representation. Thus it is sensible to work with a Dirichlet process (DP) mixture model in which each class of objects

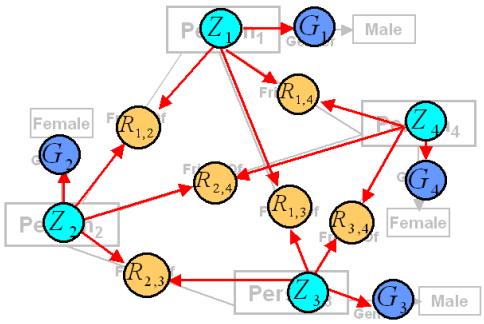

Figure 2: A hidden relational model (HRM) for a simple sociogram.

can optimize their own representational complexity in a self-organized way. Conceptionally, the number of states in the hidden variables in the HRM becomes infinite. In practice, the DP mixture sampling process only occupies a finite number of components. The combination of the hidden relational model and the DP mixture model is the *infinite hidden relational model* (IHRM) [26]. The IHRM is related to the model introduced in [16]. Section 6 discusses similarities and differences between the two models.

The IHRM has been presented first in [26]. In this paper, we explore social network analysis with IHRM for modeling, clustering and prediction. We also develop two inference methods for efficient inference: one is the blocked Gibbs sampling with truncated stick-breaking (TSB) construction, the other is the mean-field approximation with TSB. We perform empirical analysis on the MovieLens data and Sampson's monastery data for community discovery and product recommendation. The paper is organized as follows. In the next section, we analyze the infinite hidden relational model for social networks. In Section 3 and Section 4 we describe a Gibbs sampling method and a mean-field approximation for inference in the IHRM. Section 5 provides experimental analysis. In Section 6 we review and discuss some related work. Finally Section 7 concludes the paper.

2. MODEL DESCRIPTION

Based on the analysis in Section 1, we will give a detailed description of the IHRM. In this section, we first introduce the finite hidden relational model (HRM), and then extend it to an infinite version (IHRM). In addition, we provide a generative model describing how to generate data from an IHRM.

2.1 Hidden Relational Model

A hidden relational model (HRM) for a simple sociogram is shown in Figure 2. The basic innovation of the HRM is to introduce for each person a hidden variable, in the example denoted as Z. They can be thought of as unknown attributes of the persons. We then assume that attributes of a person only depend on the hidden variable of the person, and a relationship only depends on the hidden variables of the persons involved in the relationship. It implies that if the hidden variables were known, both person attributes and relationships can be well predicted.

Given the HRM model shown as Figure 2, information can propagate via interconnected hidden variables. Let us

predict whether the person 2 will be a friend of the person 3, i.e. the relationship $R_{2,3}$. The probability is computed on the evidence about: (1) the attributes of the immediately related persons, i.e. G_2 and G_3, (2) the known relationships associated with the persons of interest, i.e. the friendships $R_{2,1}$ and $R_{2,4}$ about the person 2, and the friendships $R_{1,3}$ and $R_{3,4}$ about the person 3, (3) *high-order* information transferred via hidden variables, e.g. the information about G_1 and G_4 propagated via Z_1 and Z_4. If the attributes of persons are informative, those will determine the hidden states of the persons, therefore dominate the computation of predictive probability of relationship R. Conversely, if the attributes of persons are weak, then the hidden state of a person might be determined by his relationships to other persons and the hidden states of those persons. In summary, by introducing hidden variables, information can globally distribute in the ground network defined by the relational structure. This reduces the need for extensive structural learning, which is particularly difficult in relational models due to the huge number of potential parents. Note that a similar propagation of information can be observed in hidden Markov models used in speech recognition or in the hidden Markov random fields used in image analysis [27]. In fact, the HRM can be viewed as a direct generalization of both for relational data. Additionally, the HRM provides a cluster analysis of relational data. The assignments of hidden variables specify the clusters of the persons. The HRM can be applied to domains with multiple classes of objects and multiple classes of relationships and relationships can be of arbitrary order, i.e. are not constraint to binary relationships[26]. Also note that the sociogram is quite related to the RDF-graph used as the basic data model in the semantic web [3].

We now complete the model by introducing the variables and parameters in Figure 2. There is a hidden variable Z_i for each person. The state of Z_i specifies the cluster of the person i. Let K denote the number of clusters. In the HRM, K is a hyperparameter, whose value is either given or can be computed with an empirical Bayesian method. Z follows multinomial distribution with parameter vector $\pi = (\pi_1, \ldots, \pi_K)$ $(\pi_k > 0, \sum_k \pi_k = 1)$, which specify the probability of a person belonging to a cluster, i.e. $P(Z_i = k) = \pi_k$. π is sometimes referred to as mixing weights, and is drawn from a conjugated Dirichlet prior with hyperparameters α_0. Note that α_0 is a K-dimensional vector in the HRM.

All person attributes are assumed to be discrete and multinomial variables (resp., binary and Bernoulli). Thus a particular person attribute G_i is a sample drawn from a multinomial (resp., Bernoulli) distribution with parameters θ_k, where k denotes the cluster assignment of the person. θ_k is sometimes referred to as mixture component, which is associated with the cluster k. For all persons, there are totally K mixture components $\Theta = (\theta_1, \ldots, \theta_K)$. Each person in the cluster k inherits the mixture component, thus we have: $P(G_i = s|Z_i = k, \Theta) = \theta_{k,s}$ $(\theta_{k,s} > 0, \sum_s \theta_{k,s} = 1)$. These mixture components are independently drawn from a prior G_0. For computational efficiency, we assume that G_0 is a conjugated Dirichlet prior with hyperparameters β.

We now consider the variables and parameters about relationships (FriendOf). The relationship R is assumed to be discrete with two states. A particular relationship $R_{i,j}$ between two persons (i and j) is a sample drawn from a binomial distribution with a parameter $\phi_{k,\ell}$, where k and ℓ denote cluster assignments of the person i and the person j, respectively. There are totally $K \times K$ parameters $\phi_{k,\ell}$, and each $\phi_{k,\ell}$ is independently drawn from the prior G_0^r. For computational efficiency, we assume that G_0^r is a conjugated Beta distribution with hyperparameters β^r.

From mixture model point of view, the most interesting term in the HRM is $\phi_{k,\ell}$, which can be interpreted as a *correlation mixture component*. If a person i is assigned to a cluster k, i.e. $Z_i = k$, then the person inherits not only θ_k, but also $\phi_{k,\ell}, \ell = \{1, \ldots, K\}$.

2.2 Infinite Hidden Relational Model

Since the hidden variables play a key role in the HRM, we would expect that the HRM might require a flexible number of states for the hidden variables. Consider again the simple sociogram example. With little information about past friendships, all persons might look the same; with more information available, one might discover certain clusters in the persons (different habits of making friends); but with an increasing number of past friendships, the clusters might show increasingly detailed structure ultimately indicating that everyone is an individual. It thus makes sense to permit an arbitrary number of clusters by using a Dirichlet process mixture model. This permits the model to decide itself about the optimal number of clusters and to adopt the optimal number with increasing data. For our discussion it suffices to say that we obtain an infinite HRM by simply letting the number of clusters approach infinity, $K \to \infty$. Although from a theoretical point of view there are indeed an infinite number of components, a sampling procedure would only occupy a finite number of components.

The graphical representations of the IHRM and the HRM are identical, shown as Figure 2. However, the definitions of variables and parameters are different. For example, the hidden variables Z of persons have infinite states, and thus the parameter vector π is infinite-dimensional. The parameter is not generated from a Dirichlet prior, but from a *stick breaking construction* Stick$(\cdot|\alpha_0)$ with a hyperparameter α_0 (more details in the next section). Note that α_0 is a positive real-valued scalar and is referred to as *concentration parameter* in DP mixture modeling. It determines the tendency of the model to either use a large number or a small number of states in the hidden variables [2]. If α_0 is chosen to be small, only few clusters are generated. If α_0 is chosen to be large, the coupling is loose and more clusters are formed. Since there are an infinite number of clusters, there are an infinite number of mixture components θ_k, each of which is still independently drawn from G_0. G_0 is referred to as *base distribution* in DP mixture modeling.

2.3 Generative Model

Now we describe the generative model for the IHRM. There are mainly two methods to generate samples from a Dirichlet Process (DP) mixture model, i.e. the Chinese restaurant process (CRP) [2] and the stick breaking construction (SBC) [24]. We will discuss how SBC can be applied to the IHRM (for CRP-based generative model, please refer to [26]).

To describe the generative model, we need some notation. (summarized in Table 1). Assume that there are C classes of objects and B classes of relationships. For an object class c, there are N^c objects e_i^c indexed by i, a base distribution

Table 1: Notation used in this paper.

Symbol	Description
C	number of object classes
B	number of relationship classes
N^c	number of objects in the class c
α_0^c	concentration parameter of an object class c
e_i^c	an object indexed by i in a class c
A_i^c	an attribute of an object e_i^c
θ_k^c	mixture component indexed by the hidden state k in the object class c
G_0^c	base distribution of an object class c
β^c	parameters of the base distribution G_0^c
$R_{i,j}^b$	relationship of class b between objects i, j
$\phi_{k,\ell}^b$	correlation mixture component indexed by hidden states k for c_i and ℓ for c_j, where c_i and c_j are indexes of object classes involved in the relationship class b
G_0^b	base distribution of a relationship class b
β^b	parameters of the base distribution G_0^b

G_0^c, and a concentration parameter α_0^c. θ_k^c denotes a mixture component, which is the parameter vector of the distribution of an object attribute. For a relationship class b between two object classes c_i and c_j, there is a base distribution G_0^b associated. $\phi_{k,\ell}^b$ denotes a correlation mixture component indexed by hidden states k for c_i and ℓ for c_j, which is the parameter vector of the distribution of a relationship. Here we restrict ourselves that the object attributes and relationships are drawn from exponential family distributions with parameters θ_k^c and $\phi_{k,\ell}^b$, respectively. The base distributions G_0^c and G_0^b are conjugated priors with hyperparameters β^c and β^b.

The stick breaking construction (SBC) [24] is a representation of a DP, by which we can explicitly sample the random distributions of attribute parameters and relationship parameters. In the following we describe the generative model of the IHRM in terms of the SBC.

1. For each object class c,
   (a) Draw mixing weights $\pi^c \sim \text{Stick}(\cdot|\alpha_0^c)$, defined as breaking construction defined as
   $$V_k^c \overset{\text{iid}}{\sim} \text{Beta}(1, \alpha_0^c);$$
   $$\pi_1^c = V_1^c, \quad \pi_k^c = V_k^c \prod_{k'=1}^{k-1} (1 - V_{k'}^c),\ k > 1. \quad (1)$$
   (b) Draw i.i.d. mixture components $\theta_k^c \sim G_0^c$, $k = 1, 2, \ldots$
2. For each relationship class b between two object classes c_i and c_j, draw $\phi_{k,\ell}^b \sim G_0^b$ i.i.d. with component indices k for c_i and ℓ for c_j.
3. For each object e_i^c in a class c,
   (a) Draw cluster assignment $Z_i^c \sim \text{Mult}(\cdot|\pi^c)$;
   (b) Draw object attributes $A_i^c \sim P(\cdot|\theta^c, Z_i^c)$.
4. For objects $e_i^{c_i}$ and $e_j^{c_j}$ with a relationship of class b, draw $R_{i,j}^b \sim P(\cdot|\phi^b, Z_i^{c_i}, Z_j^{c_j})$.

The basic property of the SBC is that: the distributions of the parameters (θ_k^c and $\phi_{k,\ell}^b$) are sampled, e.g., the distribution of θ_k^c can be represented as $G^c = \sum_{k=1}^{\infty} \pi_k^c \delta_{\theta_k^c}$, where $\delta_{\theta_k^c}$ is a distribution with a point mass on θ_k^c. In terms of this property, the SBC can sample objects independently; thus the SBC might be efficient when a large domain is involved.

3. INFERENCE WITH GIBBS SAMPLING

The key inferential problem in the IHRM is to compute the joint posterior distribution of unobservable variables given the data D, i.e. $P(\{\pi^c, \Theta^c, Z^c\}_c, \{\Phi^b\}_b | D, \{\alpha_0^c, G_0^c\}_c, \{G_0^b\}_b)$. Unfortunately, the computation of the joint posterior is analytically intractable, thus we consider approximate inference methods to solve the problem.

Markov chain Monte Carlo (MCMC) sampling has been used to approximate posterior distribution with a DP mixture prior. In this section, we extend these MCMC methods to the IHRM. [26] explored a Gibbs sampler with the Chinese restaurant process, which is a collapsed version of Pólya urn sampling [17]. Blocked sampling typically exhibits more rapid mixing of the Markov chain than collapsed sampling [14]. Thus we extend the efficient blocked Gibbs sampling (GS) with truncated stick breaking representation [13] to the IHRM.

In the blocked GS, the posterior distributions of parameters (π^c, Θ^c and Φ^b) are explicitly sampled in the form of truncated stick breaking construction [13]. The advantage is that given the posterior distributions, we can independently sample the hidden variables in a block, which highly accelerates the computation. The Markov chain is thus defined not only on the hidden variables, but also on the parameters. At the iteration t, the sampled variables include $Z_i^{c(t)}$, $\pi^{c(t)}$, $\Theta^{c(t)}$ and $\Phi^{b(t)}$.

Truncated stick breaking construction (TSB) fixes a value K^c for each class of objects and lets $V_{K^c}^c = 1$. That means the mixing weights π_k^c are equal to 0 for $k > K^c$ (refer to Equation 1). The number of the clusters is thus reduced to K^c. When K^c is large enough, the truncated Dirichlet process provides a close approximation to the true Dirichlet process [13]. Note, that K^c is an additional parameter in the inference method.

At each iteration, we first update the hidden variables conditioned on the parameters sampled in the last iteration, and then update the parameters conditioned on the hidden variables. In detail:

1. For each class of objects,
   (a) Update each hidden variable $Z_i^{c(t+1)}$ with probability proportional to:
   $$\pi_k^{c(t)} P(A_i^c | Z_i^{c(t+1)} = k, \Theta^{c(t)}) \times \prod_{b'} \prod_{j'} P(R_{i,j'}^{b'} | Z_i^{c(t+1)} = k, Z_{j'}^{c_{j'}(t)}, \Phi^{b'(t)}), \quad (2)$$
   where A_i^c and $R_{i,j'}^{b'}$ denotes the known attributes and relationships about the object i. $c_{j'}$ denotes the class of the object j', $Z_{j'}^{c_{j'}(t)}$ denotes the hidden variable of j' at the last iteration t. Intuitively, the equation represents to what extent the cluster k agrees with the data D_i^c of the object.
   (b) Update $\pi^{c(t+1)}$ as follows:

i. Sample $v_k^{c(t+1)}$ from Beta$(\lambda_{k,1}^{c(t+1)}, \lambda_{k,2}^{c(t+1)})$ for $k = \{1, \ldots, K^c - 1\}$ with

$$\lambda_{k,1}^{c(t+1)} = 1 + \sum_{i=1}^{N^c} \delta_k(Z_i^{c(t+1)}),$$

$$\lambda_{k,2}^{c(t+1)} = \alpha_0^c + \sum_{k'=k+1}^{K^c} \sum_{i=1}^{N^c} \delta_{k'}(Z_i^{c(t+1)}), \quad (3)$$

and set $v_{K^c}^{c(t+1)} = 1$. $\delta_k(Z_i^{c(t+1)})$ equals to 1 if $Z_i^{c(t+1)} = k$ and 0 otherwise.

ii. Compute $\pi^{c(t+1)}$ as: $\pi_1^{c(t+1)} = v_1^{c(t+1)}$ and

$$\pi_k^{c(t+1)} = v_k^{c(t+1)} \prod_{k'=1}^{k-1} (1 - v_{k'}^{c(t+1)}), \quad k > 1. \quad (4)$$

2. Update parameters:

$$\theta_k^{c(t+1)} \sim P(\cdot|A^c, Z^{c(t+1)}, G_0^c),$$
$$\phi_{k,\ell}^{b(t+1)} \sim P(\cdot|R^b, Z^{(t+1)}, G_0^b). \quad (5)$$

The parameters are drawn from their posterior distributions conditioned on the sampled hidden states. Again, since we assume conjugated priors as the base distributions (G_0^c and G_0^b), the simulation is tractable.

After convergence, we collect the last W samples to make predictions for the relationships of interest. Note that in blocked Gibbs sampling, the MCMC sequence is defined by hidden variables and parameters, including $Z^{c(t)}$, $\pi^{c(t)}$, $\Theta^{c(t)}$, and $\Phi^{b(t)}$. The predictive distribution of a relationship $R_{new,j}^b$ between a new object e_{new}^c and a known object $e_j^{c_j}$ is approximated as

$$\begin{aligned}
&P(R_{new,j}^b|D, \{\alpha_0^c, G_0^c\}_{c=1}^C, \{G_0^b\}_{b=1}^B) \\
&\approx \frac{1}{W} \sum_{t=w+1}^{W+w} P(R_{new,j}^b|D, \{Z^{c(t)}, \pi^{c(t)}, \Theta^{c(t)}\}_{c=1}^C, \{\Phi^{b(t)}\}_{b=1}^B) \\
&\propto \frac{1}{W} \sum_{t=w+1}^{W+w} \sum_{k=1}^{K^c} P(R_{new,j}^b|\phi_{k,\ell}^{b(t)})\, \pi_k^{c(t)} P(A_{new}^c|\theta_k^{c(t)}) \\
&\times \prod_{b'} \prod_{j'} P(R_{new,j'}^{b'}|\phi_{k,\ell'}^{b'(t)}),
\end{aligned}$$

where ℓ and ℓ' denote the cluster assignments of the objects j and j', respectively. The equation is quite intuitive. The prediction is a weighted sum of predictions $P(R_{new,j}^b|\phi_{k,\ell}^{b(t)})$ over all clusters. The weight of each cluster is the product of the last three terms, which represents to what extent this cluster agrees with the known data (attributes and relationships) about the new object. Since the blocked method also samples parameters, the computation is straightforward.

4. INFERENCE WITH VARIATIONAL APPROXIMATION

The IHRM has multiple DPs which interact through relationships, thus blocked Gibbs sampling is still slow due to the slow exchange of information between DPs. To solve the problem, we explore an alternative solution by variational inference method. The main strategy of these methods is to convert a probabilistic inference problem into an optimization problem, and then to solve the problem with the known optimization techniques. In particular, the methods assume a distribution q, referred to as a *variational distribution*, to approximate the true posterior P as close as possible. The difference between the variational distribution q and the true posterior P can be measured via *Kullback-Leibler* (KL) divergence. Let ξ denote a set of unknown quantities, let D denote the known data. The KL divergence between $q(\xi)$ and $P(\xi|D)$ is defined as:

$$KL(q(\xi)||P(\xi|D)) = \sum_{\xi} q(\xi) \log q(\xi) - \sum_{\xi} q(\xi) \log P(\xi|D). \quad (6)$$

The smaller the divergence, the better is the fit between the true and the approximate distributions.

Thus the probabilistic inference problem (i.e. computing the posterior) is converted into an optimization problem: to minimize the KL divergence with respect to the variational distribution. In practice, the minimization of the KL divergence is formulated as the maximization of the lower bound of the log-likelihood of the data.

$$\begin{aligned}
\log P(D) &= \sum_{\xi} q(\xi) \log P(D, \xi) - \sum_{\xi} q(\xi) \log q(\xi) \\
&\quad + KL(q(\xi)||P(\xi|D)) \\
&\geq \sum_{\xi} q(\xi) \log P(D, \xi) - \sum_{\xi} q(\xi) \log q(\xi). \quad (7)
\end{aligned}$$

The challenge is now to find suitable forms of variational distributions to make the optimization problem computationally tractable. For the IHRM, we assume variational distribution as mean field with TSB, motivated by [5]. In this context mean field means that the variational distributions are assumed in the family of fully-factorized distributions. For more details about variational inference, please refer to [15].

A mean-field method was explored in [5] to approximate the posterior of unobservable quantities in a DP mixture model. We extend it to the IHRM. The main difference is that in the IHRM, there are multiple DP mixture models coupled together with relationships and correlation mixture components. In the IHRM, unobservable quantities include Z^c, π^c, Θ^c and Φ^b. Since the mixing weights π^c are computed on V^c (see Equation 1), we can replace π^c with V^c in the set of unobservable quantities. To approximate the posterior $P(\{V^c, \Theta^c, Z^c\}_c, \{\Phi^b\}_b|D, \{\alpha_0^c, G_0^c\}_c, \{G_0^b\}_b)$, we define a variational distribution $q(\{Z^c, V^c, \Theta^c\}_{c=1}^C, \{\Phi^b\}_{b=1}^B)$ as:

$$\left[\prod_c^C \prod_i^{N^c} q(Z_i^c|\eta_i^c) \prod_k^{K^c} q(V_k^c|\lambda_k^c) q(\theta_k^c|\tau_k^c)\right] \left[\prod_b^B \prod_k^{K^{c_i}} \prod_\ell^{K^{c_j}} q(\phi_{k,\ell}^b|\rho_{k,\ell}^b)\right], \quad (8)$$

where c_i and c_j denote the object classes involved in the relationship class b. k and ℓ denote the cluster indexes for c_i and c_j. Variational parameters include $\{\eta_i^c, \lambda_k^c, \tau_k^c, \rho_{k,\ell}^b\}$. $q(Z_i^c|\eta_i^c)$ is a multinomial distribution with parameters η_i^c. Note, that there is one η_i^c for each object e_i^c. $q(V_k^c|\lambda_k^c)$ is a Beta distribution. $q(\theta_k^c|\tau_k^c)$ is a distribution with the same form as G_0^c. $q(\phi_{k,\ell}^b|\rho_{k,\ell}^b)$ is a distribution with the same form as G_0^b.

We substitute Equation 8 into Equation 7 and optimize the lower bound with a coordinate ascent algorithm, which generates the following equations to iteratively update the

variational parameters until convergence:

$$\lambda^c_{k,1} = 1 + \sum_{i=1}^{N^c} \eta^c_{i,k}, \quad \lambda^c_{k,2} = \alpha^c_0 + \sum_{i=1}^{N^c} \sum_{k'=k+1}^{K^c} \eta^c_{i,k'}, \tag{9}$$

$$\tau^c_{k,1} = \beta^c_1 + \sum_{i=1}^{N^c} \eta^c_{i,k} \mathrm{T}(A^c_i), \quad \tau^c_{k,2} = \beta^c_2 + \sum_{i=1}^{N^c} \eta^c_{i,k}, \tag{10}$$

$$\rho^b_{k,\ell,1} = \beta^b_1 + \sum_{i,j} \eta^{c_i}_{i,k} \eta^{c_j}_{j,\ell} \mathrm{T}(R^b_{i,j}), \quad \rho^b_{k,\ell,2} = \beta^b_2 + \sum_{i,j} \eta^{c_i}_{i,k} \eta^{c_j}_{j,\ell}, \tag{11}$$

$$\eta^c_{i,k} \propto \exp\left(E_q[\log V^c_k] + \sum_{k'=1}^{k-1} E_q[\log(1 - V^c_{k'})] + E_q[\log P(A^c_i|\theta^c_k)] + \sum_{b'} \sum_{j} \sum_{\ell} \eta^{c_j}_{j,\ell} E_q[\log P(R^{b'}_{i,j}|\phi^{b'}_{k,\ell})] \right), \tag{12}$$

where λ^c_k denotes parameters of Beta distribution $q(V^c_k|\lambda^c_k)$, thus λ^c_k is a two-dimensional vector $\lambda^c_k = (\lambda^c_{k,1}, \lambda^c_{k,2})$. τ^c_k denotes parameters of the exponential family distribution $q(\theta^c_k|\tau^c_k)$. We decompose τ^c_k such that $\tau^c_{k,1}$ contains the first $dim(\theta^c_k)$ components and $\tau^c_{k,2}$ is a scalar. Similarly, β^c_1 contain the first $dim(\theta^c_k)$ components and β^c_2 is a scalar. $\rho^b_{k,\ell,1}$, $\rho^b_{k,\ell,2}$, β^b_1 and β^b_2 are defined equivalently. $\mathrm{T}(A^c_i)$ and $\mathrm{T}(R^b_{i,j})$ denote the *sufficient statistics* of the exponential family distributions $P(A^c_i|\theta^c_k)$ and $P(R^b_{i,j}|\phi^b_{k,\ell})$, respectively.

It is clear that Equation 9 and Equation 10 correspond to the updates for variational parameters of object class c, and they follow equations in [5]. Equation 11 represents the updates of variational parameters for relationships, which is computed on the involved objects. The most interesting updates are Equation 12, where the posteriors of object cluster-assignments are *coupled together*. These essentially connect the DPs together. Intuitively, in Equation 12 the posterior updates for $\eta^c_{i,k}$ include a prior term (first two expectations), the likelihood term about object attributes (third expectation), and the likelihood terms about relationships (last term). To calculate the last term we need to sum over all the relationships of the object e^c_i weighted by $\eta^{c_j}_{j,\ell}$ that is variational expectation about cluster-assignment of the other object involved in the relationship.

Once the procedure reaches stationarity, we obtain the optimized variational parameters, by which we can approximate the predictive distribution of the relationship $R^b_{new,j}$ between a new object e^c_{new} and a known object $e^{c_j}_j$:

$$\begin{aligned} & P(R^b_{new,j}|D, \{\alpha^c_0, G^c_0\}^C_{c=1}, \{G^b_0\}^B_{b=1}) \\ \approx\ & q(R^b_{new,j}|D, \lambda, \eta, \tau, \rho) \\ \propto\ & \sum_{k}^{K^c} \sum_{\ell}^{K^{c_j}} q(R^b_{new,j}|\rho^b_{k,\ell}) q(Z^{c_j}_j = \ell|\eta^{c_j}_j) q(Z^c_{new} = k|\lambda^c) \\ & \times q(A^c_{new}|\tau^c_k) \prod_{b'} \prod_{j'} \sum_{\ell'} q(Z^{c_{j'}}_{j'} = \ell'|\eta^{c_{j'}}_{j'}) q(R^{b'}_{new,j'}|\rho^{b'}_{k,\ell'}). \end{aligned} \tag{13}$$

The prediction is a weighted sum of predictions $q(R^b_{new,j}|\rho^b_{k,\ell})$ over all clusters. The weight consists of two parts. One is to what extent the cluster ℓ agrees with the object $e^{c_j}_j$ (i.e. the 2nd term), the other is to what extent the cluster k agrees with the new object (i.e. the product of the last 3 terms). The computations about the two parts are different. The reason is that $e^{c_j}_j$ is a known object, we have optimized variational parameters $\eta^{c_j}_j$ about its cluster assignment.

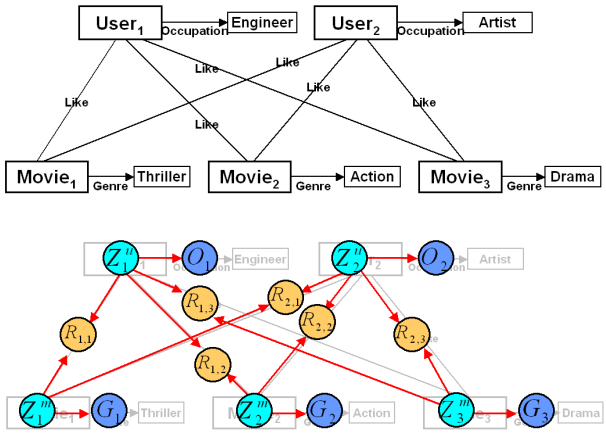

Figure 3: Top: A sociogram for movie recommendation system, illustrated with 2 users and 3 movies. For readability, only two attributes (user's occupation and movie's genre) show in the figure. Bottom: IHRM for the sociogram.

5. EXPERIMENTAL ANALYSIS

5.1 Movie Data

We first evaluate the IHRM on the MovieLens data [23]. There are in total 943 users and 1680 movies, and we obtain 702 users and 603 movies after removing low-frequent ones. Each user has about 112 ratings on average. The model is shown in Figure 3. There are two classes of objects (users and movies) and one class of relationships (Like). The task is to predict preferences of users. The users have attributes Age, Gender, Occupation, and the movies have attributes Published-year, Genres and so on. The relationships have two states, where $R = 1$ indicates that the user likes the movie and 0 otherwise. The user ratings in MovieLens are originally based on a five-star scale, so we transfer each rating to binary value with $R = 1$ if the rating is higher than the user's average rating, vice versa. The performance of the IHRM is analyzed from 2 points: prediction accuracy and clustering effect. To evaluate the prediction performance, we perform 4 sets of experiments which respectively select 5, 10, 15 and 20 ratings for each test user as the known ratings, and predict the remaining ratings. These experiments are referred to as *given5*, *given10*, *given15* and *given20* in the following. For testing the relationship is predicted to exist (i.e., $R = 1$) if the predictive probability is larger than a threshold $\varepsilon = 0.5$.

We implement the following 3 inference methods: Chinese restaurant process Gibbs sampling (CRPGS), truncated stick-breaking Gibbs sampling (TSBGS), and the corresponding mean field method TSBMF. The truncation parameters Ks for TSBGS and TSBMF are initially set to be the number of entities. For TSBMF we consider $\alpha_0 = \{5, 10, 100, 1000\}$, and obtain the best prediction when $\alpha_0 = 100$. For CRPGS and TSBGS α_0 is 100. For the variational methods, the change of variational parameters between two iterations is monitored to determine the convergence. For the Gibbs samplers, the convergence was analyzed by three measures: Geweke statistic on likelihood, Geweke statistic on the number of components for each class of objects, and autocorrelation. Figure 4 (left) shows the trace of the number of

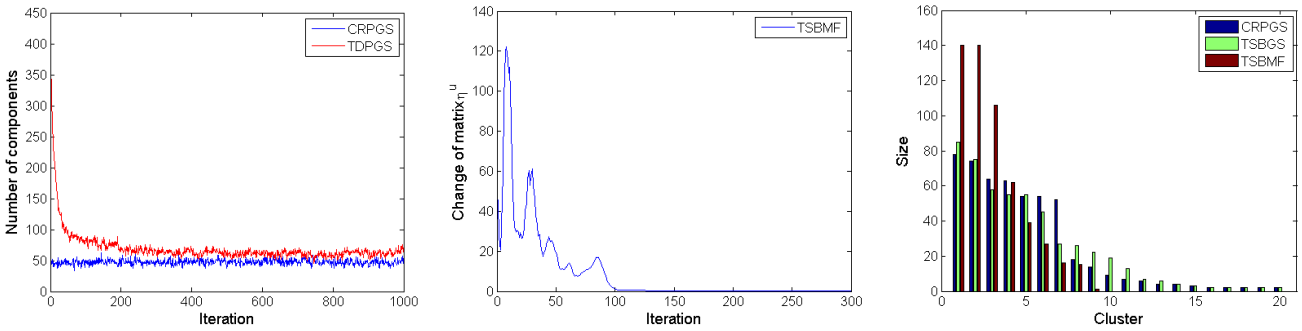

Figure 4: Left: The traces of the number of user clusters for the runs of 2 Gibbs samplers. Middle: The trace of the change of the variational parameter η^u for the mean field method. Right: The sizes of the largest user clusters of the three inference methods.

Table 2: Performance of the IHRM on MovieLens data.

	CRPGS	TSBGS	TSBMF	Pearson	SVD
Given5	65.13	65.51	65.26	57.81	63.72
Given10	65.71	66.35	65.83	60.04	63.97
Given15	66.73	67.82	66.54	61.25	64.49
Given20	68.53	68.27	67.63	62.41	65.13
Time(s)	164993	33770	2892	-	-
Time/iter.	109	17	19	-	-
$\#\mathbf{C}.^{\mathbf{u}}$	47	59	9	-	-
$\#\mathbf{C}.^{\mathbf{m}}$	77	44	6	-	-

user clusters in the 2 Gibbs samplers. Figure 4 (middle) illustrates the change of variational parameters η^u in the variational method. For CRPGS, the first $w = 50$ iterations (6942 s) are discarded as burn-in period, and the last $W = 1400$ iterations are collected to approximate the predictive distributions. For TSBGS, we have $w = 300$ (5078 s) and $W = 1700$. Although the number of iterations for the burn-in period is much less in the CRPGS if compared to the blocked Gibbs sampler, each iteration is approximately a factor 5 slower. The reason is that CRPGS samples the hidden variables one by one, which causes two additional time costs. First, the expectations of attribute parameters and relational parameters have to be updated when sampling each user/movie. Second, the posterior of hidden variables have to be computed one by one, thus we can not use fast matrix multiplication techniques to accelerate the computation. Therefore if we include the time, which is required to collect a sufficient number of samples for inference, the CRPGS is slower by a factor of 5 (the row Time(s) in Table 2) than the blocked sampler. The mean field method is again by a factor around 10 faster than the blocked Gibbs sampler and thus almost two orders of magnitude faster than the CRPGS.

The prediction results are shown in Table 2. All IHRM inference methods under consideration achieve comparably good performance; the best results are achieved by the two Gibbs sampling methods. To demonstrate the performance of the IHRM, we also implement Pearson-coefficient based collaborative filtering (CF) method [20] and an SVD-based CF method [22]. It is clear that the IHRM outperforms the traditional CF methods, especially when there are few known ratings for the test users. The main advantage of the IHRM is that it can exploit attribute information. If the attribute information is removed, the performance of the IHRM becomes close to the performance of the SVD approach. For example, after ignoring all attribute information, the TSBMF generates the predictive results: 64.55% for Given5, 65.45% for Given10, 65.90% for Given15, and 66.79% for Given20.

The IHRM provides cluster assignments for all objects involved, in our case for the users and the movies. The rows $\#C.^u$ and $\#C.^m$ in Table 2 denote the number of clusters for users and movies, respectively. The Gibbs samplers converge to 46-60 clusters for the users and 44-78 clusters for the movies. The mean field solution have a tendency to converge to a smaller number of clusters, depending on the value of α_0. Further analysis shows that the clustering results of the methods are actually similar. First, the sizes of most clusters generated by the Gibbs samplers are very small, e.g., there are 72% (75.47%) user clusters with less than 5 members in CRPGS (TSBGS). Figure 4 (right) shows the sizes of the 20 largest user clusters of the 3 methods. Intuitively, the Gibbs samplers tend to assign the outliers to new clusters. Second, we compute the rand index (0-1) of the clustering results of the methods, the values are 0.8071 between CRPGS and TSBMF, 0.8221 between TSBGS and TSBMF, which demonstrates the similarity of the clustering results.

Table 3 gives the movies with highest posterior probability in the 6 largest clusters generated from TSBMF. In **cluster 1** most movies are very new and popular (the data set was collected from September 1997 through April 1998). Also they tend to be action and thriller movies. **Cluster 2** includes many old movies, or movies produced by the non-USA countries. They tend to be drama movies. **Cluster 3** contains many comedies. In **cluster 4** most movies include relatively serious themes. Overall we were quite surprised by the good interpretability of the clusters. Figure 5 (top) shows the relative frequency coefficient (RFC) of the attribute Genre in these movie clusters. RFC of a genre s in a cluster k is calculated as $(f_{k,s} - \overline{f_s})/\sigma_s$, where $f_{k,s}$ is the frequency of the genre s in the movie cluster k, $\overline{f_s}$ is mean frequency, and σ_s is standard deviation of frequency. The labels for each cluster specify the dominant genres in the cluster. For example, action and thriller are the two most frequent genres in cluster 1. In general, each cluster involves several genres. It is clear that the movie clusters

are related to, but not just based on, the movie attribute Genre. The clustering effect depends on both movie attributes and user ratings. Figure 5 (middle) shows RFC of the attribute Occupation in user clusters. Equivalently, the labels for each user cluster specify the dominant occupations in the cluster. The correlation (COR) between user clusters and movie clusters is illustrated as Figure 5 (bottom). It is computed as the probability of positive ratings in the combination of a user cluster k and a movie cluster ℓ, $COR_{k,l} = N^+_{k,\ell}/(N^+_{k,\ell} + N^-_{k,\ell})$, where $N^+_{k,\ell}$ and $N^-_{k,\ell}$ denote the numbers of positive and negative ratings between the user cluster k and the movie cluster ℓ. The darker the cell is, the more likely the members in the user cluster like the members in the movie cluster. The interesting phenomenon is the column 3 about the correlations of the movie cluster 3. Comedy is one of the main genres in movie cluster 3. One would assume that comedies are well liked, but the learned positive probability (the 3rd column in Figure 5 (bottom)) is not very high. It turns out that comedy movies are indeed not so popular in this data set, their positive rating probability is only 37.04%, which is less than average (42.21%).

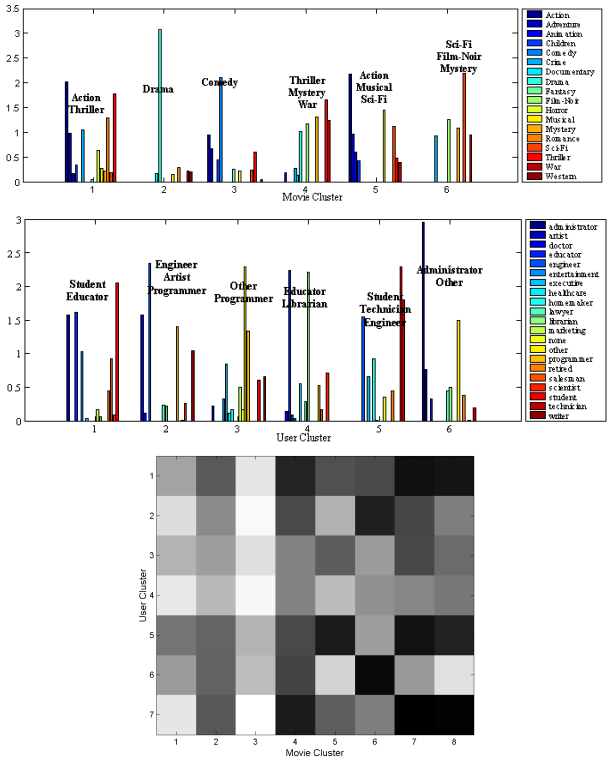

Figure 5: Top: The relative frequency coefficient of the attribute Genre in different movie clusters. Middle: The relative frequency coefficient of the attribute Occupation in different user clusters. Bottom: The correlation between user clusters and movie clusters. The darker the cell is, the more likely the members in the user cluster like the members in the movie cluster.

Note that in the experiments we predicted a relationship attribute R indicating the rating of a user for a movie. The underlying assumption is that in principle anybody can rate any movie, no matter whether that person has watched the movie or not. If the latter is important, we could introduce an additional attribute Exist to specify if a user actually watched the movie. The relationship R would then only be included in the probabilistic model if the movie was actually watched by a user.

5.2 Monastery Data

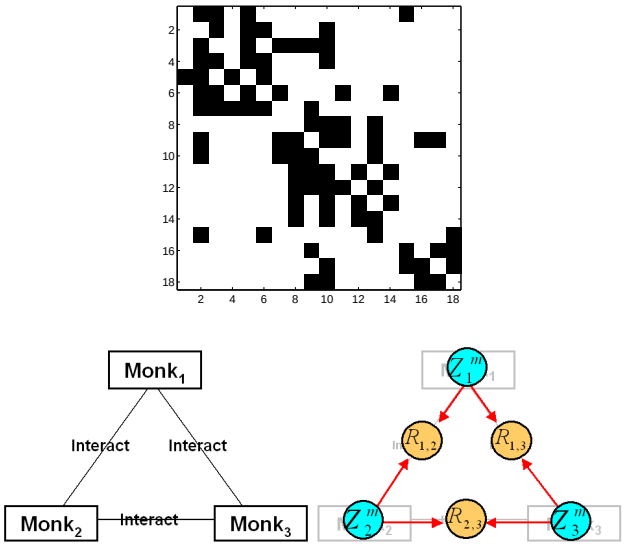

Figure 6: Top: The matrix about interactions between Monks. Left: A sociogram for three monks. Right: The IHRM model for the monastery sociogram.

The second experiment is performed on Sampson's monastery dataset [21]. Sampson surveyed social relationships between 18 monks in an isolated American monastery. The relationships between monks included esteem/disesteem, like/dislike, positive influence/negative influence, praise and blame. Breiger et al. [6] summarized these relationships and yielded a single relationships matrix, which reflected interactions between monks, shown as Figure 6 (top).

After observing the monks in the monastery for several months, Sampson provided a description of the factions among the monks: the loyal opposition (Peter, Bonaventure, Berthold, Ambrose and Louis), the young turks (John Bosco, Gregory, Mark, Winfrid, Hugh, Boniface and Albert) and the outcasts (Basil, Elias and Simplicius). The other three monks (Victor, Ramuald and Amand) wavered between the loyal opposition and the young turks, and were identified as the fourth group, the waverers. Sampson's observations were confirmed by the event that the young turks group resigned after the leaders of the group (John Bosco and Gregory) were expelled over religious differences. The task of the experiment is to cluster the actors.

Figure 6 left shows a sociogram with 3 monks. The IHRM model for the monastery network is illustrated as Figure 6 right. There is one auxiliary hidden variable for each monk. The relationships between monks are conditioned on the hidden variables of the involved monks. The mean field method is used for inference. We initially assume that each monk is in his own cluster. After convergence, the cluster number is optimized as 4, which is exactly the same number of the groups that Sampson identified. The clustering result

Table 3: The largest movie clusters generated by TSBMF on MovieLens data.

Cluster 1	Cluster 2
Independence Day (1996) Truth About Cats and Dogs (1996) Scream (1996) Top Gun (1986) Ransom (1996) Sleepless in Seattle (1993) Phenomenon (1996) Birdcage (1996) Star Trek IV (1986) Mission Impossible (1996) Mrs. Doubtfire (1993) Twister (1996) Starship Troopers (1997) Courage Under Fire (1996) Clear and Present Danger (1994) While You Were Sleeping (1995) Ghost (1990) Sabrina (1995) That Thing You Do (1996) My Best Friend's Wedding (1997)...	A Fish Called Wanda (1988) English Patient (1996) Stand by Me (1986) Leaving Las Vegas (1995) Butch Cassidy and the Sundance Kid (1969) Young Frankenstein (1974) Chasing Amy (1997) Groundhog Day (1993) Willy Wonka and the Chocolate Factory (1971) Full Metal Jacket (1987) E.T. the Extra-Terrestrial (1982) Monty Python's Life of Brian (1979) Contact (1997) Dances with Wolves (1990) Jaws (1975) When Harry Met Sally (1989) Blues Brothers (1980) ...
Cluster 3	**Cluster 4**
Volcano (1997) Cable Guy (1996) Down Periscope (1996) Jungle2Jungle (1997) Waterworld (1995) Batman Returns (1992) Chain Reaction (1996) Multiplicity (1996) Sgt. Bilko (1996) Phantom (1996) Broken Arrow (1996) Vegas Vacation (1997) Nine Months (1995) Murder at 1600 (1997) Escape from L.A. (1996) Net (1995) Wolf (1994) Mimic (1997) McHale's Navy (1997) Dante's Peak (1997)...	Fargo (1996) Godfather (1972) Amadeus (1984) Blade Runner (1982) Casablanca (1942) To Kill a Mockingbird (1962) Rear Window (1954) Das Boot (1981) Citizen Kane (1941) North by Northwest (1959) It's a Wonderful Life (1946) Vertigo (1958) Monty Python and the Holy Grail (1974) Manchurian Candidate (1962) Chinatown (1974) Secrets and Lies (1996) Usual Suspects (1995) Lawrence of Arabia (1962) Schindler's List...
Cluster 5	**Cluster 6**
Indiana Jones a nd the Last Crusade (1989) Return of the Jedi (1983) Fugitive (1993) Sound of Music (1965) Back to the Future (1985) Beauty and the Beast (1991) Hunt for Red October (1990)	A Clockwork Orange (1971) Dr. Strangelove or: How I Learned to Stop Worrying and Love the Bomb (1963) Pulp Fiction (1994) Maltese Falcon (1941)

Table 4: Clustering result of the IHRM on Sampson's monastery data.

Cluster	Members
1	Peter, Bonaventure, Berthold, Ambrose, Louis, Victor, Ramuald
2	John, Gregory, Mark, Winfrid, Hugh, Boniface, Albert
3	Basil, Elias, Simplicius
4	Amand

of the IHRM is shown as Table 4. It is quite close to the real groups. **Cluster 1** corresponds to the loyal opposition. **Cluster 2** is the young turks, and **cluster 3** is the outcasts. The waverers are split. Amand is assigned to **cluster 4**, Victor and Ramuald are assigned to the loyal opposition. Actually, previous research analysis has questioned the distinction of the waverers, e.g., [6, 11] clustered Victor and Ramuald into the loyal opposition, which coincides with the result of the IHRM.

6. RELATED WORK

Some research efforts have been made on nonparametric approaches for relational learning. The work on infinite relational model (IRM) [16] is similar to the IHRM, and they have been developed independently. One difference is that the IHRM can specify any reasonable probability distribution for an attribute given its parent, whereas the IRM would model an attribute as a unary predicate, i.e. would need to transform the conditional distribution into a logical binary representation. Aukia et al. also develop a DP mixture model for large networks [4]. The model associates an infinite-dimensional hidden variable for each link (relationship), and then the objects involved in the link are drawn from a multinomial distribution conditioned on the hidden variable of the link. The model is applied to the community web data with promising experimental results. The latent mixed-membership model [1] can be viewed as a generalization of LDA model on relational data. Although it is not nonparametric, it exploits hidden variables to avoid the extensive structure learning and provides a principled way to model the relational networks. The model associates each object with a membership probability-like vector. For each relationship, cluster assignments of the involved objects are generated with respect to their membership vectors, and then the value of the relationship is conditioned on the cluster assignments.

There are some other important SRL research works for complex relational networks. The probabilistic relational model (PRM) with class hierarchies [9] specializes distinct probabilistic dependency for each subclass, and thus obtains refined probabilistic models for relational data. Taskar et al. explore a classification/clustering relational model, which associates a finite-dimensional latent variable with each object. The probabilistic dependency can be learned from the data or be specified in advance. A group-topic model for text mining is proposed in [25]. It jointly discovers the latent groups in a network as well as the latent topics of events between objects. The latent group model in [18] introduces two latent variables c_i and g_i for an object, and c_i is conditioned on g_i. The object attributes depends on c_i and relations depend on g_i of the involved objects. The limitation is that only relations between members in the same group are considered. These models demonstrate good performance in certain applications. However, most are restricted to domains with simple relationships.

7. CONCLUSIONS

We explored a nonparametric relational model IHRM for social network modeling and analysis. The IHRM enables expressive knowledge representation of social networks and allows for flexible probabilistic inference without the need for extensive structural learning. The IHRM can be applied to community discovery and product recommendation. The empirical analysis on social network data showed encouraging results. We analyzed the cluster structure discovered

in the experiments and found interpretable clusters for the objects. For example, the clusters learned from Sampson's monastery dataset are quite close to the real groups, and coincide with the results of previous research work. For the future work, it will be interesting to explore even more complex relational structures in social network systems, such as domains including hierarchical class structures (ontologies) or on dynamic domains.

8. ACKNOWLEDGMENTS

This research work was sponsored by the German Federal Ministry of Economy and Technology (BMWi) project THESEUS, and by the EU FP7 project LarKC.

9. REFERENCES

[1] E. M. Airoldi, D. M. Blei, E. P. Xing, and S. E. Fienberg. A latent mixed-membership model for relational data. In *Proc. ACM SIGKDD Workshop on Link Discovery*, 2005.

[2] D. Aldous. Exchangeability and related topics. In *Ecole d'Ete de Probabilites de Saint-Flour XIII 1983*, pages 1–198. Springer, 1985.

[3] G. Antoniou and F. van Harmelen. *A Semantic Web Primer*. MIT Press, 2004.

[4] J. Aukia, S. Kaski, and J. Sinkkonen. Inferring vertex properties from topology in large networks. In *Proc. NIPS'07 workshop on statistical models of networks*, 2007.

[5] D. Blei and M. Jordan. Variational inference for dp mixtures. *Bayesian Analysis*, 1(1):121–144, 2005.

[6] R. L. Breiger, S. A. Boorman, and P. Arabie. An algorithm for clustering relational data with applications to social network analysis and comparison to multidimensional scaling. *Journal of Mathematical Psychology*, 12, 1975.

[7] S. Dzeroski and N. Lavrac, editors. *Relational Data Mining*. Springer, Berlin, 2001.

[8] L. Getoor, N. Friedman, D. Koller, and A. Pfeffer. Learning probabilistic relational models. In S. Dzeroski and N. Lavrac, editors, *Relational Data Mining*. Springer-Verlag, 2001.

[9] L. Getoor, D. Koller, and N. Friedman. From instances to classes in probabilistic relational models. In *Proc. ICML 2000 Workshop on Attribute-Value and Relational Learning*, 2000.

[10] L. Getoor and B. Taskar, editors. *Introduction to Statistical Relational Learning*. MIT Press, 2007.

[11] M. S. Handcock, A. E. Raftery, and J. M. Tantrum. Model-based clustering for social networks. *Journal of the Royal Statistical Society*, 170, 2007.

[12] T. Hofmann and J. Puzicha. Latent class models for collaborative filtering. In *Proc. 16th International Joint Conference on Artificial Intelligence*, 1999.

[13] H. Ishwaran and L. James. Gibbs sampling methods for stick breaking priors. *Journal of the American Statistical Association*, 96(453):161–173, 2001.

[14] H. Ishwaran and M. Zarepour. Markov chain monte carlo in approximate dirichlet and beta two-parameter process hierarchical models. *Biometrika*, 87(2):371–390, 2000.

[15] M. I. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul. An introduction to variational methods for graphical models. In M. I. Jordan, editor, *Learning in Graphical Models*, pages 105–161. MIT Press, 1998.

[16] C. Kemp, J. B. Tenenbaum, T. L. Griffiths, T. Yamada, and N. Ueda. Learning systems of concepts with an infinite relational model. In *Proc. 21st Conference on Artificial Intelligence*, 2006.

[17] R. Neal. Markov chain sampling methods for dirichlet process mixture models. *Journal of Computational and Graphical Statistics*, 9:249–265, 2000.

[18] J. Neville and D. Jensen. Leveraging relational autocorrelation with latent group models. In *Proc. 4th international workshop on Multi-relational mining*, pages 49–55, New York, USA, 2005. ACM Press.

[19] L. D. Raedt and K. Kersting. Probabilistic logic learning. *SIGKDD Explor. Newsl.*, 5(1):31–48, 2003.

[20] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl. Grouplens: An open architecture for collaborative filtering of netnews. In *Proc. of the ACM 1994 Conference on Computer Supported Cooperative Work*, pages 175–186. ACM, 1994.

[21] F. S. Sampson. *A Novitiate in a Period of Change: An Experimental and Case Study of Social Relationships*. PhD thesis, 1968.

[22] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Application of dimensionality reduction in recommender systems–a case study. In *Proc. ACM WebKDD Workshop 2000*, 2000.

[23] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Riedl. Analysis of recommender algorithms for e-commerce. In *Proc. ACM E-Commerce Conference*, pages 158–167. ACM, 2000.

[24] J. Sethuraman. A constructive definition of dirichlet priors. *Statistica Sinica*, 4:639–650, 1994.

[25] X. Wang, N. Mohanty, and A. McCallum. Group and topic discovery from relations and text. In *Proc. 3rd international workshop on Link discovery*, pages 28–35. ACM, 2005.

[26] Z. Xu, V. Tresp, K. Yu, and H.-P. Kriegel. Infinite hidden relational models. In *Proc. 22nd UAI*, 2006.

[27] J. Yedidia, W. Freeman, and Y. Weiss. Constructing free-energy approximations and generalized belief propagation algorithms. *IEEE Transactions on Information Theory*, 51(7):2282–2312, 2005.