

Friends Recommendation with Rating Side Information

Xiang Hu^{1,2,*}, Wengdong Wang¹, Xiangyang Gong¹, Bai Wang¹, Xirong Que¹ and Hongke Xia³

¹State Key Laboratory of Network and Switching, Beijing University of Posts and Telecommunications, Beijing, 100876, P.R. China; ²School Of Control And Computer Engineering, North China Electric Power University, Beijing, 102206, P.R. China; ³Computer School, Beijing Information Science & Technology University, Beijing, 100101, P.R. China

Abstract: With the broad application of web 2.0 technology, various kinds of online social networks arise at present. The emergence of social networks not only helps the public to facilitate the sharing and communicating, but also helps them to know more new friends from cyberspace social circle. Therefore, friends recommendation becomes a critical function for various online social networks. Problem of friends recommendation is actually the problem of link prediction in essential, and vast majority of the solutions are traditionally based on the social relation, yet common interests are also the important factor of forming friendships in real life besides social relations. This article proposes a new matrix factorization based method for friends recommendation. The proposed method considers two data sources of social relations and interest ratings simultaneously, utilizes Gaussian kernel to capture interest relevance (interest similarity) between persons, makes use of Gaussian process to generate the users' profile vectors, and eventually achieves a recommendation method having the capability of commending friends with common interests. Experiments show that our method outperforms the art-of-the-state traditional link prediction methods using only social relations.

Keywords: Friends recommendation, gaussian process, gaussian kernel, matrix factorization.

1. INTRODUCTION

The emergence of online social network (OSN) dramatically stirs up a significant reform in procedure of information creation, propagation and consumption. OSN has brought the public a new style of social lives parallel to our daily activities in the real life. Popular social network sites, such as Douban¹ and Sina weibo², have already collected millions of extensively distributed users and are still attracting thousands of passionate freshmen each day. Doubtlessly, online social networks have become a major platforms for building friendships and sharing interesting things.

In OSN, most users' friends come from those in real life, such as schoolmates, colleagues and so on. As more and more users interact with real friends on the Web, personalized friend recommendation service should be provided by OSN, which can not only help user find potential friends with similar interests but also enhance their loyalties to the OSN and therefore boost the traffics of the OSN. Research of [1, 2] shows that social phenomenon of homophily governs the users' behaviors in the OSN. The homophily means that people with similar interests tend to contact to each other and are more likely to be friends. For instance, a soccer fan may want to add other local soccer fans to be friends for watching FIFA World Cup together, and researchers with similar academic interests more tend to build up friend relationship. The homophily also means that friends in a small social circle are more likely to make similar decisions when

purchasing goods or services. For example, colleagues in the same office incline to prefer the same brand of computer. Though almost large part of major OSNs provide friends recommendation service and item recommendation service, very small part of them make full use of homophily phenomenon to enhance their systems so as to recommend potential friends with common interests or recommend items considering their social context.

This article mainly focuses on the problem of friends recommendation in context of OSN. Friends recommendation is also known as problem of link prediction [3, 4], which is the problem of predicting the presence or absence of edges between nodes of a graph. In the OSN circumstance, on the one hand link prediction is confronted with problem of data sparsity, which means a user only make friends with very little fraction of available users, on the other hand OSNs routinely collect huge social relations among users and interactions (e.g. ratings) between uses and items, and it craves for new methods to recommend friends with exploiting the aided interactions information. However, most of traditional link prediction algorithms are merely based on the structure of social relations, and either ignore the side information related to every users, or have no capability of exploiting side information. So traditional link prediction algorithms are suffering from the problem of sparse data, yet making poor use of those rich and useful side information at the same time.

To overcome such limitations, this paper has proposed a Gaussian Process based Probabilistic Matrix Factorization (GPPMF) for friends recommendation. GPPMF models the social adjacent matrix as the product of two profile matrices which are sampled from two different zero-mean Gaussian

processes (GP), and incorporates the rating side information over the rows and columns of the profile matrices through Gaussian kernel. The covariance kernel functions of the GPs are derived from the rating side information, and encapsulate the covariance structure between rows and between columns respectively. This article takes care for deriving the covariance function from vectorial side information, for example: all of the ratings a special user has marked on his favorite items form a vectorial data denoting the interest feature of the users, however we can also incorporate other type of side information (e.g. graph or tree) via various covariance kernel functions according to the specific domain problem.

Although GPPMF is highly related to PMF [5] and its Bayesian version BPFM [6], there are two crucial difference between them: PMF and BPFM put the Gaussian Prior on each row/column of profile matrix, while GPPMF puts a Gaussian process prior across all row/column of profile matrix; GPPMF is capable of capturing the relevance between rows/columns of profile matrix, while PMF and BPFM can't do that. From the perspective of algorithm framework, GPPMF is similar to KPMF [7], while they are obviously different in the data type of side information and the corresponding covariance kernel functions.

The contributions of this work exists in three aspects:

1) We propose a Gaussian process based probabilistic matrix factorization model for recommending friends with common interests in the OSN circumstance.

2) We use covariance kernel functions to capture the interests relevance between users, and inject interests relevance into our model.

3) We derive a gradient descent based methods for training our model.

The remainder of this paper is organized as follows: Section 2 introduces some related works, Section 3 describes the problem in state, Section 4 and Section 5 propose the new GPPMF model, Section 6 shows the experimental result and analysis, and the last Section gives the conclusion.

2. RELATED WORKS

Friends recommendation is a problem of link prediction in essential. There are two type of link prediction: (i) structural, where the input is partially observed graph, and the target is to predict whether the edges between nodes in unobserved graph exist or not, and (ii) dynamic, where the input is the edges appeared before the given time, and the task is to predict what edges will appear after this time. Friends recommendation in this article will focus on the structural link prediction.

Existing link prediction methods can be divided into two classes: unsupervised and supervised. Most unsupervised methods are similarity-based methods, where non-observed links are ranked according to their scores, and the links connecting more similar nodes are considered to be of higher existence likelihoods [8]. According to the measurement method, the similarity-based methods can be classified into local method, global method and quasi-local method [8]. Local method merely consider finding friends in the 2-hop network neighborhood. It has been proved that the quasi-

local method, e.g., Local Path (LP), perform remarkably better than the local method, such as Resource Allocation (RA), Adamic-Adar (AA). Although globe method, such as Katz method and Random Walk with Restart (RWR), has better performance than local or quasi-local method in some application scenario, they are infeasible for large-scale networks because of prohibitive computational costs.

Unsupervised methods depend on the topological information so that they haven't capability of exploiting side information related to the nodes or edges. Unsupervised methods are invariant to the specific topology of the network, and thus do not involve any learning process. While supervised methods, on the other hand, attempt to predict the links by learning a parameter vector θ via:

$$\arg \min_{\theta} \frac{1}{|\mathcal{O}|} \sum_{(i,j) \in \mathcal{O}} \ell(G_{i,j}, \hat{G}_{i,j}(\theta)) + \Omega(\theta) \quad (1)$$

where $\hat{G}_{i,j}(\theta)$ is the model's predicted score for the pair of nodes (i, j) , $\ell(\cdot)$ is a loss function, and $\Omega(\cdot)$ is a regularization term guarding against overfitting. Different choices of these terms will form different models, can also blend side information in the models in different ways.

Here we summarize three classes of popular approaches:

- Latent feature models. These models model the link prediction as matrix completion problem and factorize $G \cup L(U\Lambda U^T)$, where $U \in \mathbb{R}^{n \times k}$, $\Lambda \in \mathbb{R}^{k \times k}$ and $L(\cdot)$ is link function. Each node i has a corresponding latent feature vector u_i . The predictive function can be written as $\hat{G}_{i,j} = L(u_i \Lambda u_j^T)$, and regularization term is usually $\Omega(U, \Lambda) = \frac{\lambda_U}{2} \|U\|_F^2 + \frac{\lambda_\Lambda}{2} \|\Lambda\|_F^2$. Such models (e.g. PMF [5]) and their Bayesian (e.g. BPFM [6]) version have been applied in collaborative filtering [9] and political science [10], but they haven't been extensively studied in the link prediction literature.
- Graph regularization models. These models assume that each node i has an associated feature vector $x_i \in \mathbb{R}^d$, and a kernel $K_{i'i'j'j'}$ can be constructed to compare the node pair (i, j) and (i', j') based on the nodes' associated feature vectors. The predicted matrix \hat{G} is computed by constraining the values in matrix \hat{G} should vary smoothly in term of kernel K , where K play the role of graph regularizer. This approach is a popular semi-supervised learning method [11], and is also called link propagation [12]. According to the framework of Eq. (1), we have:

$$\Omega(\hat{G}) = \frac{\lambda}{2} \sum_{i,i',j,j'} K_{i'i'j'j'} (\hat{G}_{i,j} - \hat{G}_{i',j'})^2 + \frac{\mu}{2} \sum_{(i,j) \in \mathcal{O}} \hat{G}_{i,j}^2$$

K is pre-defined and used to integrate side information (feature vectors of nodes) into link prediction.

- Feature-based models. These models suppose that each node i has an associated feature vector $x_i \in \mathbb{R}^d$, and also each pair of nodes (i, j) has an associated feature vector $z_{ij} \in \mathbb{R}^D$. Then these models can be described by a formula instantiated from Eq. (1):

$$\hat{G}_{i,j}(w, v) = L(f_P(z_{ij}; w) + f_N(x_i, x_j; v)),$$

where $f_P(\cdot), f_N(\cdot)$ are functions on the node pairs' and nodes' feature vector respectively, and $L(\cdot)$ is link function.

$f_P(\cdot)$ and $f_N(\cdot)$ may be linear [3] or nonlinear [13, 14], and the corresponding regularizer typically is $\frac{\lambda_w}{2} \|w\|^2 + \frac{\lambda_v}{2} \|v\|^2$.

Although link prediction problem has been researched for a time, few researches [2] can be directly applied to recommend friendships with rating side information in application scenario of OSN. Our works is enlightened by PMA [15] and KPMF, which make use of two GPs to generate the profile matrices of factorization. Different from traditional PMF or BPMF, PMA and KPMF generate the profile matrices across all rows/columns rather than along the rows/columns. By this way, the covariance structure between rows/columns, which is encoded by covariance kernel functions on the side information, can be injected into the generative procedure of profile matrices. The main difference is that PMA generate the data matrix from the addition of the profile matrices while KPMF generate the data matrix from the product of the profile matrices. The proposed model in this paper takes rating data as side information, derives the covariance structure between rows/columns of profile matrices on the vectorial rating data, and generates the predictive adjacent matrix from the product of the profile matrices.

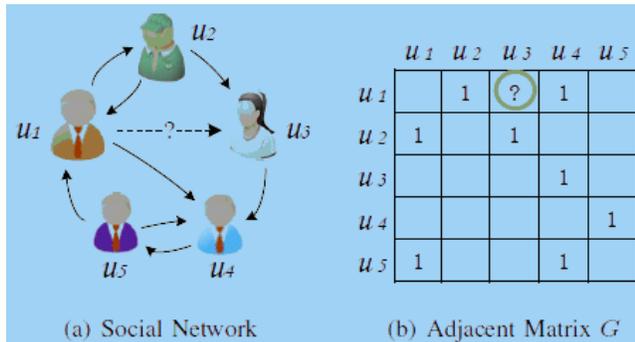


Fig. (1). Description of friends recommendation problem.

3. PROBLEM DISCRPTION

Social network can be represented by a graph $\mathbb{G}(\mathbb{U}, \mathbb{E})$, where \mathbb{U} is the set of nodes denoting persons and \mathbb{E} is the set of edges denoting social relations. Social network can also be represented by adjacent matrix $G = [G_{i,j}]_{M \times M}$,

where $G_{i,j}$ indicates whether there exists social relation between user i and j . If the relation exists then $G_{i,j} = 1$, otherwise $G_{i,j} = 0$. The friend recommendation problem, which is shown as Fig. (1), is that recommending the most possible friends for the special user. Recommendation system contains a set of users $\mathbb{U} = \{u_1, u_2, \dots, u_M\}$, a set of items $\mathbb{I} = \{i_1, i_2, \dots, i_N\}$, and rating matrix $R = [R_{u,i}]_{M \times N}$ gathering all the rating scores users marked. In the context of OSN, persons in social networks are also the users in the recommendation system. Therefore the social network of users may be denoted by $\mathbb{G}(\mathbb{U}, \mathbb{E})$, where each node in \mathbb{U} represents a user in recommendation system.

Different from the traditional friend recommendation task, the task of this works arms to recommend the most possible friends for the special user with exploiting both his social relations and his rating side information. If we use G and R to describe the social friendship relations and interests of users respectively, it needs to exploit G and R so as to recommend each user the most possible new friends with similar interests. The procedure of recommendation consider not only the social friendship relations but also the common interests between users.

3.1. Notations

In order to facilitate the formalization and deduction of the proposed model and related models, we introduce the main notations which will be used in the paper as follows:

Table 1. The explanation of notations.

Notation	Explanation
R	$N \leftrightarrow M$ data matrix
$R_{i,:}$	i^{th} row of R
$R_{:,j}$	j^{th} column of R
N	Number of rows
M	Number of columns
D	Dimension of latent factors
U	$N \leftrightarrow D$ latent matrix for rows
V	$M \leftrightarrow D$ latent matrix for columns
$U_{i,:} \in \mathbb{R}^D$	Latent factors for i^{th} row
$V_{j,:} \in \mathbb{R}^D$	Latent factors for j^{th} column
$U_{:,d} \in \mathbb{R}^N$	d^{th} latent factor for all rows
$K_U \in \mathbb{R}^{N \times N}$	Covariance matrix for rows
$S_U \in \mathbb{R}^{N \times N}$	Inverse of K_U
$[n]_i^N$	$n = \{1, 2, \dots, N\}$

3.2. Probabilistic Matrix Factorization (PMF)

As an effective and high efficient model, Probabilistic Matrix Factorization takes the item recommendation problem as matrix completion problem of rating matrix R , and approaches this problem from matrix factorization aspect. The friend recommendation problem can also be similarly treated as the matrix completion problem of adjacent matrix G , so here we discuss the PMF model firstly with taking the item recommendation as the application context.

The PMF model uses two latent matrices $U_{N \times D}$ and $V_{M \times D}$ to capture the row and column features of rating matrix R respectively, and generates the rating matrix R as follows:

- 1) For each row i in R , $[i]_1^N$, generate latent factor from a multivariate normal distribution with zero-mean, that is $U_{i,:} \sim \mathcal{N}(\mathbf{0}, \sigma_U^2 \mathbf{I})$, where \mathbf{I} denoted an $N \leftrightarrow N$ identity matrix.
- 2) For each column j in R , $[j]_1^M$, generate latent factor from a multivariate normal distribution with zero-mean, that is $V_{j,:} \sim \mathcal{N}(\mathbf{0}, \sigma_V^2 \mathbf{I})$.
- 3) For each non-missing entries $R_{i,j}$, generate $R_{i,j}$ from a univariate normal distribution with mean decided by inner product of $U_{i,:}$ and $V_{j,:}$, that is $R_{i,j} \sim \mathcal{N}(U_{i,:} V_{j,:}^T, \sigma^2)$.

The graphical model of PMF can be shown as Fig. (2a). From the generative process of PMF, the log-posterior over the latent matrices U and V is given by:

$$\begin{aligned} \log p(U, V \mid \mathbf{Q}, \sigma^2, \sigma_U^2, \sigma_V^2) &= -\frac{1}{2\sigma^2} \sum_{i=1}^N \sum_{j=1}^M \delta_{i,j} (R_{i,j} - U_{i,:} V_{j,:}^T)^2 \\ &\quad - \frac{1}{2\sigma_U^2} \sum_{i=1}^N U_{i,:}^T U_{i,:} - \frac{1}{2\sigma_V^2} \sum_{j=1}^M V_{j,:}^T V_{j,:} \\ &\quad - \frac{1}{2} (A \log \sigma^2 + ND \log \sigma_U^2 + MD \log \sigma_V^2) + C \end{aligned} \tag{2}$$

where $\delta_{i,j}$ is an indicator, and its value is 1 if $R_{i,j}$ is non-missing, and 0 otherwise. A is the amount of non-missing entries in rating matrix R , and C is a constant that is not relevant to the latent matrices U and V . By maximizing the log-posterior according to U and V , the appropriate matrices U and V can be obtained, and then they can be used to predict the missing entries in R .

As the Bayesian counterpart of PMF, BPMPF [6] introduces a full prior for each $U_{i,:}$ and $V_{j,:}$. Each $U_{i,:}$ (and similarity for $V_{j,:}$) is sampled from $\mathcal{N}(\mu_U, \Sigma_U)$ with hyper-

parameters $\{\mu_U, \Sigma_U\}$, and the hyperparameters are further sampled from conjugate Gaussian-Whishart priors.

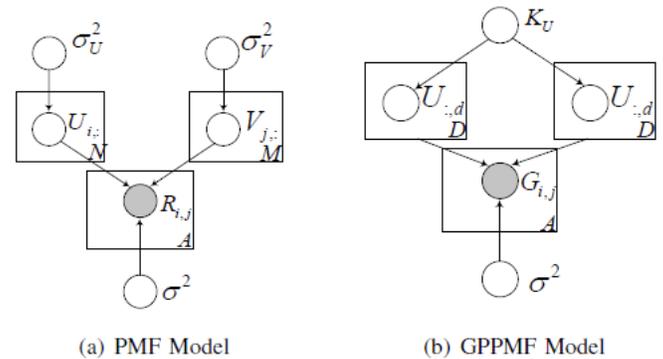


Fig. (2). Graphic model of PMF and GPPMF.

4. GPPMF MODEL

Although PMF and BPMPF are powerful, they lack the capability of utilizing side information. In this section, we describe the proposed Gaussian Process based Probabilistic Matrix Factorization (GPPMF) model and present the gradient descent method for training the model.

4.1. Element Specification of GPPMF

We treat friend recommendation problem as matrix completion with $G \cup U^T U$ where both column and row profile matrix are U , and each column of U (denoted as $U_{:,d} \mid \sim \mathcal{N}$) is generated by a Gaussian process. As a generation of the multivariate Gaussian distribution, Gaussian process $\mathcal{GP}(m(x), k(x, x'))$ is determined by a mean function $m(x)$ and a covariance function $k(x, x')$. In our model, x and x' denote the row and column index of adjacent matrix G respectively. Without loss of generality, our model let $m(x) = 0$ and $k(x, x')$ be the vectorial kernel function measuring the interest similarities among users according to the vectorial rating side information in matrix R .

Given that $K_U \mid \sim \mathcal{N}^{N \times N}$ denotes the full covariance matrix for row/column of G and is known, the generative process of GPPMF is described as follows:

- 1) For each column of U , generate $U_{:,d} \sim \mathcal{GP}(\mathbf{0}, K_U)$, $[d]_1^D$.
- 2) For each non-missing entry $G_{i,j}$ (total number is A), generate $G_{i,j} \sim \mathcal{N}(U_{i,:} U_{j,:}^T, \sigma^2)$, where σ is a constant.

Given latent matrix U , the likelihood over the non-missing entries in the adjacent matrix G is:

$$p(R \mid \mathbf{0}; \mathbf{G}) = \prod_{i=1}^N \prod_{j=1}^M \mathcal{N}(G_{i,j} \mid U_{i,:} U_{j,:}^T, \sigma^2)^{\delta_{i,j}}, \tag{3}$$

and the prior over the U is given by:

$$p(U | \mathbf{G}, \mathbf{R}, \mathbf{K}_U) = \prod_{d=1}^D \mathcal{GP}(U_{:,d} | 0, K_U), \quad (4)$$

According to Bayesian rule, the posterior over the latent matrices U is given by:

$$p(U | \mathbf{G}, \mathbf{R}, \mathbf{K}_U) \propto p(U | K_U) p(\mathbf{R} | U, \sigma^2),$$

and the log-posterior over the latent matrices U is given by:

$$\begin{aligned} & \log p(U | \mathbf{G}, \sigma^2, K_U) \\ &= -\frac{1}{2\sigma^2} \sum_{i=1}^N \sum_{j=1}^N \delta_{i,j} (G_{i,j} - U_{i,:} U_{j,:}^T)^2 \\ & - \frac{1}{2} \sum_{d=1}^D U_{:,d}^T S_U U_{:,d} - A \log \sigma^2 - \frac{D}{2} \log |K_U| + C \end{aligned} \quad (5)$$

where S_U denotes K_U^{-1} , $|K_U|$ is the determinant of K_U , A is the number of non-missing entries in G , and C is a constant term not relevant to U .

In PMF/BPMF, the U is sampled in a "row-wise" manner, i.e., $U_{i,:}$ is sampled for each row of R . Though elements in $U_{i,:}$ are conditionally dependent, rows in $\{U_{i,:}, [i]_1^N\}$ are conditionally independent and therefore these models can't capture the relevance between the rows of R . In contrast, in GPPMF, each column of the latent matrices $U_{:,d}$ is sampled from a zero-mean Gaussian process [16] and U is sampled in a "column-wise" manner Fig. (2b), also each $U_{:,d}$ is sampled for all rows of G and hence relevance between the rows of G can be captured.

4.2. Learning Method for GPPMF

In order to learning the profile matrix U , we perform a MAP estimate on the Eq.(5) with respect to U , which is equivalent to minimize objective function \mathcal{L} (Eq.(6)) with reversing Eq.(5) and discarding the terms unrelated to U ,

$$\mathcal{L} = \frac{1}{2\sigma^2} \sum_{i=1}^N \sum_{j=1}^N \delta_{i,j} (G_{i,j} - U_{i,:} U_{j,:}^T)^2 + \frac{1}{2} \sum_{d=1}^D U_{:,d}^T S_U U_{:,d}. \quad (6)$$

The local minimum point of \mathcal{L} can be searched by gradient descent, and the gradients are derived as:

$$\frac{f\mathcal{L}}{fU_{i,d}} = -\frac{1}{\sigma^2} \sum_{j=1}^N \delta_{i,j} (G_{i,j} - U_{i,:} U_{j,:}^T) U_{d,j} + \mathbf{e}_{(i)}^T S_U U_{:,d} \quad (7)$$

where $\mathbf{e}_{(i)}$ is an N -dimensional unit vector corresponding to the i^{th} column of identity matrix $\mathbf{I}_{N \times N}$. The iterative update equation for U is:

$$U_{i,d}^{(t+1)} = U_{i,d}^{(t)} - \eta \frac{f\mathcal{L}}{fU_{i,d}} \quad (8)$$

where η is the learning rate to control the speed of learning. Taking the non-missing entries in G as the training data, the algorithm updates U repeatedly until its convergence. It is important to note that even though S_U (the Inverse matrix of K_U) is computational intensive, it is calculated only once for initialization during the iteration.

As stochastic gradient descent (SGD) usually converges much faster than gradient descent, we derive the SGD update equation for GPPMF below. The objective function Eq.(6) could be rewritten as

$$\begin{aligned} \mathcal{L} &= \frac{1}{\sigma^2} \sum_{i=1}^N \sum_{j=1}^N \delta_{i,j} (G_{i,j} - U_{i,:} U_{j,:}^T)^2 + \sum_{i=1}^N \sum_{j=1}^N S_U(i,j) U_{i,:}^T U_{j,:} \\ &= \sum_{i=1}^N \sum_{j=1}^N \delta_{i,j} \left[\frac{1}{\sigma^2} (G_{i,j} - U_{i,:} U_{j,:}^T)^2 + \frac{1}{\tilde{M}_i} U_{i,:}^T \sum_{i'=1}^N S_U(i,i') U_{i',j} \right] \\ &= \sum_{i=1}^N \sum_{j=1}^N \delta_{i,j} \mathcal{L}_{i,j}. \end{aligned} \quad (9)$$

where \tilde{M}_i is the number of non-missing social links in row i . And then, for each non-missing social links (i,j) , taking gradient of $\mathcal{L}_{i,j}$ with respect to $U_{i,:}$ gives:

$$\begin{aligned} \frac{f\mathcal{L}_{i,j}}{fU_{i,:}} &= -\frac{2}{\sigma^2} (G_{i,j} - U_{i,:} U_{j,:}^T) U_{j,:} \\ & + \frac{1}{\tilde{M}_i} \sum_{i'=1}^N S_U(i,i') U_{i',j} + S_U(i,i) U_{i,:}. \end{aligned} \quad (10)$$

4.3. Correction of Unbalanced Training Data

An important challenge for learning latent factors from dyadic matrix is that the matrix are extremely sparse with only positive values observable. That is, for adjacent matrix G , we typically do not observe explicit information that user i does not like user j . Since the missing/unobserved G_{ij} might also imply that user i do not even know about j , the missing social link in G can't be *absolutely* considered as negative information.

At the same time, ignoring the negative information of missing entries will inevitably train a predictor which is overly optimistic with regard to positive entries. To eliminate ill effects of unbalanced training data, we randomly sample a handful set of missing (unobserved) entries $\{G_{i,j'}\}_{j'=1:m}$ for every positive entry G_{ij} in each iteration, and treat entries in $\{G_{i,j'}\}_{j'=1:m}$ (m is number of sampled entries) as negative examples (e.g. $G_{i,j'} = -1$) with probability $1/m$ each, and then those pseudo-negative examples are trained in the same iteration together with their positive counterpart $G_{i,j}$.

Because the procedure of sampling is random, the set of pseudo-negative examples changes at each iteration, and as a

consequence each missing entry is treated as a potentially very weak negative instance.

4.4. Kernels for GPPMF

In Gaussian process $\mathcal{GP}(m(x), k(x, x'))$, kernel function $k(x, x')$ is used to calculate the covariance between x and x' , while in GPPMF model, we measure this covariance by the interest similarity. With taking the missing entries as 0, interest features of user x and x' are described by their historical ratings $R_{x,:}$ and $R_{x',:}$, respectively, and *vectorial* kernels provide a way to measure the interest similarity between those features. In this work, we examine two different vectorial kernels.

- Radial basis kernel function (RBF): A radial basis kernel function is a real-valued function whose value depends on the squared Euclidean distance between the input feature vectors. In this work, the RBF kernel on interest features of user x and x' , represented as $R_{x,:}$ and $R_{x',:}$, respectively, is defined as:

$$k(x, x') = \exp \frac{-(\|R_{x,:} - R_{x',:}\|)^2}{2\sigma^2}$$

where σ is free parameter.

- Sigmoid kernel function (SKF): The sigmoid kernel comes from the neural networks field, and also is known as hyperbolic tangent kernel, is defined as:

$$k(x, x') = \tanh(\alpha R_{x,:} R_{x',:}^T + c).$$

where two adjustable parameters are the slope α and the intercept constant c . A common value for α is $1/M$, where M is the data dimension.

Both of the kernels map the interest features of users into a real-valued similarity in $[0, 1]$, and eventually construct the covariance matrix K_U .

5. EXPERIMENTAL RESULTS AND EVALUATION

5.1. Datasets

Experiments of this work are based on the real Douban dataset³ and Flixster dataset⁴. Douban⁵ is one of greatest online social networks in China, which provides marking scores, comments and recommendation services on movies, music and books and so forth. Users can create friendships with each other through such ways like Email, and they can mark scores on items (such as movies, etc.,) with the values range from 1 to 5. Douban dataset is collected by Chinese University of Hong Kong by crawlers, which contains 129 thousand users, 59 thousand movies, 1.68 million rates and 1.69 million bidirectional friend relations.

Flixster⁶ is a social network site about movies which let users' share their marked scores on movies, discuss new ones, and recognize persons who have similar interests. The Flixster dataset in our experiments is collected by Jamali, which contains 1 million users, 8.2 millions rates, 4.9 million movies and 26.7 million bidirectional friend relations. The rating scores are discrete value arrange in $[0.5, 5]$, and can be divided into 10 levels. The degree distribution of Douban and Flixster is shown in Fig. (3).

5.2. Metrics

When offering friend recommendation service, OSNs usually provide a personal top N recommendation list for every user. The classic metrics for top N recommendation are *recall* and *precision* [17], and this work use them to quantify the accuracy of proposed friend recommendation method. In order to evaluate recall and precision, entire social links (that is the entries in social adjacent matrix G) is split into training set $G_{training}$ and testing set G_{test} . Given $R(u)$ is the top N friends recommendation list of user according to recommendation model on $G_{training}$, and $T(u)$ is the real friend list from G_{test} , the recall and precision can be defined as follows:

- 1) Recall: Given the ranked top N list of the non-observed links is $R(u)$, and the really observed links list is $T(u)$, the recall is defined as the ratio of correctly predicted links to all real observed links.

$$recall = \frac{|R(u) \cap T(u)|}{|T(u)|}$$

- 2) Precision: The precision is defined as the ratio of correctly predicted links to all predicted links.

$$precision = \frac{|R(u) \cap T(u)|}{|R(u)|}$$

Precision can be taken as a measure of exactness or quality, and recall is a measure of completeness or quantity. High precision means that an algorithm returned substantially more relevant results than irrelevant, while high recall means that an algorithm returned most of the relevant results.

5.3. Results and Analysis

There is no representative friend recommendation method taking into account both users' social networks and common interests. To evaluate the effectiveness of our proposed model, this work compares its performance with other two homogeneous methods on two different datasets:

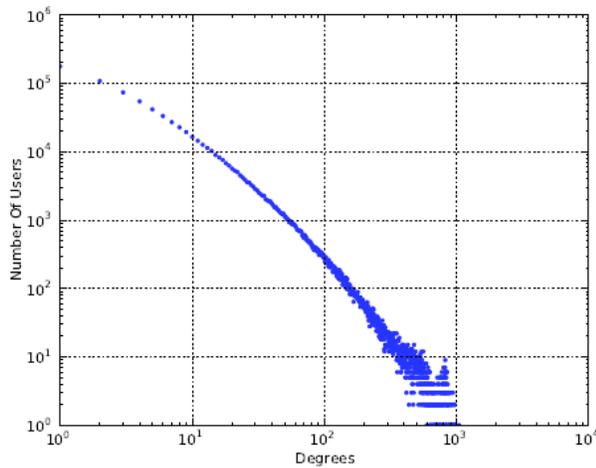
- Local Path Index (LP) [18, 19]: LP is a state-of-the-art unsupervised method for link prediction, and takes consideration of local paths and provides a good tradeoff of accuracy and computational complexity with wider ho-

³ <https://www.cse.cuhk.edu.hk/irwin.king/pb/data/home>

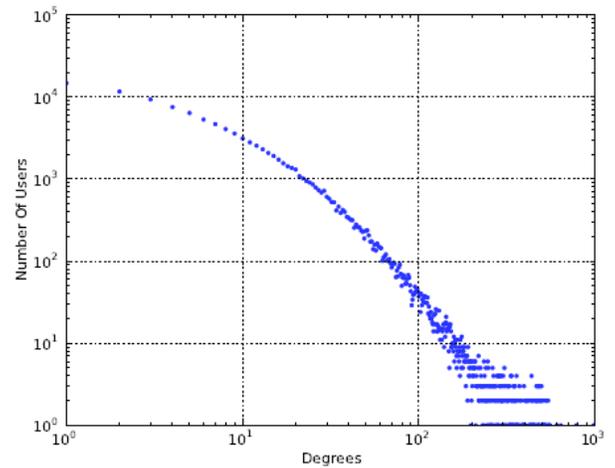
⁴ <http://www.cs.ubc.ca/~jamalim/datasets/>

⁵ <http://www.douban.com.cn>

⁶ <http://www.flixster.com>



(a) Degrees of Flixster



(b) Degrees of Douban

Fig. (3). Degree distribution of datasets.

rather than the most widely used Common Neighbors Index.

- Probability Matrix Factorization (PMF) [5]: PMF is a canonical matrix factorization method, and may also be used to social adjacent matrix for link prediction.

This work designs the experiment to compare the performance of the GPPMF model with the baseline methods. First of all, the experiment randomly draws 20%, 40%, 60%, 80% datasets as training set respectively and the rest are used as testing set, and then *recall* and *precision* are measured on each training set with different models. In order to get stable results, each percentage of datasets is repeated for five times and the average of five results is used as the final result.

In experiments, the dimensionality parameter D is set to 10 as the experimental value in [5], the feature of each user $U_{n,:}$ is initialized with $\mathcal{N}(0, 0.01)$, and learning rate $\eta = 0.01$. To compare the impact of different kernels, we use GPPMF_RBF and GPPMF_SKF to represent GPPMF model with radial basis kernel function and sigmoid kernel function respectively. The ideal parameter values of kernel functions are determined *via* cross validation, and the σ in RBF is set to 4.17 and intercept constant c in Sigmoid function is set to 0.1. The length of recommended friends list is set to 200. Parameters that aren't mentioned here use the default values. The recalls and precisions on different dataset is shown as Fig. (4).

In Figs. (4a and 4b), precisions are measured on Flixster dataset and Douban dataset respectively, and precisions increase with percentage of training set despite of dataset and methods. At each group, GPPMF_RBF and GPPMF_SKF have higher precisions than PMF and LP, GPPMF_RBF has the maximum precision while LP has the minimum one. On account that Douban dataset is denser than Flixster dataset, precisions in Fig. (4b) is slightly higher than those in Fig. (4a). In Figure Fig. (4c), Fig. (4d), recalls increase with percentage of training set similar to precisions, however GPPMF_RBF and GPPMF_SKF have lower recalls than PMF and LP in most cases, also recalls on denser Douban dataset is lower than Flixster dataset.

Taking the percentage of improvement on precision and recall as criterion, we compare the proposed model with PMF in details. The improvements of GPPMF_RBF and GPPMF_SKF in contrast to PMF on experimental dataset are shown as Fig. (5). Figs. (5a and 5b) show that the improvements on precision are positive and decrease with the percentage of training set. Also it can be observed that the improvement of GPPMF_RBF on precision is higher than that of GPPMF_SKF, and it is more obvious in the case of the low percentage of training set. These observations justify that the proposed GPPMF improve the precision of friend recommendation and the effect of improvement is clear especially in context of the sparse social relations, also manifest that RBF kernel function has better effect than sigmoid kernel function on injecting the rating side information into GPPMF. Figs. (5c and 5d) show that the improvements on recall are negative, and tend to be a stable value -5% with the percentage of training set on both dataset. These observations indicate that the proposed model boosts the precisions at the cost of a bit of decreases in recalls with conforming to the universal law that the higher precision usually means the lower recall.

6. CONCLUSION

In this paper, we propose a new matrix factorization based method for friends recommendation with considering two data sources of social relations and interest ratings simultaneously. The proposed method employs Gaussian kernel function to capture interest relevance (interest similarity) between persons, utilize Gaussian process to generate the users' profile vectors, and eventually get a new recommendation method with the capability of recommending social friends with common interests. Experiment shows that our method can promote the precision at the cost of a bit of reduction of recalls in contrast with the art-of-the-state traditional link prediction methods and matrix factorization based methods. With the aid of rating side information, the effects of promotion are more distinct especially when the social relations are sparse. Experiment also shows that RBF kernel function performs better than sigmoid kernel function on injecting the rating side into GPPMF.

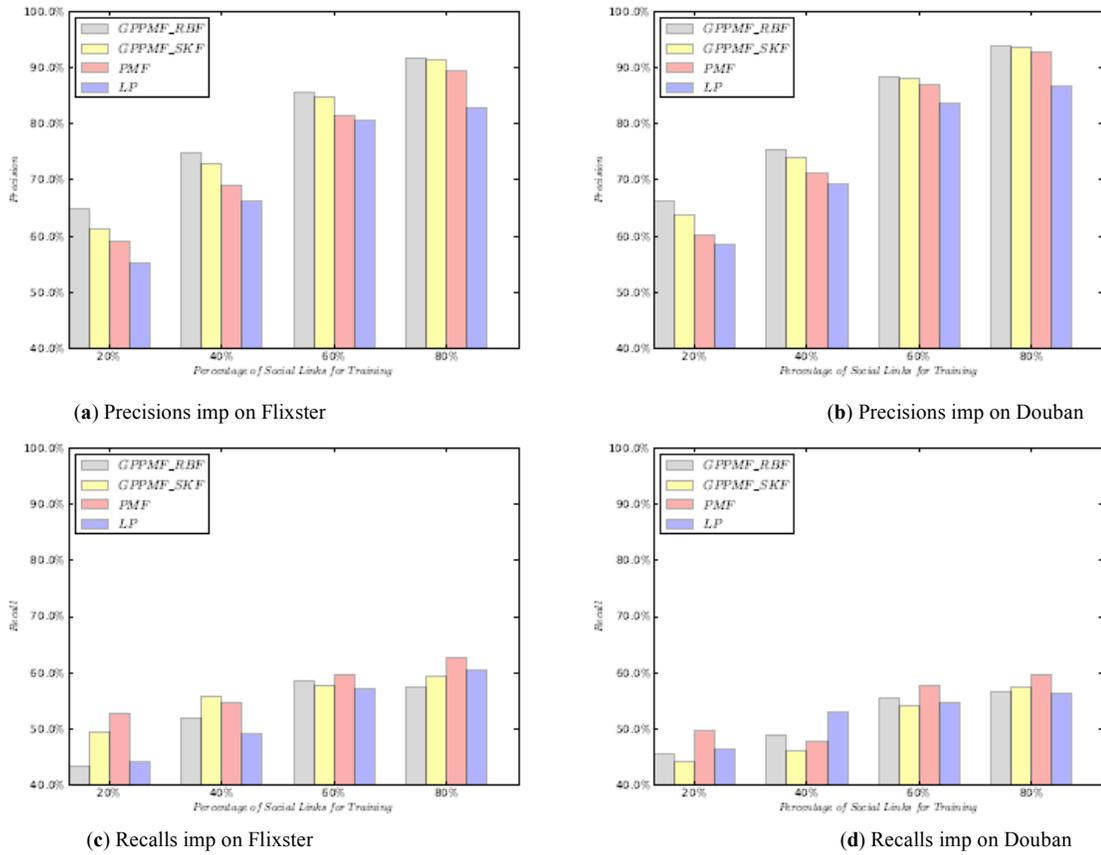


Fig. (4). Recalls and precisions on flixster and douban.

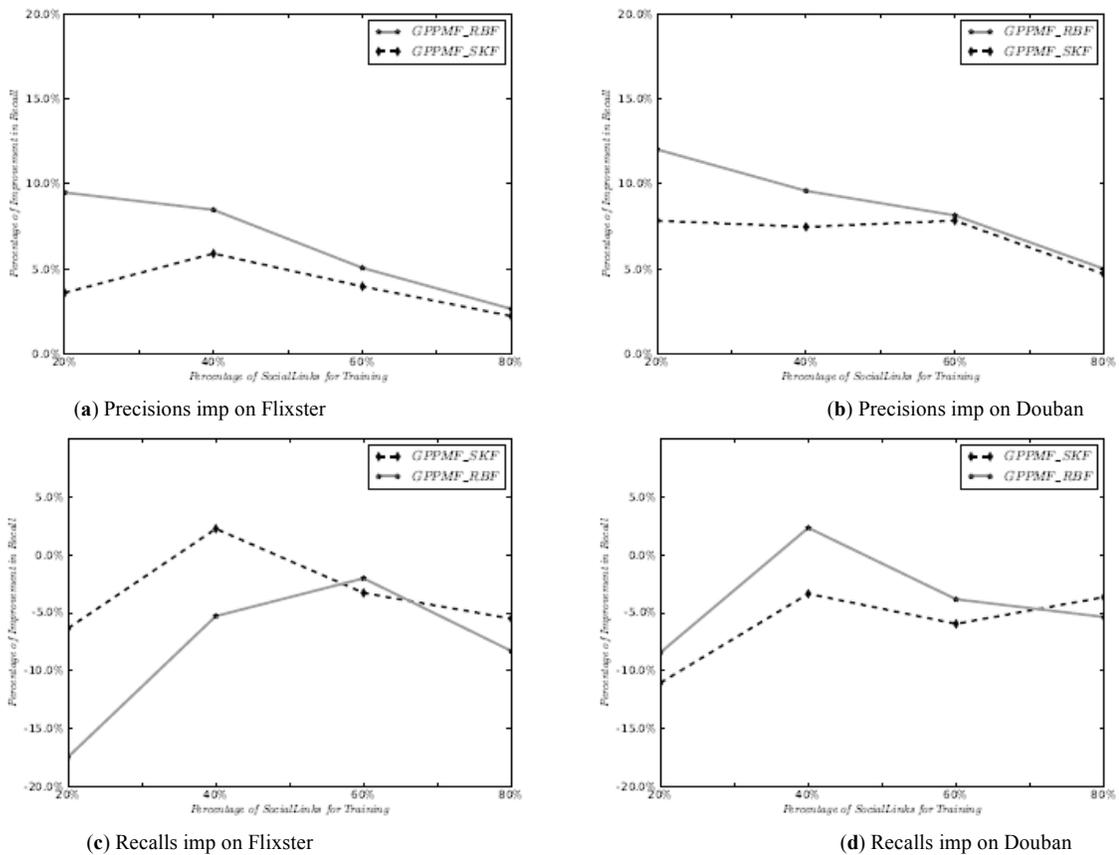


Fig. (5). Improvements on flixster and douban.

CONFLICT OF INTEREST

The authors confirm that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

The authors would like to thank the supports of Specialized Research Fund for the Doctoral Program of Higher Education (No. 20130005110011), the Fundamental Research Funds for the Central Universities (No. 2012RC0605), and Beijing Higher Education Young Elite Teacher Project(No.71A1411117, No. YETP1499).

REFERENCES

- [1] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a feather: Homophily in social networks," *Annual Review of Sociology*, vol. 27, pp. 415–444, 2001.
- [2] S.-H. Yang, B. Long, A. Smola, N. Sadagopan, Z. Zheng, and H. Zha, "Like like alike: joint friendship and interest propagation in social networks," in *Proceedings of the 20th international conference on World wide web*, ser. WWW '11, 2011, pp. 537–546.
- [3] C. Wang, V. Satuluri, and S. Parthasarathy, "Local probabilistic models for link prediction," in *ICDM, 2007*, pp. 322–331.
- [4] R. Lichtenwalter, J. T. Lussier, and N. V. Chawla, "New perspectives and methods in link prediction," in *KDD, 2010*, pp. 243–252.
- [5] R. Salakhutdinov and A. Mnih, "Probabilistic matrix factorization," in *Advances in Neural Information Processing Systems*, vol. 20, 2008.
- [6] R. Salakhutdinov and A. Mnih, "Bayesian probabilistic matrix factorization using Markov chain Monte Carlo," in *Proceedings of the International Conference on Machine Learning*, vol. 25, 2008.
- [7] T. Zhou, H. Shan, A. Banerjee, and G. Sapiro, "Kernelized probabilistic matrix factorization: Exploiting graphs and side information," in *SDM, 2012*, pp. 403–414. [Online]. Available: <http://dblp.uni-trier.de/db/conf/sdm/sdm2012.html#ZhouSBS12>
- [8] L. Lü and T. Zhou, "Link prediction in complex networks: A survey," *Physica A: Statistical Mechanics and its Applications*, vol. 390, no. 6, pp. 1150–1170, 2011.
- [9] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009. [Online]. Available: <http://www.computer.org/portal/web/csdl/doi/10.1109/MC.2009.263>
- [10] M. D. Ward, R. M. Siverson, and X. Cao, "Disputes, democracies, and dependencies: A reexamination of the kantian peace," *American Journal of Political Science*, vol. 51, no. 3, pp. 583–601, 2007.
- [11] X. Zhu and Z. Ghahramani, "Learning from labeled and unlabeled data with label propagation," *Technical Report CMU-CALD-02-107*, Carnegie Mellon University, Tech. Rep., 2002.
- [12] H. Kashima, T. Kato, Y. Yamanishi, M. Sugiyama, and K. Tsuda, "Link propagation: A fast semi-supervised learning algorithm for link prediction," in *SDM. SIAM, 2009*, pp. 1099–1110.
- [13] M. A. Hasan, V. Chaoji, S. Salem, and M. Zaki, "Link prediction using supervised learning," in *In Proc. of SDM 06 workshop on Link Analysis, Counterterrorism and Security, 2006*. [Online]. Available:<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.61.1225>
- [14] N. Beck, G. King, and L. Zeng, "Improving quantitative studies of international conflict: A conjecture," *American Political Science Review*, vol. 94, pp. 21–35, 2000.
- [15] A. Agovic, A. Banerjee, and S. Chatterjee, "Probabilistic matrix addition," in *ICML. Omnipress, 2011*, pp. 1025–1032.
- [16] L. Lov'asz, "Random Walks on Graphs: A Survey," *Bolyai Soc. Math. Stud.*, vol. 2, pp. 1–46, 1993.
- [17] S. Geisser, *Predictive inference*. CRC Press, 1993, vol. 55.
- [18] T. Zhou, L. Lü, and Y. Zhang, "Predicting missing links via local information," *The European Physical Journal B-Condensed Matter and Complex Systems*, vol. 71, no. 4, pp. 623–630, 2009.
- [19] L. Lü, C. Jin, and T. Zhou, "Similarity index based on local paths for link prediction of complex networks," *Physical Review E*, vol. 80, no. 4, p. 046122, 2009.

Received: September 16, 2014

Revised: December 23, 2014

Accepted: December 31, 2014

© Hu *et al.*; Licensee *Bentham Open*.

This is an open access article licensed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted, non-commercial use, distribution and reproduction in any medium, provided the work is properly cited.