

# TagRec: Leveraging Tagging Wisdom for Recommendation

Tom Chao Zhou, Hao Ma, Irwin King, Michael R. Lyu  
 Department of Computer Science and Engineering  
 The Chinese University of Hong Kong, Shatin, Hong Kong  
 Email: {czhou, hma, king, lyu}@cse.cuhk.edu.hk

**Abstract**—Due to the exponential growth of information on the Web, Recommender Systems have been developed to generate suggestions to help users overcome information overload and sift through huge amounts of information efficiently. Many existing approaches to recommender systems can neither handle very large datasets nor easily deal with users who have made very few ratings. Moreover, traditional recommender systems consider only the rating information, resulting in the loss of flexibility. Tagging has recently emerged as a popular way for users to annotate, organize and share resources on the Web. Several research tasks have shown that tags can represent users' judgments about Web contents quite accurately. In the light of the facts that both the rating activity and tagging activity can reflect users' opinions, this paper proposes a factor analysis approach called TagRec based on a unified probabilistic matrix factorization by utilizing both users' tagging information and rating information. The complexity analysis indicates that our approach can be applied to very large datasets. Furthermore, experimental results on MovieLens data set show that our method performs better than the state-of-the-art approaches.

## I. INTRODUCTION

Because of the exponential growth of information on the Web, users are in great need of effective recommendations in order to efficiently navigate through vast collections of items. Recommender Systems have been developed to suggest items that may interest users. Typically, recommender systems are based on Collaborative Filtering, which has been widely employed, such as in Amazon<sup>1</sup>, MovieLens<sup>2</sup> and etc. Recently, [1] has shown that collaborative filtering outperformed humans on the average through comprehensive experiments. Two trends have risen in recommendation algorithm: one is memory-based algorithms [2], [3], [4], and the other is model-based algorithms [5]. However, both types of algorithms suffer two weaknesses: (1) The recommendation performances deteriorate when the available ratings are very sparse. As claimed in [6], data sparsity is a common phenomenon in recommender systems, and the density of available ratings in commercial recommender systems is often less than 1%. (2) Almost all the traditional recommendation algorithms only employ the user-item rating matrix information, but ignore other user behaviors, leading to the loss of flexibility.

Social tagging systems have recently emerged as a popular way for users to annotate, organize and share resources on

the Web, such as del.icio.us<sup>3</sup>, Flickr<sup>4</sup> and MovieLens. As a type of social media sites [7], [8], [9], social tagging systems transform the Web into a participatory medium where users are actively creating, evaluating and distributing information. Previously, [10], [11], [12] have shown that tags can represent users' judgments about Web contents quite accurately, which are also good candidates to describe the resources.

In order to overcome the data sparsity problem and non-flexibility problem confronted by traditional recommendation algorithms mentioned above, this paper proposes a factor analysis approach by utilizing both users' rating information and tagging information based on probabilistic matrix factorization, and we refer to this method as TagRec. The experimental results on MovieLens 10M/100K data set<sup>5</sup> show that our method performs better than the state-of-the-art approaches; in the meanwhile, our complexity analysis also implies that our approach can be scaled to very large data sets.

The rest of the paper is organized as follows. In Section II, we introduce related work. Our TagRec framework is presented in Section III. Section IV shows the experimental results. Finally, we draw conclusions and discuss future work in Section V.

## II. RELATED WORK

In recommendation algorithms, the most studied memory-based approaches include user-based approaches [2], [13] and item-based approaches [6]. User-based methods look for some similar users who have similar rating styles with the active user and then employ the ratings from those similar users to predict the ratings for the active user. Item-based methods share similar idea with user-based methods except for finding similar items for each item. PCC [14] and VSS [2] are often applied in memory-based algorithms. The problems with memory-based recommendation algorithms is that the recommendation performances deteriorate when the rating data are very sparse. Model-based approaches include [15], [16]. Recently, the matrix factorization method which focuses on modeling the user-item rating matrix using low-rank approximations has been proposed for collaborative filtering [5], [17], [18]. However, almost all the approaches only engage the user-item rating matrix, and ignore other user behaviors. They also suffer the

<sup>1</sup><http://www.amazon.com>

<sup>2</sup><http://movielens.umn.edu>

<sup>3</sup><http://delicious.com>

<sup>4</sup><http://flickr.com>

<sup>5</sup><http://grouplens.org/node/73>

problem of data sparsity. Our method, on the other hand, shows three key differences: (1) Our method can deal with the missing value problem and performs better when the data are sparse. (2) The proposed framework incorporates tagging information with the user-item rating information, resulting in more flexibility, and includes the idea of multiple-source learning, such as in [19]. (3) Complexity analysis indicates that our method is scalable to very large data sets.

Recently, there are plenty of research efforts on social tagging systems. Several papers studied the utility value of tags, and found that tags were good at representing users' opinions about Web contents, including [10], [11], [12]. However, little is known about whether we can utilize tagging information to help improve recommendation quality. Our method differs from above work because we leverage tagging information to improve recommendation quality. The most recent work in [20] proposed an algorithm that predicts users' preferences for items based on their inferred preferences for tags. Our method differs from this work in two aspects: (1) Our method is interpreted using a probabilistic factor analysis model by utilizing rating information and tagging information together, thus can combine the best elements of both types of information. (2) Our model is a unified model instead of a two-step model in [20], thus is more systematic and general.

### III. TAGREC FRAMEWORK

#### A. Preliminaries

To facilitate our discussions, Table I defines basic terms and notations used throughout this paper.

#### B. User-Item Rating Matrix Factorization

As shown in Table I, we have  $m$  users and  $n$  items. The user-item rating matrix is denoted as  $R$ , and the element  $r_{ij}$  in  $R$  means the rating to item  $i_j$  given by user  $u_i$ , where values of  $r_{ij}$  are within the range  $[0, 1]$ . In recommender systems, ratings reflect users' judgments about the items, and most recommender systems use discrete rating values. Suppose the original rating values range from  $r_{min}$  to  $r_{max}$ , we use the function  $f(x) = (x - r_{min}) / (r_{max} - r_{min})$  as the mapping function to map the original rating values to values in the interval  $[0, 1]$ . As listed in Table I,  $U$  denotes the user latent feature matrix, and  $V$  denotes the item latent feature matrix, with column vectors  $U_i$  and  $V_j$  denoting the  $l$ -dimensional user-specific and item-specific latent feature vectors respectively. We define the conditional distributions over the observed ratings in Eq. (1):

$$p(R|U, V, \sigma_R^2) = \prod_{i=1}^m \prod_{j=1}^n [\mathcal{N}(r_{ij}|g(U_i^T V_j), \sigma_R^2)]^{I_{ij}^R}, \quad (1)$$

where  $I_{ij}^R$  is an indicator variable with the value of 1 if user  $u_i$  rated item  $i_j$ , and 0 otherwise. The meaning of  $U_i^T V_j$  is the rating user  $u_i$  gave to item  $i_j$  predicted by the model, and this is the typical matrix factorization approach.  $g(x) = 1 / (1 + e^{-x})$  is the logistic function to map the value of  $U_i^T V_j$

within the range of  $[0, 1]$ . Similar to [5], zero-mean spherical Gaussian priors are placed on the user and the item latent feature matrices, which are defined in Eq. (2):

$$\begin{aligned} p(U|\sigma_U^2) &= \prod_{i=1}^m \mathcal{N}(U_i|0, \sigma_U^2 \mathbf{I}), \\ p(V|\sigma_V^2) &= \prod_{j=1}^n \mathcal{N}(V_j|0, \sigma_V^2 \mathbf{I}). \end{aligned} \quad (2)$$

Through a Bayesian inference, the posterior distributions of  $U$  and  $V$  based only on the observed ratings are derived in Eq. (3):

$$\begin{aligned} &p(U, V|R, \sigma_V^2, \sigma_U^2, \sigma_R^2) \\ &\propto p(R|U, V, \sigma_R^2) p(U|\sigma_U^2) p(V|\sigma_V^2) \\ &= \prod_{i=1}^m \prod_{j=1}^n [\mathcal{N}(r_{ij}|g(U_i^T V_j), \sigma_R^2)]^{I_{ij}^R} \\ &\times \prod_{i=1}^m \mathcal{N}(U_i|0, \sigma_U^2 \mathbf{I}) \times \prod_{j=1}^n \mathcal{N}(V_j|0, \sigma_V^2 \mathbf{I}). \end{aligned} \quad (3)$$

#### C. User-Tag Tagging Matrix Factorization

As listed in Table I, we have  $m$  users and  $o$  tags. The user-tag tagging matrix is denoted as  $C$ , where the element  $c_{ik}$  in  $C$  represents the extent of user  $u_i$ 's preference for tag  $t_k$ . Users' tagging activities indicate users' preference for tags, so the meaning of  $c_{ik}$  can be interpreted as whether the user  $u_i$  has used the tag  $t_k$  (a binary representation), or how strong the user  $u_i$ 's preference is for the tag  $t_k$  (a real value representation). We represent  $c_{ik}$  in Eq. (4):

$$c_{ik} = g(f(u_i, t_k)), \quad (4)$$

where  $g(\cdot)$  is the logistic function, and  $f(u_i, t_k)$  represents the number of times user  $u_i$  uses tag  $t_k$ .

The idea of user-tag tagging matrix factorization is to derive two low-rank  $l$ -dimensional matrices  $U$  and  $T$ , representing the user latent feature matrix and the tag latent feature matrix respectively, based on the observed user-tag tagging matrix  $C$ . Denoting column vectors  $U_i$  and  $T_k$  as user-specific and tag-specific latent feature vectors respectively, we can define the conditional distributions over the observed user-tag tagging matrix in Eq. (5):

$$p(C|U, T, \sigma_C^2) = \prod_{i=1}^m \prod_{k=1}^o [\mathcal{N}(c_{ik}|g(U_i^T T_k), \sigma_C^2)]^{I_{ik}^C}, \quad (5)$$

where  $I_{ik}^C$  is an indicator variable with the value of 1 if user  $u_i$  has at least used tag  $t_k$  once, and 0 otherwise.

We also place the zero-mean spherical Gaussian priors, and through a Bayesian inference, we can derive the posterior

TABLE I  
BASIC NOTATIONS THROUGHOUT THIS PAPER

Notation	Description
$US = \{u_i\}_{i=1}^m$	$US$ is the set of users, $u_i$ is the $i$ -th user, $m$ is the total number of users
$IS = \{i_j\}_{j=1}^n$	$IS$ is the set of items, $i_j$ is the $j$ -th item, $n$ is the total number of items
$TS = \{t_k\}_{k=1}^o$	$TS$ is the set of tags, $t_k$ is the $k$ -th tag, $o$ is the total number of tags
$l \in \mathbb{R}$	$l$ is number of dimensions of latent feature space
$U \in \mathbb{R}^{l \times m}$	$U$ is the user latent feature matrix
$V \in \mathbb{R}^{l \times n}$	$V$ is the item latent feature matrix
$T \in \mathbb{R}^{l \times o}$	$T$ is the tag latent feature matrix
$R = \{r_{ij}\}, R \in \mathbb{R}^{m \times n}$	$R$ is the user-item rating matrix, $r_{ij}$ is rating that user $u_i$ gave to item $i_j$
$C = \{c_{ik}\}, C \in \mathbb{R}^{m \times o}$	$C$ is the user-tag tagging matrix, $c_{ik}$ is extent of user $u_i$ 's preference for tag $t_k$
$D = \{d_{jk}\}, D \in \mathbb{R}^{n \times o}$	$D$ is the item-tag tagging matrix, $d_{jk}$ is extent of how much tag $t_k$ can represent the concept of item $i_j$
$\mathcal{N}(x \mu, \sigma^2)$	Probability density function of the Gaussian distribution with mean $\mu$ and variance $\sigma^2$

distributions of  $U$  and  $T$  in Eq. (6):

$$\begin{aligned}
& p(U, T|C, \sigma_U^2, \sigma_T^2, \sigma_C^2) \\
& \propto p(C|U, T, \sigma_C^2) p(U|\sigma_U^2) p(T|\sigma_T^2) \\
& = \prod_{i=1}^m \prod_{k=1}^o [\mathcal{N}(c_{ik}|g(U_i^T T_k), \sigma_C^2)]^{I_{ik}^C} \\
& \times \prod_{i=1}^m \mathcal{N}(U_i|0, \sigma_U^2 \mathbf{I}) \times \prod_{k=1}^o \mathcal{N}(T_k|0, \sigma_T^2 \mathbf{I}). \quad (6)
\end{aligned}$$

#### D. Item-Tag Tagging Matrix Factorization

As denoted in Table I, we have  $n$  items and  $o$  tags. The item-tag tagging matrix is denoted as  $D$ , and the element  $d_{jk}$  in  $D$  shows the extent of how much tag  $t_k$  can represent the concept of item  $i_j$ . Users annotate items with tags to express their judgments about items and distinguish one item from another. The meaning of  $d_{jk}$  can be interpreted as whether item  $i_j$  has been annotated with the tag  $t_k$  (a binary representation), or how strong tag  $t_k$ 's representing ability is for item  $i_j$  (a real value representation). We represent  $d_{jk}$  in Eq. (7):

$$d_{jk} = g(h(i_j, t_k)), \quad (7)$$

where  $g(\cdot)$  is the logistic function, and  $h(i_j, t_k)$  is the number of times item  $i_j$  is annotated with tag  $t_k$ .

The idea of item-tag tagging matrix is to derive two low-rank  $l$ -dimensional matrices  $V$  and  $T$ , representing the item latent feature matrix and the tag latent feature matrix respectively, based on the observed item-tag tagging matrix  $D$ . Denoting column vectors  $V_j$  and  $T_k$  as item-specific and tag-specific latent feature vectors respectively, we can define the conditional distributions over the observed item-tag tagging matrix in Eq. (8):

$$p(D|V, T, \sigma_D^2) = \prod_{j=1}^n \prod_{k=1}^o [\mathcal{N}(d_{jk}|g(V_j^T T_k), \sigma_D^2)]^{I_{jk}^D}, \quad (8)$$

where  $I_{jk}^D$  is an indicator variable with the value of 1 if item  $i_j$  is annotated with tag  $t_k$ , and 0 otherwise.

Through a Bayesian inference, we can derive the posterior

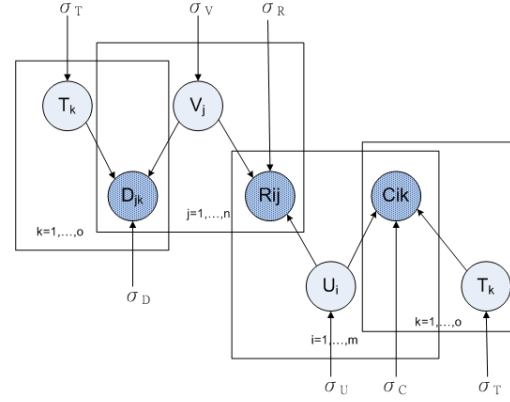


Fig. 1. Graphical Model for TagRec

distributions of  $V$  and  $T$  in Eq. (9):

$$\begin{aligned}
& p(V, T|D, \sigma_D^2, \sigma_T^2, \sigma_V^2) \\
& \propto p(D|V, T, \sigma_D^2) p(V|\sigma_V^2) p(T|\sigma_T^2) \\
& = \prod_{j=1}^n \prod_{k=1}^o [\mathcal{N}(d_{jk}|g(V_j^T T_k), \sigma_D^2)]^{I_{jk}^D} \\
& \times \prod_{j=1}^n \mathcal{N}(V_j|0, \sigma_V^2 \mathbf{I}) \times \prod_{k=1}^o \mathcal{N}(T_k|0, \sigma_T^2 \mathbf{I}). \quad (9)
\end{aligned}$$

#### E. A Unified Matrix Factorization for TagRec

As discussed in Section I, since both users' rating information and users' tagging information can reflect users' judgments about Web contents, we propose a factor analysis approach by utilizing both users' rating information and tagging information based on a unified probabilistic matrix factorization. Specifically, on the one hand, we connect users' rating information with users' tagging information through the shared user latent feature space, and on the other hand, we connect items' received rating information with items' received tagging information through the shared item latent feature space. The shared tag latent feature space is used to represent user-tag tagging information and item-tag tagging information. The graphical model describing the TagRec framework is represented in Fig. 1.

According to the graphical model described in Fig. 1, we derive the log function of the posterior distributions of TagRec in Eq. (10):

$$\begin{aligned}
& \ln p(U, V, T | R, C, D, \sigma_R^2, \sigma_V^2, \sigma_U^2, \sigma_T^2, \sigma_D^2, \sigma_R^2, \sigma_C^2) = \\
& -\frac{1}{2\sigma_R^2} \sum_{i=1}^m \sum_{j=1}^n I_{ij}^R (r_{ij} - g(U_i^T V_j))^2 \\
& -\frac{1}{2\sigma_C^2} \sum_{i=1}^m \sum_{k=1}^o I_{ik}^C (c_{ik} - g(U_i^T T_k))^2 \\
& -\frac{1}{2\sigma_D^2} \sum_{j=1}^n \sum_{k=1}^o I_{jk}^D (d_{jk} - g(V_j^T T_k))^2 \\
& -\frac{1}{2\sigma_U^2} \sum_{i=1}^m U_i^T U_i - \frac{1}{2\sigma_V^2} \sum_{j=1}^n V_j^T V_j - \frac{1}{2\sigma_T^2} \sum_{k=1}^o T_k^T T_k \\
& -\sum_{i=1}^m \sum_{j=1}^n I_{ij}^R \ln \sigma_R - \sum_{i=1}^m \sum_{k=1}^o I_{ik}^C \ln \sigma_C - \sum_{j=1}^n \sum_{k=1}^o I_{jk}^D \ln \sigma_D \\
& -l \sum_{i=1}^m \ln \sigma_U - l \sum_{j=1}^n \ln \sigma_V - l \sum_{k=1}^o \ln \sigma_T + \mathcal{C}, \quad (10)
\end{aligned}$$

where  $\mathcal{C}$  is a constant independent of the parameters. We can see the Eq. (10) is an unconstrained optimization problem, and maximizing the log-posterior distributions with fixed hyperparameters is equivalent to minimizing the sum-of-squared-errors objective function with quadratic regularized terms in Eq. (11):

$$\begin{aligned}
& E(U, V, T, R, C, D) \\
& = \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^n I_{ij}^R (r_{ij} - g(U_i^T V_j))^2 \\
& + \frac{\theta_C}{2} \sum_{i=1}^m \sum_{k=1}^o I_{ik}^C (c_{ik} - g(U_i^T T_k))^2 \\
& + \frac{\theta_D}{2} \sum_{j=1}^n \sum_{k=1}^o I_{jk}^D (d_{jk} - g(V_j^T T_k))^2 \\
& + \frac{\theta_U}{2} \sum_{i=1}^m U_i^T U_i + \frac{\theta_V}{2} \sum_{j=1}^n V_j^T V_j + \frac{\theta_T}{2} \sum_{k=1}^o T_k^T T_k, \quad (11)
\end{aligned}$$

where  $\theta_C = \sigma_R^2/\sigma_C^2$ ,  $\theta_D = \sigma_R^2/\sigma_D^2$ ,  $\theta_U = \sigma_R^2/\sigma_U^2$ ,  $\theta_V = \sigma_R^2/\sigma_V^2$ , and  $\theta_T = \sigma_R^2/\sigma_T^2$ . The local minimum can be found by performing the gradient descent on  $U_i$ ,  $V_j$  and  $T_k$ , and the derived gradient descent equations are described in Eq. (12), Eq. (13) and Eq. (14) respectively:

$$\begin{aligned}
\frac{\partial E}{\partial U_i} & = \sum_{j=1}^n I_{ij}^R (g(U_i^T V_j) - r_{ij}) g'(U_i^T V_j) V_j + \theta_U U_i \\
& + \theta_C \sum_{k=1}^o I_{ik}^C (g(U_i^T T_k) - c_{ik}) g'(U_i^T T_k) T_k, \quad (12)
\end{aligned}$$

$$\begin{aligned}
\frac{\partial E}{\partial V_j} & = \sum_{i=1}^m I_{ij}^R (g(U_i^T V_j) - r_{ij}) g'(U_i^T V_j) U_i + \theta_V V_j \\
& + \theta_D \sum_{k=1}^o I_{jk}^D (g(V_j^T T_k) - d_{jk}) g'(V_j^T T_k) T_k, \quad (13)
\end{aligned}$$

$$\begin{aligned}
\frac{\partial E}{\partial T_k} & = \theta_C \sum_{i=1}^m I_{ik}^C (g(U_i^T T_k) - c_{ik}) g'(U_i^T T_k) U_i + \theta_T T_k \\
& + \theta_D \sum_{j=1}^n I_{jk}^D (g(V_j^T T_k) - d_{jk}) g'(V_j^T T_k) V_j, \quad (14)
\end{aligned}$$

where  $g'(\cdot)$  is the first-order derivative of the logistic function. We set  $\theta_U = \theta_V = \theta_T$  in our experiments in order to reduce the model complexity.

#### F. Complexity Analysis

The major computation cost of the gradient descent methods are evaluating objective function  $E$  and corresponding gradients on variables. Due to the sparsity of matrices  $R$ ,  $C$ , and  $D$ , the complexity of evaluating the objective function in Eq. (11) is  $\mathcal{O}(n_R l + n_C l + n_D l)$ , where  $n_R$ ,  $n_C$  and  $n_D$  are the number of non-zero entries in matrices  $R$ ,  $C$  and  $D$  respectively, and  $l$  is the number of dimensions of latent feature space as shown in Table I. Similarly we can derive the complexities of Eq. (12), Eq. (13) and Eq. (14). Hence, the total complexity for one iteration is  $\mathcal{O}(n_R l + n_C l + n_D l)$ , which means it is linear with respect to the number of observations in the three sparse matrices. As claimed in [6] the density of available ratings in commercial recommender systems is often less than 1%; therefore, TagRec is efficient and is scalable to large data sets.

## IV. EXPERIMENTAL ANALYSIS

We first ask several research questions intended to give an idea of the highlights of our experimental analysis.

- RQ1 How is our approach compared with the baseline methods and the existing state-of-the-art approaches?
- RQ2 How do the model parameters  $\theta_C$  and  $\theta_D$  affect the prediction accuracies of our approach?

#### A. Description of MovieLens Data Set and Metrics

We use MovieLens 10M/100K data set in our experiments. This data set contains 10000054 ratings and 95580 tags added to 10681 movies by 71567 users of the online movie recommender service MovieLens. In order to compare the prediction quality of our method with other methods, we use the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE) as the comparison metrics. MAE is defined in Eq. (15), and RMSE is defined in Eq. (16):

$$MAE = \frac{\sum_{i,j} |r_{i,j} - \hat{r}_{i,j}|}{N}, \quad (15)$$

$$RMSE = \sqrt{\frac{\sum_{i,j} (r_{i,j} - \hat{r}_{i,j})^2}{N}}. \quad (16)$$

where  $r_{i,j}$  denotes the rating user  $i$  gave to item  $j$ ,  $\hat{r}_{i,j}$  denotes the predicted rating, and  $N$  is the total number of tested ratings.

## B. Comparison

In order to show the prediction performance improvements of TagRec, we compare TagRec with two baseline methods user mean (UMEAN) and item mean (IMEAN). UMEAN is defined in Eq. (17) and IMEAN is defined in Eq. (18):

$$\hat{r}_{i,j} = \frac{\sum_n r_{i,n}}{N}, \quad (17)$$

where  $r_{i,n}$  is the observed ratings of user  $i$  in the training data and  $N$  is the number of observed ratings of user  $i$ .

$$\hat{r}_{i,j} = \frac{\sum_m r_{m,j}}{M}, \quad (18)$$

where  $r_{m,j}$  is the observed ratings of item  $j$  in the training data and  $M$  is the number of observed ratings of item  $j$ . In addition, we ultimately compare our TagRec approach with top-performing recommendation algorithms, including Probabilistic Matrix Factorization (PMF) [5] and Singular Value Decomposition (SVD) [21].

In the comparison, we employ different amount of training data, including 80%, 50%, 30%, 20% and 10%. 80% training data means we randomly select 80% of ratings from the MovieLens 10M/100K data set as training data, and leave the remaining 20% as prediction performance testing. The procedure is carried out 5 times independently, and we report the average values in this paper. In the comparison, we set  $\theta_U = \theta_V = \theta_T = 0.004$ , set  $\theta_C = 0.4$  and set  $\theta_D = 10$ . The MAE results and RMSE results are reported in Table II and Table III respectively. From the results, we can see that our TagRec approach consistently outperforms existing algorithms, especially when there is a small amount of training data, which indicates our method performs better under sparse data settings. In addition, it is necessary to notice that in MovieLens 10M/100K data set, all the selected users have rated at least 20 movies, but in reality, according to the famous power law distribution phenomenon, in almost all kinds of Web activities most users have rated very few items. Thus, we can see the improvement of TagRec is significant, and this shows the promising future of our TagRec approach.

## C. Impact of Parameters $\theta_C$ and $\theta_D$

TagRec utilizes both users' rating information and tagging information at the same time to perform the prediction. Specifically, we incorporate user-item rating matrix, user-tag tagging matrix, and item-tag tagging matrix together based on a unified probabilistic matrix factorization. The parameter  $\theta_C$  controls the impact of the user-tag tagging matrix, and the parameter  $\theta_D$  controls the impact of the item-tag tagging matrix. If we set both  $\theta_C$  and  $\theta_D$  as 0, it means we only consider users' rating information; if we set both  $\theta_C$  and  $\theta_D$  to  $+\infty$ , it means we only utilize users' tagging information.

We test the impact of these two parameters independently. When we test the impact of parameter  $\theta_C$ , we set  $\theta_U = \theta_V = \theta_T = 0.004$ ,  $\theta_D = 10$ , and Fig. 2(a) and Fig. 2(b) show the results. When we test the impact of parameter  $\theta_D$ , we set  $\theta_U = \theta_V = \theta_T = 0.004$ ,  $\theta_C = 0.4$ , and Fig. 2(c)

and Fig. 2(d) present the results. We report results when *dimensionality* = 20 in Fig. 2, and the results are similar when *dimensionality* = 10. From the results presented in Fig. 2, we can see that both the values of  $\theta_C$  and  $\theta_D$  impact the prediction accuracies significantly, and this indicates that utilizing both users' rating information and users' tagging information simultaneously can improve the prediction quality. We further observe that as the value of  $\theta_C$  or  $\theta_D$  increases, both the MAE and RMSE first decrease (performances increase); but after  $\theta_C$  or  $\theta_D$  is greater than some threshold value, both MAE and RMSE start to increase again (performances decrease). This observation meets our expectation, because only utilizing users' rating information or only utilizing users' tagging information cannot perform better than utilizing rating information and tagging information together. Our approach performs best when  $\theta_C \in [0.1, 1]$  and  $\theta_D \in [5, 10]$ , and the relatively wide range of choosing optimal parameter indicates that the model is easy to train.

## V. CONCLUSIONS AND FUTURE WORK

Based on the intuition that both users' rating information and users' tagging information can reflect users' judgments about Web contents, and that tags added to items can represent concepts of items, we propose the TagRec framework, which employs users' rating information and tagging with a unified probabilistic matrix factorization. The experimental results show that the innovative TagRec approach outperforms existing approaches. The proposed approach uses the explicit relations directly, such as users' rating information and tagging information; the approach also considers each user and each item equally, ignoring the fact that there may be some hidden structures among all the users and all the items. In the future, we will investigate whether it is possible to first mine these explicit relations to infer some implicit relations, and then use the inferred implicit relations and the original explicit relations together to improve the recommendation quality.

## ACKNOWLEDGMENT

We thank GroupLens Research<sup>6</sup> for providing the MovieLens data sets. The work described in this paper is supported by grants from the Research Grant Council of the Hong Kong Special Administrative Region, China (Project No.: CUHK 4128/08E and CUHK 4158/08E) and Microsoft (Project No.: 6902498). This work is also affiliated with the Microsoft-CUHK Joint Laboratory for Human-centric Computing and Interface Technologies.

## REFERENCES

- [1] V. Krishnan, P. K. Narayanashetty, M. Nathan, R. T. Davies, and J. A. Konstan, "Who predicts better?: Results from an online study comparing humans and an online recommender system," in *Proceedings of the 2008 ACM Conference on Recommender Systems*, 2008.
- [2] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *UAI'98: Proceedings of Uncertainty in Artificial Intelligence*, 1998.

<sup>6</sup><http://www.grouplens.org/>

TABLE II  
MAE COMPARISON WITH OTHER APPROACHES (A SMALLER MAE VALUE MEANS A BETTER PERFORMANCE)

Training Data	Baseline Methods		Dimensionality = 10			Dimensionality = 20		
	UMEAN	IMEAN	SVD	PMF	TagRec	SVD	PMF	TagRec
80%	0.7686	0.7379	0.6169	0.6162	<b>0.6159</b>	0.6167	0.6156	<b>0.6145</b>
50%	0.7710	0.7389	0.6376	0.6354	<b>0.6352</b>	0.6349	0.6337	<b>0.6307</b>
30%	0.7742	0.7399	0.6617	0.6599	<b>0.6528</b>	0.6570	0.6569	<b>0.6494</b>
20%	0.7803	0.7416	0.6813	0.6811	<b>0.6664</b>	0.6776	0.6766	<b>0.6650</b>
10%	0.8234	0.7484	0.7315	0.7127	<b>0.6964</b>	0.7264	0.7089	<b>0.6962</b>

TABLE III  
RMSE COMPARISON WITH OTHER APPROACHES (A SMALLER RMSE VALUE MEANS A BETTER PERFORMANCE)

Training Data	Baseline Methods		Dimensionality = 10			Dimensionality = 20		
	UMEAN	IMEAN	SVD	PMF	TagRec	SVD	PMF	TagRec
80%	0.9779	0.9440	0.8087	0.8078	<b>0.8077</b>	0.8054	0.8025	<b>0.8022</b>
50%	0.9816	0.9463	0.8330	0.8326	<b>0.8321</b>	0.8289	0.8252	<b>0.8217</b>
30%	0.9869	0.9505	0.8636	0.8587	<b>0.8492</b>	0.8575	0.8553	<b>0.8450</b>
20%	1.0008	0.9569	0.8900	0.8824	<b>0.8659</b>	0.8857	0.8791	<b>0.8639</b>
10%	1.1587	0.9851	0.9703	0.9236	<b>0.9038</b>	0.9638	0.9183	<b>0.9031</b>

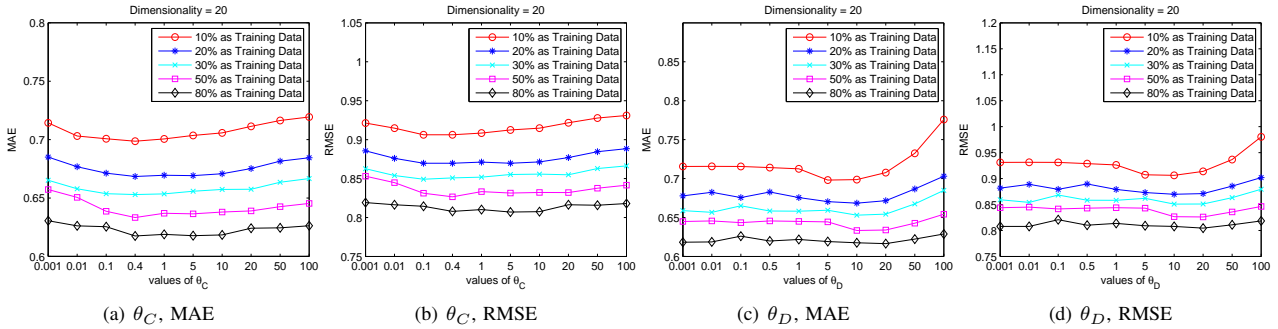


Fig. 2. Dimensionality=20, Impact of Parameters  $\theta_C$  and  $\theta_D$ .

- [3] H. Ma, I. King, and M. R. Lyu, "Effective missing data prediction for collaborative filtering," in *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2007, pp. 39–46.
- [4] Z. Zheng, H. Ma, M. R. Lyu, and I. King, "Wsrec: A collaborative filtering based web service recommender system," in *Proceedings of the 7th IEEE International Conference on Web Services*, 2009.
- [5] R. Salakhutdinov and A. Mnih, "Probabilistic matrix factorization," *Advances in Neural Information Processing Systems*, vol. 20, 2008.
- [6] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proceedings of the 10th International Conference on World Wide Web*, 2001, pp. 285–295.
- [7] Y. I. Song, C. Y. Lin, Y. Cao, and H. C. Rim, "Question utility: A novel static ranking of question search," in *Proceedings of the 23rd AAAI Conference on Artificial Intelligence*, 2008, pp. 1231–1236.
- [8] Y. Cao, H. Duan, C. Y. Lin, Y. Yu, and H. W. Hon, "Recommending questions using the mdl-based tree cut model," in *Proceedings of the 17th International Conference on World Wide Web*, 2008, pp. 81–90.
- [9] G. Cong, L. Wang, C. Y. Lin, Y. I. Song, and Y. Sun, "Finding question-answer pairs from online forums," in *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2008, pp. 467–474.
- [10] P. Heymann, G. Koutrika, and H. Garcia-Molina, "Can social bookmarking improve web search?" in *Proceedings of the International Conference on Web Search and Data Mining*, 2008, pp. 195–206.
- [11] X. Li, L. Guo, and Y. E. Zhao, "Tag-based social interest discovery," in *Proceedings of the 17th International Conference on World Wide Web*, 2008, pp. 675–684.
- [12] S. Sen, S. K. Lam, A. M. Rashid, D. Cosley, D. Frankowski, J. Osterhouse, F. M. Harper, and J. Riedl, "Tagging, communities, vocabulary, evolution," in *Proceedings of the 20th Anniversary Conference on Computer Supported Cooperative Work*, 2006, pp. 181–190.
- [13] R. Jin, J. Chai, and L. Si, "An automatic weighting scheme for collaborative filtering," in *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2004, pp. 337–344.
- [14] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: An open architecture for collaborative filtering of netnews," in *Proceedings of ACM Conference on Computer Supported Cooperative Work*, 1994.
- [15] T. Hofmann, "Latent semantic models for collaborative filtering," *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 89–115, 2004.
- [16] G. Xue, C. Lin, Q. Yang, W. Xi, H. Zeng, Y. Yu, and Z. Chen, "Scalable collaborative filtering using cluster-based smoothing," in *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2005, pp. 114–121.
- [17] H. Ma, H. Yang, M. R. Lyu, and I. King, "Sorec: Social recommendation using probabilistic matrix factorization," in *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, 2008, pp. 931–940.
- [18] H. Ma, I. King, and M. R. Lyu, "Learning to recommend with social trust ensemble," in *Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2009.
- [19] H. Deng, I. King, and M. R. Lyu, "Formal models for expert finding on dblp bibliography data," in *Proceedings of the 8th IEEE International Conference on Data Mining*, 2008, pp. 163–172.
- [20] S. Sen, J. Vig, and J. Riedl, "Tagommenders: Connecting users to items through tags," in *Proceedings of the 18th International Conference on World Wide Web*, 2009.
- [21] S. Funk, "Netflix update: Try this at home," [sifter.org/simon/journal/20061211.html](http://sifter.org/simon/journal/20061211.html), Tech. Rep., 2006.