# Cross-domain Recommendation with Probabilistic Knowledge Transfer

Qian Zhang, Dianshuang Wu, Jie Lu, and Guangquan Zhang

Decision Systems and e-Service Intelligence Laboratory, Center for Artificial Intelligence, Faculty of Engineering and Information Technology, University of Technology Sydney, Australia
{qian.zhang-1, dianshuang.wu, jie.lu, guangquan.zhang}@uts.edu.au

**Abstract.** Recommender systems have drawn great attention from both academic and practical area. One challenging and common problem in many recommendation methods is data sparsity, due to the limited number of observed user interaction with the products/services. To alleviate the data sparsity problem, cross-domain recommendation methods are developed to share group-level knowledge in several domains so that recommendation in the domain with scarce data can benefit from domains with relatively abundant data. However, divergence exists in the data of similar domains so that the extracted group-level knowledge is not always suitable to be applied in the target domain, thus recommendation accuracy in the target domain is impaired. In this paper, we propose a cross-domain recommendation method with probabilistic knowledge transfer. The proposed method maintain two sets of group-level knowledge, profiling both domain-shared and domain-specific characteristics of the data. In this way users' mixed preferences can be profiled comprehensively thus improves the performance of the cross-domain recommender systems. Experiments are conducted on five real-world datasets in three categories: movies, books and music. The results for nine cross-domain recommendation tasks show that our proposed method has improved the accuracy compared with five benchmarks.

**Keywords:** Recommender systems · Cross-domain recommender systems · Knowledge transfer · Probabilistic model

## 1 Introduction

Recommender systems are rapidly developed and widely used in e-commerce and online shopping website [7]. These systems aim to provide recommendations to users to help them choose products or services they need in the era of information explosion. One basic and challenging issue is the data sparsity problem, which greatly impairs the performance of recommender systems, leading to poor user experience and their unsatisfactory [11]. Cross-domain recommender systems are developed to deal with this problem in the fierce market competition [1]. One user may not have enough data in one domain, but have more data in another domain. The abundance of data in another domain can assist the recommendation in a

specific target domain. By taking the advantages of data in multiple domains, cross-domain recommender systems can exploit the relatively dense data in the source domain to assist recommendation with scarce data in the target domain.

Cross-domain recommender systems can be clustered into three groups: cross-domain recommender systems with side information, with partially overlapping entities and with non-overlapping entity. Due to the privacy issue, users are always de-identified and the correspondence between users are not available [13]. In this paper, we focus on the most commonly happened scenario: cross-domain with non-overlapping entities. Some methods are developed to handle this problem by transferring knowledge from group-level. Users and items are clustered into groups and knowledge is shared through group-level rating patterns. For example, codebook transfer (CBT) clusters users and items into groups and extracts group-level knowledge as a "codebook" [4]. Later, a probabilistic model named rating matrix generative model (RMGM) is extended from CBT, relaxing the hard group membership to soft membership [5]. These two methods cannot ensure that the information on the two groups from two different domains is consistent, and the effectiveness of knowledge transfer is not guaranteed.

Although cross-domain recommender systems have gained lots of attention and efforts from academia, they still suffer the "negative transfer" problem [6, 8]. The main reason is that data collected from two correlated domains are probably from two related but different distributions. Most existing methods on cross-domain recommendation ignore the domain shifts and extract group-level knowledge directly without considering domain-specific characteristics. The group-level knowledge is not suited to the target domain, thus degrades the performance of the cross-domain recommender system.

In this paper, we investigate how to improve the performance of cross-domain recommender systems by exploring domain-specific characteristics in each domain, which is crucial when divergence exists in the data. The proposed method assumes that ratings are generated from two sets of group-level knowledge. One is shared across multiple domains, i.e. the domain-shared knowledge, while the other is domain-specific knowledge which is different for each domain. Users' mixed preferences in a target domain can be profiled with the help of common features extracted from data in other domains and reserve some unique features of the data from the target domain. In this way, group-level knowledge is shared while domain-specific knowledge remained. Probabilistic model is a suitable and powerful way to generate two sets of rating patterns from two different priors, and then generate the ratings. Thus, we propose a probabilistic method for knowledge transfer in cross-domain recommendation (ProbKT). The main contributions of this paper are as follows:

(1) A cross-domain recommendation method ProbKT is developed to enable transferring domain-share knowledge while remaining domain-specific characteristics at the same time.
(2) ProbKT has advantages in effectively transferring knowledge in multiple domains with similar data where divergence may exist.

(3) The proposed method ProbKT is evaluated on five real-world datasets with nine cross-domain recommendation tasks comparing with five other non-transfer or cross-domain recommendation methods. The results show that our proposed method outperforms other recommendation methods in sparse data.

The rest of the paper is organized as follows. Section 2 gives some preliminary and a formal description of the problem. Section 3 describes our method using probabilistic model to enable cross-domain recommendation in multiple domains. In Section 4, we present our experiments on five real-world datasets containing three data categories. Finally, in Section 5, conclusion is provided with some future directions of this research.

## 2 Preliminary and Problem Formation

In this section, cross-domain recommendation by tri-factorization is briefly introduced. The problem targeted in this paper is also formally formulated.

### 2.1 Cross-domain recommendation by tri-factorization

Matrix factorization projects both users and items onto the same latent space so that they are comparable, and through their inner products reconstructs the rating matrix [3]. Similarly, the rating matrix $\boldsymbol{R} \in \mathbb{R}^{M \times N}$ (bold letters represent matrixes) can be factorized into three matrixes (suppose there are $M$ users and $N$ items). Users and items are clustered into several latent groups and in the middle is the group-level rating pattern: $\boldsymbol{R} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T$, where $\boldsymbol{U} \in \mathbb{R}^{M \times K}$ is user group membership matrix, representing users clustered into $K$ groups, $\boldsymbol{V} \in \mathbb{R}^{N \times L}$ is item group membership matrix, representing items clustered into $L$ groups and $\boldsymbol{S} \in \mathbb{R}^{K \times L}$ is the group rating pattern matrix, i.e. the group-level knowledge.

Say rating matrixes in $D$ domains are available, denoted as $\boldsymbol{R}^d$. The assumption of the cross-domain recommender systems is that the group-level knowledge can be shared if these domains are similar. Thus for the $D$ rating matrixes are reconstructed as:

$$\hat{\boldsymbol{R}}^d = \boldsymbol{U}^d \boldsymbol{S} (\boldsymbol{V}^d)^T \tag{1}$$

### 2.2 Problem formulation

In our problem setting, there is no correspondence on the users/items across the domains and users/items are treated as completely different. We assume that on both the source and target domains the data are explicit ratings. The problem is formally defined as:

Given $D$ rating matrixes $\boldsymbol{R}^D = \{\boldsymbol{R}^1, ..., \boldsymbol{R}^d, ..., \boldsymbol{R}^D\}$, $\boldsymbol{R}^d \in \mathbb{R}^{M^d \times N^d}$, our goal is to develop a cross-domain recommendation method to assist the recommendation task of predicting the rating using knowledge in one target domain $\boldsymbol{R}^t$ from all the other rating matrixes in $\boldsymbol{R}^D$, where for each source domain data $\mathcal{U}^d \cap \mathcal{U}^t = \emptyset$ and $\mathcal{I}^d \cap \mathcal{I}^t = \emptyset$.
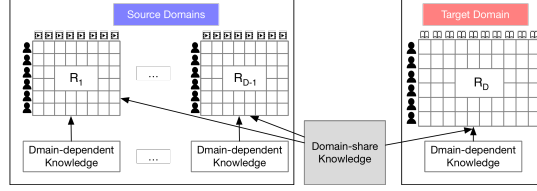
**Fig. 1.** An example to explain how knowledge is shared in the ProbKT method.

## 3   Cross-domain Recommendation with Probabilistic Knowledge Transfer

In this section, our proposed ProbKT method is to learn a joint probabilistic model using data in multiple domains.

### 3.1   The Method Description

As we have seen in the tri-factorization model, the group-level knowledge can be shared cross domains. Since users and items have no intersections, the group-level knowledge may not be totally the same, especially when divergence exists in data between two domains. If we force the group-level knowledge to be the same in two domains, it is likely that ratings predicted are not accurate in the target domain since the extracted knowledge from the source domain is not effectively adapted. We assume that only partial group-level knowledge can be shared between two domains. For a matrix factorization model, it is not easy to solve the optimization problem with constraints on part of the group-level matrix. As shown in Fig 1, the domain-shared knowledge contributes to data in each domain. Except that, each domain has its own domain-dependent knowledge that contribute to its corresponding domain. In this example shown in Fig 1, knowledge is extracted from several datasets in movie domain and applied to the targeted book domain. The cross-domain recommender systems is to recommend books to users assisted by data in movie domain. Here, a probabilistic model fits our assumption and is suitable to solve the knowledge transfer issue for this cross-domain recommendation scenario.

Given user-item rating matrixes in $D$ domains $\boldsymbol{R} = \{\boldsymbol{R}^1, ..., \boldsymbol{R}^d, ..., \boldsymbol{R}^D\}$, where $\boldsymbol{R}^d \in \mathbb{R}^{M^d \times N^d}$ in the domain $d$, $M^d$ is the number of users and $N^d$ is the number of items. In each domain, users and items are from $K$ user groups and $L$ item groups. In these groups, $K^s$ user groups are shared groups between these domains while $K^e$ user groups are specific groups for each domain ($K^s + K^e = K$). Similar notations $L^s$ and $L^e$ go for item groups. One rating $R_{ij}^d$ represents the rating provided by user $i$ on item $j$. For this rating, it is associated with two latent variables: $Z_{ij}^u$ and $Z_{ij}^v$, which represent the user-group and item-group of this rating. There are two group-level rating patterns: $\boldsymbol{\phi}^s$ representing domain-shared group-level knowledge and $\boldsymbol{\phi}^e$ representing domain-specific group-level knowledge.
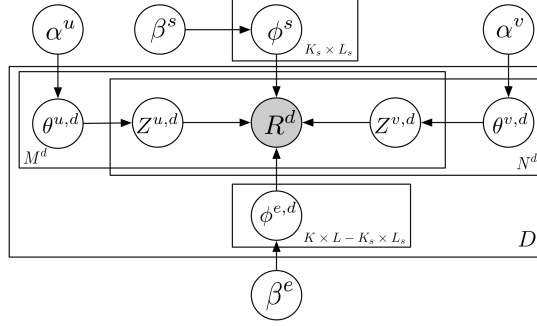
**Fig. 2.** Graphical model representation of the ProbKT method.

The graphical model representation is in Fig 2. The generative process is as follows:

(1) For domain-shared user-item joint groups $(k^s, l^s)$
   Generate $\phi^s_{k^s, l^s} \sim Dir(\beta^s)$
(2) For each domain $d$
   (a) For domain-specific user-item joint groups $(k^e, l^e)$
      Generate $\phi^e_{k^e, l^e} \sim Dir(\beta^e)$
   (b) For each user $i = 1, ..., M^d$
      Generate $\theta^{u,d}_i \sim Dir(\alpha^u)$
   (c) For each item $j = 1, ..., N^d$
      Generate $\theta^{v,d}_j \sim Dir(\alpha^v)$
   (d) For the rating $R^d_{ij}$
      (i) Generate a user group $Z^u_{ij} \sim Multi(\theta^{u,d}_i)$
      (ii) Generate an item group $Z^v_{ij} \sim Multi(\theta^{v,d}_j)$
      (iii) For user groups $Z^u_{ij} = 1, ..., K^s$ and item groups $Z^v_{ij} = 1, ..., L^s$
         Generate a rating $R^d_{ij} \sim Multi(\phi^s_{Z^u_{ij}, Z^v_{ij}})$;
         For user groups $Z^u_{ij} = 1, ..., K^s$ and item groups $Z^v_{ij} = L^s + 1, ..., L$,
         user groups $Z^u_{ij} = K^s + 1, ..., K$ and item groups $Z^v_{ij} = 1, ..., L$
         Generate a rating $R^d_{ij} \sim Multi(\phi^{e,d}_{Z^u_{ij}, Z^v_{ij}})$

Here, $\alpha^u$, $\alpha^v$, $\beta^e$ and $\beta^s$ are hyper-parameters of the Dirichlet priors. Through this model, the domain-shared knowledge $\phi^s$ and domain-independent knowledge $\phi^e$ compose the group-level knowledge $\phi$ in the target domain. The group-level rating matrix, together with the user group membership matrix and item group membership matrix are calculated by: $S_{kl} = \sum_{r=1}^{R_0} r\phi_{klr}$, $U^t_{ik} = \theta^u_{ik}$, and $V^t_{jl} = \theta^v_{jl}$. In our experiment settings, ratings are all from $\{1, 2, 3, 4, 5\}$, thus $R_0 = 5$.

### 3.2   The Method Learning

For our probabilistic method, we have latent variable set $\Omega = \{\theta^{u, 1:\sum_{d=1}^D M^d},$
$\theta^{v, 1:\sum_{d=1}^D N^d}, Z^{u, 1:\sum_{d=1}^D M^d N^d}, Z^{v, 1:\sum_{d=1}^D M^d N^d}, \phi^{e, 1:D(KL - K_s L_s)}, \phi^{s, 1:KL}\}$ and pa-

rameter set $\Theta = \{\alpha^u, \alpha^v, \beta^s, \beta^e\}$. We use Jensen's inequality to acquire the lower bound of the log-likelihood of our proposed probabilistic method:

$$\log P(\boldsymbol{R}^d; \Theta) \geq \mathbb{E}_Q[\log P(\boldsymbol{R}^d, \Omega; \Theta)] - \mathbb{E}_Q[\log Q(\Omega)] \tag{2}$$

$Q(\Omega)$ is the approximating distribution governed by a set of variational parameters $\Omega = \{\gamma^{u,1:\sum_{d=1}^D M^d}, \gamma^{v,1:\sum_{d=1}^D N^d}, \xi^{u,1:\sum_{d=1}^D M^d N^d}, \xi^{v,1:\sum_{d=1}^D M^d N^d},$
$\eta^{e,1:D(KL-K_s L_s)}, \eta^{s,1:KL}\}$. The distance between the lower bound and the true log-likelihood is the KullbackLeibler (KL) divergence. The KL divergence is zero if the distribution $Q(\Omega)$ is equal to the true posterior. Distribution $Q(\Omega)$ should be restricted that it is tractable while at the same time allowing it to provide a good approximation to the true posterior distribution. Usually, mean field theory [10] is used and the distribution $Q(\Omega)$ is:

$$Q(\Omega) = \prod_{k=1}^{K_s} \prod_{l=1}^{L_s} Dir(\eta_{k,l}^s) \prod_{d=1}^{D} \{\prod_{i=1}^{M^d} Dir(\gamma_i^{u,d}) \prod_{j=1}^{N^d} Dir(\gamma_j^{v,d}) \prod_{k=K_s+1}^{K} \prod_{l=1}^{L}$$
$$Dir(\eta_{k,l}^{e,d}) \prod_{k=1}^{K_s} \prod_{l=L_s+1}^{L} Dir(\eta_{k,l}^{e,d}) \prod_{i=1}^{M^d} \prod_{j=1}^{N^d} [Multi(\xi_{i,j}^{u,d}) Multi(\xi_{i,j}^{v,d})]\} \tag{3}$$

The optimization of minimizing the KL-divergence can be done using the following update equations:

$$\gamma_{i,k}^{u,d} = \alpha_k^u + \sum_{j=1}^{N^d} I_{i,j}^d \xi_{i,j,k}^{u,d} \tag{4}$$

$$\gamma_{j,l}^{v,d} = \alpha_l^v + \sum_{i=1}^{M^d} I_{i,j}^d \xi_{i,j,l}^{v,d} \tag{5}$$

In each domain $d \in \{1, ..., D\}$, for $k \in \{1, ..., K_s\}$,

$$\xi_{i,j,k}^{u,d} = exp\{\psi(\gamma_{i,k}^{u,d}) - \psi(\hat{\gamma}_i^{u,d}) + \sum_{l=1}^{L_s} \xi_{i,j,l}^{v,d}(\psi(\eta_{k,l,R_{i,j}^d}^s) - \psi(\hat{\eta}_{k,l,r}^s)) +$$
$$\sum_{l=L_s+1}^{L} \xi_{i,j,l}^{v,d}(\psi(\eta_{k,l,R_{i,j}^d}^{e,d}) - \psi(\hat{\eta}_{k,l,r}^{e,d}))\} \tag{6}$$

where $\hat{\gamma}_i^{u,d} = \sum_{k=1}^K \gamma_{i,k}^{u,d}$, $\hat{\eta}_{k,l,r}^s = \sum_{r=1}^{R_0} \eta_{k,l,r}^s$ and $\hat{\eta}_{k,l,r}^{e,d} = \sum_{r=1}^{R_0} \eta_{k,l,r}^{e,d}$.

For $k \in \{K_s + 1, ..., K\}$,

$$\xi_{i,j,k}^{u,d} = exp\{\psi(\gamma_{i,k}^{u,d}) - \psi(\hat{\gamma}_i^{u,d}) + \sum_{l=1}^{L} \xi_{i,j,l}^{v,d}(\psi(\eta_{k,l,R_{i,j}^d}^{e,d}) - \psi(\hat{\eta}_{k,l,r}^{e,d}))\} \tag{7}$$

$$\eta_{k,l,r}^s = \beta_r^s + \sum_{d=1}^{D} \sum_{i=1}^{M_d} \sum_{j=1}^{N_d} I_{i,j,r}^d \xi_{i,j,k}^{u,d} \xi_{i,j,l}^{v,d} \tag{8}$$

$$\eta_{k,l,r}^{e,d} = \beta_r^e + \sum_{i=1}^{M_d} \sum_{j=1}^{N_d} I_{i,j,r}^d \xi_{i,j,k}^{u,d} \xi_{i,j,l}^{v,d} \tag{9}$$

## 4   Experiments

In this section, the proposed method ProbKT is evaluated. First, we introduce the datasets and the used evaluation metrics in Section 4.1, followed by experimental settings and the baseline methods in Section 4.2. The results of the experiments are presented in Section 4.3.

### 4.1   Datasets and Evaluation Metrics

To test our proposed method, the source domain data and the target domain data are chosen where they are similar but still have divergence between them. The five real-world datasets we used are: EachMovie[1], Movielens1M[2], LibraryThing[3], Amazon Book[4] and YahooMusic[5]. Each of these datasets is publicly available. Numerous experiments are conducted on those in single-domain recommendation methods, but experiments of those datasets on cross-domain recommendation methods are deficient. For AmazonBooks, we removed all users who had given exactly the same rating for every book, as these data are not effective for constructing a recommender system [12]. EachMovie and LibraryThing were normalized to the range of $\{1, 2, 3, 4, 5\}$ before conducting experiments. Refer to [12] about details of the five datasets.

For all the datasets, we filtered out items that are rated less than 10 times and users that who have rated less than 20 items. In our experiment setting, the source domain dataset is more dense than the target domain data. 500 users and 1000 items are randomly chosen for both the source domain and the target domain. But for the target domain, 300 users are randomly selected to be new customers, who are given only 5 observed ratings, and the left are put in the test set. The details of the chosen subsets are listed in Table 1. Three categories are in our chosen datasets, and our recommendation tasks are all the combinations of the three categories.

Evaluation metrics are root mean square error (RMSE) and mean absolute error (MAE): $RMSE = \sqrt{\sum_{u,v,X_{uv} \in Z} \frac{(\hat{X}_{uv} - X_{uv})^2}{|Z|}}$ and $MAE = \sum_{u,v,X_{uv} \in Z} \frac{|\hat{X}_{uv} - X_{uv}|}{|Z|}$, where $Z$ is the test set, and $|Z|$ is the test ratings number.

### 4.2   Baselines and Experimental Settings

We use three non-transfer learning methods and two cross-domain methods for comparison. The non-transfer learning methods were: Pearson's correlation coefficient (PCC) [2], single value decomposition (SVD) [3] and FMM [9].

---

[1] http://www.cs.cmu.edu/~lebanon/IR-lab/data.html#intro
[2] http://grouplens.org/datasets/movielens/1m/
[3] https://www.librarything.com
[4] http://jmcauley.ucsd.edu/data/amazon/
[5] https://webscope.sandbox.yahoo.com/catalog.php?datatype=r

**Table 1.** Description of subsets in five real-word datasets

| Data type | Data source | Domain | Sparsity | Average |
|-----------|-------------|--------|----------|---------|
| Movie | EachMovie | source | 96.00% | 4.32 |
|  | Movielens1M | target | 98.50% | 2.91 |
| book | LibraryThing | source | 87.43% | 3.97 |
|  | AmazonBook | target | 97.87% | 3.13 |
| music | YahooMusic_1 | source | 95.70% | 4.14 |
|  | YahooMusic_2 | target | 97.27% | 2.66 |

The cross-domain methods were: RMGM [5] and CBT [4]. These two cross-domain recommendation methods are all developed without fully considering the domain-shift widely existed in data of two domains. PCC is a user-based CF recommendation method, and the neighborhoods was set to 50. The latent feature number in SVD, FMM, CBT and RMGM was fixed at 40. In SVD, the learning rate was set to 0.003 and the regularization factor was set to 0.015. To be fair, the number of user groups and item groups in our proposed method are both set to be the same at latent feature number in other methods. The number of domain-shared user and item groups is set to be half of the total number of user/item groups. For hyper-parameters, $\alpha^s$ and $\alpha^t$ are set to be 0.1 while $\beta^e$ and $\beta^s$ are set to be 0.2. Twenty random initializations in our experiments are set and the averaged results and standard deviations are reported.

### 4.3   Results

The experiment results of our proposed ProbKT compared with the other five baselines on two accuracy metrics are presented in Table 2, 3 and 4. Overall, ProbKT has the best performance in all the nine tasks. These results indicate that ProbKT can extract common knowledge to share between the source and target domain, which can help increase the recommendation accuracy in the target domain.

Compared with non-transfer learning recommendation methods like PCC, FMM and SVD, experiment results suggest that transfer learning in ProbKT is effective since it can improve the recommendation. The other two cross-domain recommendation methods CBT and RMGM, sometimes fail to improve the performance of recommender systems in the target domain. According to the results, these two methods are largely dependent on their basic method, FMM. If FMM can make accurate recommendation in the target domain, it is likely these two methods can transfer knowledge from the source domain to the target domain. Otherwise, their performance is not as good as the non-transfer learning recommendation method SVD. ProbKT, on the other hand, does not have this concern, due to that the probabilistic model is more flexible and able to fit to different datasets.

**Table 2.** Cross-domain recommendation results on the movie target domain

| Methods | | Source data | MAE | RMSE |
|---|---|---|---|---|
| non-trans | PCC | - | 1.2123 | 1.5722 |
| | FMM | - | 1.3529±0.0025 | 1.6751±0.0019 |
| | SVD | - | 1.0949±0.0049 | 1.3489±0.0035 |
| CDRS | CBT | movie | 1.3772±0.0375 | 1.7152±0.0332 |
| | | book | 1.1640±0.0209 | 1.4429±0.0323 |
| | | music | 1.2094±0.0122 | 1.5177±0.0216 |
| | RMGM | movie | 1.3098±0.0188 | 1.6327±0.0189 |
| | | book | 1.1692±0.0063 | 1.4547±0.0075 |
| | | music | 1.2371±0.0135 | 1.5409±0.0152 |
| | ProbKT | movie | **0.9980±0.0016** | **1.2152±0.0016** |
| | | book | **0.9937±0.0017** | **1.2144±0.0021** |
| | | music | **0.9991±0.0027** | **1.2222±0.0027** |

Compared with the other two cross-domain recommendation methods CBT and RMGM, ProbKT also has better performance. The core part of CBT is to directly apply extracted group-level knowledge from the source domain to the target domain without adaptation or adjustment. RMGM replaced the hard-membership of user/item groups in CBT to soft-membership and relaxes the constraints, which enhances its effectiveness of transferring knowledge. But these two methods cannot properly deal with the divergence existed between two domains. Through the probabilistic method we have proposed in modeling the group-level knowledge in two parts: domain-dependent part and domain-independent part. The domain-independent part is the knowledge that shared between two domains. The domain-dependent part is able to capture the characteristics in each domain and allows the method to be more flexible.

### 4.4 Parameter Analysis and Complexity Analysis

We analyzed how the parameters $K$ and $L$ affect the performance of ProbKT. Due to the space limitation, only the result of movie as source domain and target domain is presented. To analyze $K$ and $L$, grid search is used with evaluation metrics of both MAE and RMSE as shown in Fig. 3. The result of analysis shows that the larger of $K$ and $L$, the better of the performance of ProbKT. This fits to the intuitive fact that more groups of user and items are, the more delicate the probabilistic method can model the users and items. However, the complexity of the method will significantly increase with the increase of $K$ and $L$. The time consumed by different $K$ is shown in Table 5. For simplicity, the setting of $L$ is set to be the same as $K$. The time consumption shown in Table 5 is for 20

**Table 3.** Cross-domain recommendation results on the book target domain

| Methods | | Source data | MAE | RMSE |
|---|---|---|---|---|
| non-trans | PCC | - | 1.1802 | 1.4907 |
| | FMM | - | 1.0274±0.0075 | 1.2631±0.0082 |
| | SVD | - | 1.2117±0.0133 | 1.5283±0.0166 |
| CDRS | CBT | movie | 1.2001±0.0142 | 1.5228±0.0127 |
| | | book | 1.0775±0.0076 | 1.3489±0.0155 |
| | | music | 1.1024±0.0039 | 1.4003±0.0091 |
| | RMGM | movie | 1.0235±0.0077 | 1.2691±0.0108 |
| | | book | 1.0197±0.0064 | 1.2623±0.0102 |
| | | music | 1.0228±0.0043 | 1.2644±0.0094 |
| | ProbKT | movie | **0.9757±0.0044** | **1.1811±0.0049** |
| | | book | **0.9838±0.0044** | **1.1858±0.0050** |
| | | music | **0.9822±0.0054** | **1.1889±0.0070** |

**Table 4.** Cross-domain recommendation results on the music target domain

| Methods | | Source data | MAE | RMSE |
|---|---|---|---|---|
| non-trans | PCC | - | 1.4843 | 1.8539 |
| | FMM | - | 1.3040±0.0060 | 1.5787±0.0053 |
| | SVD | - | 1.4778±0.0134 | 1.7195±0.0181 |
| CDRS | CBT | movie | 1.7532±0.0208 | 2.0958±0.0223 |
| | | book | 1.5843±0.0122 | 1.8599±0.0182 |
| | | music | 1.6251±0.0329 | 1.9246±0.0504 |
| | RMGM | movie | 1.3369±0.0148 | 1.6216±0.0149 |
| | | book | 1.3670±0.0162 | 1.6470±0.0218 |
| | | music | 1.3413±0.0197 | 1.6261±0.0228 |
| | ProbKT | movie | **1.3053±0.0069** | **1.5030±0.0058** |
| | | book | **1.3050±0.0083** | **1.5037±0.0069** |
| | | music | **1.3114±0.0077** | **1.5072±0.0081** |

iterations of the proposed method. For fair comparison with other baselines, we choose 40 for both $K$ and $L$ for ProbKT in our comparison experiments. The total complexity of ProbKT is $O(n)$.

**Table 5.** Time consumption with different settings of $K$

| $K$ | MAE | RMSE | time(s) |
|---|---|---|---|
| $K = 10$ | $1.0464 \pm 0.0072$ | $1.2563 \pm 0.0076$ | 469.15 |
| $K = 20$ | $1.0282 \pm 0.0032$ | $1.2358 \pm 0.0035$ | 566.04 |
| $K = 30$ | $1.0106 \pm 0.0031$ | $1.2198 \pm 0.0026$ | 704.02 |
| $K = 40$ | $0.9981 \pm 0.0023$ | $1.2153 \pm 0.0024$ | 848.16 |
| $K = 50$ | $0.9922 \pm 0.0025$ | $1.2168 \pm 0.0027$ | 1045.00 |
| $K = 60$ | $0.9891 \pm 0.0026$ | $1.2190 \pm 0.0020$ | 1239.54 |
| $K = 70$ | $0.9861 \pm 0.0017$ | $1.2199 \pm 0.0017$ | 1573.37 |
| $K = 80$ | $0.9854 \pm 0.0017$ | $1.2207 \pm 0.0015$ | 2228.76 |
| $K = 90$ | $0.9838 \pm 0.0016$ | $1.2206 \pm 0.0013$ | 2663.10 |
| $K = 100$ | $0.9829 \pm 0.0017$ | $1.2209 \pm 0.0009$ | 4691.15 |



(a) MAE result of different K and L        (b) RMSE result of different K and L
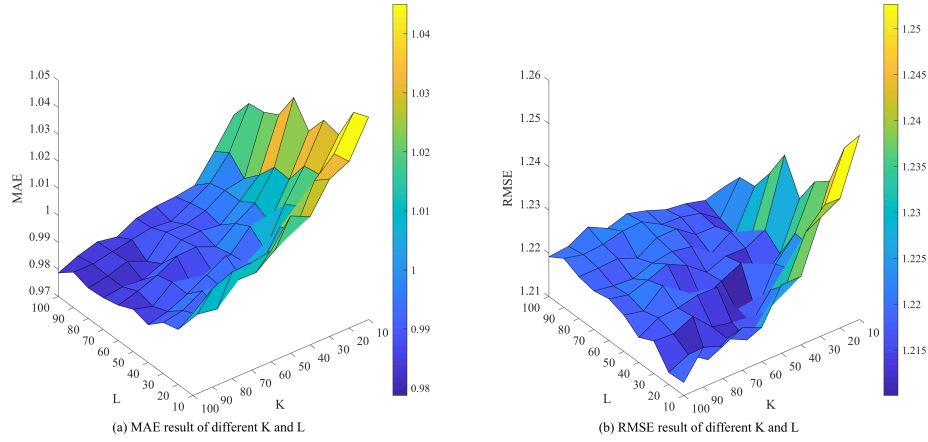
**Fig. 3.** Results with different settings on parameter $K$ and $L$.

## 5   Conclusion

In this paper, we develop a probabilistic method named ProbKT for cross-domain recommendation in multiple domains. Unlike previously developed cross-domain recommendation methods sharing group-level knowledge, our proposed ProbKT method takes both domain-share and domain-specific knowledge into consideration. Through generating group-level knowledge from two different priors, ProbKT relaxes the constrains of previous cross-domain recommendation methods sharing group-level knowledge. In this way, ProbKT is more flexible and is able to deal with data from multiple domains with divergence existed. With the virtue of knowledge extracted from multiple source domains, ProbKT alleviates the data sparsity problem and increases the prediction accuracy in cross-domain recommendation. Experiments on five real-world datasets with nine cross-domain recommendation tasks demonstrate that our method ProbKT

achieves the best performance compared with five baselines including both non-transfer learning and cross-domain recommendation methods. In the future, we will try to develop methods that can deal with heterogeneous data in this problem setting. Also, a Bayesian deep learning will be established as a more delicate method involving various user behaviors.

### Acknowledgment

## References

1. Cremonesi, P., Quadrana, M.: Cross-domain recommendations without overlapping data: myth or reality? In: Proceedings of the 8th ACM Conference on Recommender Systems. pp. 297–300. ACM (2014)
2. Deshpande, M., Karypis, G.: Item-based top-n recommendation algorithms. ACM Transactions on Information Systems **22**(1), 143–177 (2004)
3. Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. Computer **42**(8) (2009)
4. Li, B., Yang, Q., Xue, X.: Can movies and books collaborate? cross-domain collaborative filtering for sparsity reduction. In: IJCAI. vol. 9, pp. 2052–2057 (2009)
5. Li, B., Yang, Q., Xue, X.: Transfer learning for collaborative filtering via a rating-matrix generative model. In: Proceedings of the 26th Annual International Conference on Machine Learning. pp. 617–624. ACM (2009)
6. Lu, J., Behbood, V., Hao, P., Zuo, H., Xue, S., Zhang, G.: Transfer learning using computational intelligence: a survey. Knowledge-Based Systems **80**, 14–23 (2015)
7. Lu, J., Wu, D., Mao, M., Wang, W., Zhang, G.: Recommender system application developments: a survey. Decision Support Systems **74**, 12–32 (2015)
8. Lu, J., Xuan, J., Zhang, G., Luo, X.: Structural property-aware multilayer network embedding for latent factor analysis. Pattern Recognition **76**, 228–241 (2018)
9. Si, L., Jin, R.: Flexible mixture model for collaborative filtering. In: Proceedings of the 20th International Conference on Machine Learning. pp. 704–711 (2003)
10. Xing, E.P., Jordan, M.I., Russell, S.: A generalized mean field algorithm for variational inference in exponential families. In: Proceedings of the 19th Conference on Uncertainty in Artificial Intelligence. pp. 583–591. Morgan Kaufmann Publishers Inc. (2002)
11. Xu, J., Yao, Y., Tong, H., Tao, X., Lu, J.: Rapare: A generic strategy for cold-start rating prediction problem. IEEE Transactions on Knowledge and Data Engineering **29**(6), 1296–1309 (2017)
12. Zhang, Q., Wu, D., Lu, J., Liu, F., Zhang, G.: A cross-domain recommender system with consistent information transfer. Decision Support Systems **104**, 49–63 (2017)
13. Zhao, L., Pan, S.J., Yang, Q.: A unified framework of active transfer learning for cross-system recommendation. Artificial Intelligence **245**, 38–55 (2017)