Cross-Domain Collaborative Filtering with Review Text

Xin Xin[1*], **Zhirun Liu**[1], **Chin-Yew Lin**[2], **Heyan Huang**[1], **Xiaochi Wei**[1], **Ping Guo**[3]
[1]BJ ER Center of HVLIP&CC, School of Comp. Sci., Beijing Institute of Technology, Beijing, China
[2]Microsoft Research Asia, Beijing, China
[3]Image Processing and Pattern Recognition Lab, Beijing Normal University, Beijing, China
{xxin,zrliu}@bit.edu.cn, cyl@microsoft.com, {hhy63,wxchi}@bit.edu.cn, pguo@ieee.org

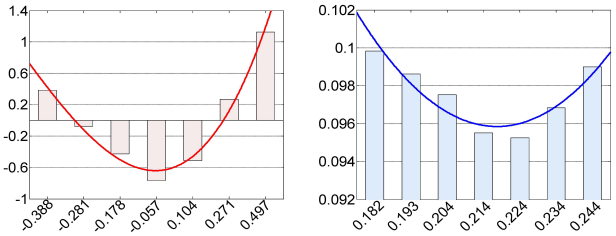
Abstract

Most existing cross-domain recommendation algorithms focus on modeling ratings, while ignoring review texts. The review text, however, contains rich information, which can be utilized to alleviate data sparsity limitations, and interpret transfer patterns. In this paper, we investigate how to utilize the review text to improve cross-domain collaborative filtering models. The challenge lies in the existence of non-linear properties in some transfer patterns. Given this, we extend previous transfer learning models in collaborative filtering, from linear mapping functions to non-linear ones, and propose a cross-domain recommendation framework with the review text incorporated. Experimental verifications have demonstrated, for new users with sparse feedback, utilizing the review text obtains 10% improvement in the AUC metric, and the non-linear method outperforms the linear ones by 4%.

1 Introduction

The cold-start problem [Schein *et al.*, 2002] for new users is one of collaborative filtering (CF)'s inherent limitations for recommender systems. To solve it, cross-domain CF utilizes the user feedback in the auxiliary domain to assist the preference prediction in the target domain [Hu *et al.*, 2013], which has been demonstrated effective in many applications [Li *et al.*, 2009; Pan and Yang, 2013; Singh and Gordon, 2008].

In spite of the significant progress, most existing cross-domain CF methods only model the numerical rating, while ignoring the accompanied review text. The review text, however, contains rich information of items. For example, "Harry Potter" is popular for its story, while "James Bond" is attractive for its action. Dropping such content will aggravate the data sparsity problem. In addition, the learned transfer patterns are not interpretable. In reality, if we know "action" movies are correlated with "rock" music, for a new user in the music domain, who has watched many action movies, it is reasonable to recommend rock music to her/him. But most cross-domain CF models are built on latent vectors generated from the matrix factorization, which are difficult to interpret.



(a) Rating biases (b) Percentage biases

Figure 1: The nonlinear transfer pattern from youth movies to investment books. X-axis: average biases for youth movies for each group. Y-axis: average biases for investment books.

Therefore, the goal of this paper is to investigate how to utilize the review text to improve cross-domain CF models, in order to solve the above limitation.

The challenge we confront is the existence of non-linear properties in some transfer patterns. Figure 1 shows a study of the transfer pattern from "youth" movies to "investment" books. These two categories are representative topics with a topic model [Wang and Blei, 2011] being conducted on movie reviews and book reviews, respectively. The analysis is conducted on 8704 Douban[1] users, who have given at least 18 movie ratings and 18 book ratings. We rank the users by her/his rating bias for youth movies in the ascending order, and divide them into 7 groups. A user's rating bias for youth movies (or investment books) refers to her/his average rating of all reviewed youth movies (or investment books), minus her/his average rating of all movies (or books). For each group, the average rating biases for youth movies and investment books are shown in the left figure. It is observed that as the preference for youth movies increases, the preference for investment books first goes down and then rises. We draw a similar figure based on the percentage bias in the right. A user's percentage bias for youth movies (or investment books) refers to the number of her/her youth movie (or investment book) ratings, divided by the total number of her/his movie (or book) ratings. From the figures, it concludes that the transfer pattern from youth movies to investment books is non-linear. Users who favor or reject youth movies are more likely to take interest in investment books; while moderate

[*]the corresponding author.

[1]http://www.douban.com, a popular review site in China.

users are likely to take less interest. Previous linear cross-domain CF models, however, cannot work well in such cases.

In this paper, we propose a non-linear cross-domain collaborative filtering framework with the review text incorporated. The contribution lies in the following three aspects:

- *Exploring the utility of the review text*. The previously ignored review text has been studied to improve cross-domain CF, complemented with the rating. The rich content alleviates the data sparsity problem, and makes the learned transfer patterns interpretable.
- *Dealing with non-linear transfer patterns*. We extend previous cross-domain CF, from utilizing linear mapping functions to utilizing non-linear ones. The radial basis function (RBF) kernel is employed to map a user's preference vectors between two domains.
- *Real evaluation*. Through experimental verifications in a real-world dataset, we demonstrate for new users, incorporating the review text improves the performance by 10% in the AUC metric, and the proposed non-linear framework outperforms linear ones by 4%.

2 Related Work

2.1 Fundamental Collaborative Filtering

CF algorithms are divided into memory-based [Breese *et al.*, 1998; Ma *et al.*, 2007] and model-based [Koren *et al.*, 2009]. A competitive representative is the factorization-based model [Salakhutdinov and Mnih, 2007; Zhang and Koren, 2007]. Compared with explicit ratings, implicit ratings such as purchase histories, have attracted significant attention in industry [Hu *et al.*, 2008; Rendle *et al.*, 2009; Weimer *et al.*, 2007], because it is easier to obtain from users.

Recently, the review text has been proven effective in improving recommender systems [Diao *et al.*, 2014; McAuley and Leskovec, 2013], where topic models are utilized to analyze the review text, and then incorporated with the matrix factorization to be a joint framework [Agarwal and Chen, 2010; Blei *et al.*, 2003; Wang and Blei, 2011].

2.2 Link-Based Cross-domain CF

The link-based cross-domain CF model links items of different domains, which share similar side information [Berkovsky *et al.*, 2007; Cremonesi *et al.*, 2011; Shapira *et al.*, 2013]. Tags have been typically utilized to bridge these items [Enrich *et al.*, 2013; Shi *et al.*, 2011; Fernndez-Tobłas and Cantador, 2013]. The assumption is that if the active user prefers items with a certain tag in the auxiliary domain, she/he is also likely to prefer items with the same tag in the target domain. For example, users who like a "romantic" movie might also like a "romantic" book.

Differences. The work in this paper does not belong to this category. When dealing with review texts, words across domains might not always have overlaps. In this case, mining transfer patterns between cross-domain review topics is required (e.g., youth movies and investment books), which is the target of this paper, rather than relying on the existence of common words across domains, as in the above work. But our work also has limitation, such as relying on a set of cross-domain users, which the above works do not need.

2.3 Transfer-Based Cross-domain CF

The transfer-based cross-domain CF model aims at mining transfer patterns in modeling the user feedback from multiple domains. Typical methods include collective matrix factorization [Singh and Gordon, 2008], collective SVD [Pan and Yang, 2013], the tensor model [Hu *et al.*, 2013], factorization machines [Loni *et al.*, 2014], etc [Cremonesi and Quadrana, 2014; Li *et al.*, 2009; Tang *et al.*, 2012].

Differences. Although our work belongs to this category, there are two differences. (1) The above algorithms only focus on ratings. As a complement, our work incorporates the review text.(2) The above transfer learning algorithms linearly map the user's cross-domain latent vectors, while our framework utilize non-linear mapping functions.

3 Problem Definition

Users' implicit feedback from two domains is shown in the left matrix of Fig. 2 (a), with the entry "1" denoting that the user has visited the item. The left part of the matrix denotes the target domain, and the right part denotes the auxiliary domain. In the system, some users have feedback in both domains. For items in a single domain, an item-word matrix is utilized to present all reviews, with each entry denoting a word's occurrence time in an item's reviews, as shown in the right of Fig. 2 (a). By defining the cross-domain feedback matrix and the item-word matrices, the problem to be studied is, how to leverage a user's implicit feedback in the auxiliary domain, and the item review texts in both domains, to improve her/his preference predictions in the target domain?

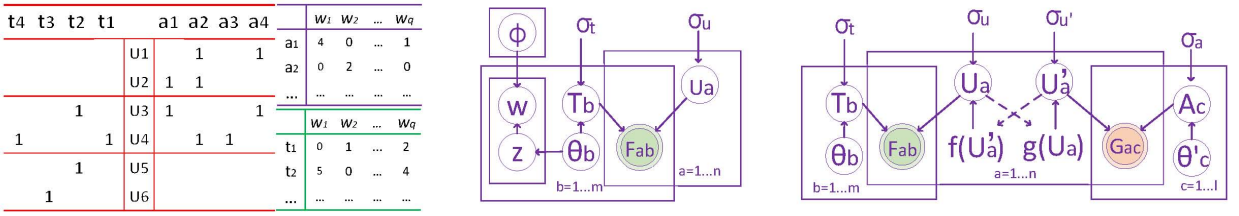
4 Non-linear Transfer Learning Framework

4.1 Collaborative Topic Ranking

In a single domain, we deploy a novel collaborative topic ranking (CTR) model, extended from the collaborative topic modeling (CTM) [Wang and Blei, 2011], to incorporate the review information into factorization based CF models. The previous CTM is designed to deal with explicit ratings, while we confront the implicit feedback. Therefore, a variance of the CTM is designed, by borrowing the ranking-based optimization objective from the Bayesian personalized ranking [Rendle *et al.*, 2009] model.

The intuition of the CTR is to utilize the topic proportion of an item as its feature vector, to substitute for its previous latent vector learned by factorization. Topics are learned from the review corpus. They divide characteristic descriptions of items into categories, presented by word distributions over a fixed vocabulary. A topic proportion is a distribution over all topics. Consequently, an item's topic proportion reveals its characteristics; and the numerical value in each dimension of a user's latent vector reveals her/his preference for the corresponding characteristic. It makes latent feature vectors interpretable. A direct advantage is to alleviate the data sparsity problem for new items. For a new movie as an example, by only obtaining very few reviews, or just utilizing its meta data (actors, genre, etc.), its topic proportion can be learned accurately, and further be utilized for preference predictions.

For the review text, suppose there are m items and k topics. Let ϕ_z denote the word distribution of topic z ($1 \leq z \leq k$),



(a) Problem Definition (b) Collaborative Topic Ranking (c) Non-linear Cross-domain CF Framework

Figure 2: Problem definition and graphical models. The CTR in (b) is a combination of topic analysis (left) and matrix factorization (right); and the proposed non-linear cross-domain CF framework in (c) combines the CTR models in multiple domains, and is jointly learned with the non-linear mapping functions (denoted by dashed lines), in a way of regularization.

and let $\boldsymbol{\theta}_b$ denote the topic proportion of item b's review ($1 \leq b \leq m$). It is assumed that item b's review is generated following the left part of Fig. 2 (b). For each word w, a topic z is first sampled from the proportion $\boldsymbol{\theta}_b$, and then the word is sampled from topic z's word distribution $\boldsymbol{\phi}_z$. Let $\mathcal{C}$ denote the corpus, and H_b denote the word length in item b's review, the likelihood for generating the overall review corpus is

在给定分布参数后，生成所有物品评论的概率

$$p(\mathcal{C}|\boldsymbol{\theta}, \boldsymbol{\phi}) = \prod_{b \in \mathcal{C}} \prod_{j=1}^{H_b} \left(\sum_{z_{bj}=1}^{k} \theta_{b z_{bj}} \phi_{z_{bj} w_{bj}} \right).$$

For the implicit feedback, as shown in the right part of Fig. 2 (b), let n be the user number, and let $\boldsymbol{\mathcal{F}}$ be an $n \times m$ data matrix, whose element F_{ab} is the preference score of user a on item b. This matrix is factorized by two matrices, $\boldsymbol{\mathcal{U}}$ and $\boldsymbol{\mathcal{T}}$. $\boldsymbol{\mathcal{U}}$ is an $n \times k$ matrix, with each row $\boldsymbol{U}_a$ denoting a k-dimensional latent feature vector of user a; and $\boldsymbol{\mathcal{T}}$ is an $m \times k$ matrix, with each row $\boldsymbol{T}_b$ denoting a k-dimensional vector of item b. For user a, if she/he has visited item b, and has not visited item b', we say user a prefers item b more than item b', denoted by $b \succ_a b'$. The probability is defined as

二分类，只有大于和小于 $$p(b \succ_a b' | \boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{T}}) = \frac{1}{1+e^{-(\boldsymbol{U}_a^T \boldsymbol{T}_b - \boldsymbol{U}_a^T \boldsymbol{T}_{b'})}}.$$

Let D denote all the triples of (a, b, b') being observed, the likelihood of observing these triples is

$$p(D|\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{T}}) = \prod_{a,b,b' \in D} p(b \succ_a b' | \boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{T}}).$$

To bridge $\boldsymbol{\theta}_b$ and $\boldsymbol{T}_b$ for an arbitrary item b, a zero-mean Gaussian-distributed offset $\boldsymbol{\epsilon}_b$ is designed between them, which is denoted as

$$\boldsymbol{T}_b = \boldsymbol{\theta}_b + \boldsymbol{\epsilon}_b, \boldsymbol{\epsilon}_b \sim \mathcal{N}(0, \sigma_t^2 \boldsymbol{I}).$$

The offset $\boldsymbol{\epsilon}_b$ models an item's individual bias. When sufficient feedback is obtained, $\boldsymbol{T}_b$ is dominated by both $\boldsymbol{\theta}_b$ and $\boldsymbol{\epsilon}_b$. But when item b is new, $\boldsymbol{\epsilon}_b$ tends to be near 0. Thus $\boldsymbol{T}_b$ is dominated by $\boldsymbol{\theta}_b$, which alleviates the data sparsity problem.

The overall likelihood of the model in Fig. 2 (b) is

$$p(\mathcal{C}, D, \boldsymbol{\mathcal{U}}, \boldsymbol{\epsilon}; \boldsymbol{\theta}, \boldsymbol{\phi}) = p(\mathcal{C}|\boldsymbol{\theta}, \boldsymbol{\phi}) \cdot p(D|\boldsymbol{\mathcal{U}}, \boldsymbol{\theta}, \boldsymbol{\epsilon}) \cdot p(\boldsymbol{\mathcal{U}}) \cdot p(\boldsymbol{\epsilon}),$$

where $p(\boldsymbol{\mathcal{U}})$ is the zero-mean Gaussian prior. The intuition of $p(\mathcal{C}|\boldsymbol{\theta}, \boldsymbol{\phi})$ is that words for the same item tend to be the same topic, by which item characteristics are automatically clustered according to the word occurrence pattern; the intuition of $p(D|\boldsymbol{\mathcal{U}}, \boldsymbol{\theta}, \boldsymbol{\epsilon})$ is to assume that the feedback matrix has the low-rank property, which has been demonstrated effective in modeling user preference patterns; and the intuition of the last two terms are regularization. The joint optimization also has another intuition, that the topics are learned by considering both word occurrence patterns and user preference patterns, as $\boldsymbol{\theta}$ occurs in two terms, which is more reasonable.

$$\begin{aligned} L_{CTR}(\boldsymbol{\mathcal{U}}, \boldsymbol{\epsilon}, \boldsymbol{\theta}, \boldsymbol{\phi}) = & \sum_{b \in \mathcal{C}} \sum_{j=1}^{H_b} \ln \left(\sum_{z_{bj}=1}^{k} \theta_{b z_{bj}} \phi_{z_{bj} w_{bj}} \right) \\ & - \sum_{a,b,b' \in D} \ln \left(1 + \exp(-(\boldsymbol{U}_a^T (\boldsymbol{\theta}_b + \boldsymbol{\epsilon}_b) - \boldsymbol{U}_a^T (\boldsymbol{\theta}_{b'} + \boldsymbol{\epsilon}_{b'})) \right) \\ & - \frac{1}{\sigma_u^2} \sum_{a=1}^{n} \boldsymbol{U}_a^T \boldsymbol{U}_a - \frac{1}{\sigma_t^2} \sum_{b=1}^{m} \boldsymbol{\epsilon}_b^T \boldsymbol{\epsilon}_b. \end{aligned}$$

The log transform of $p(\mathcal{C}, D, \boldsymbol{\mathcal{U}}, \boldsymbol{\epsilon}; \boldsymbol{\theta}, \boldsymbol{\phi})$ is presented in the above equation, which is maximized by a scholastic gradient descent method[2]. In the following steps, we fix the learned $\{\boldsymbol{\theta}, \boldsymbol{\phi}\}$, and adjust other parameters only. Through experiments, jointly tuning them will increase the computational complexity, and obtain only marginal improvements.

4.2 Non-linear User Vector Mapping

利用一个域的向量推出该用户在另一个域的向量

Mapping a user's latent feature vectors in two different domains is the main idea in this paper to bridge the cross-domain implicit feedback. For user a, suppose her/his latent feature vector in the target domain is $\boldsymbol{U}_a$, and the one in the auxiliary domain is $\boldsymbol{U}'_a$. The target is to find mapping functions to transfer them between each other, to simultaneously improve the performances in the two domains. Intuitively, one invertible mapping function between $\boldsymbol{U}_a$ and $\boldsymbol{U}'_a$ is an ideal choice. But as we demonstrated in Fig. 1, by fixing a value in the y-axis, there are two values in the x-axis. Thus invertible functions cannot be found in this case. We propose to find two mapping functions, $\boldsymbol{f}(\boldsymbol{U}'_a) \approx \boldsymbol{U}_a$ and $\boldsymbol{g}(\boldsymbol{U}_a) \approx \boldsymbol{U}'_a$[3]. Consequently, the user feature vector of one domain can be transferred, and then utilized for inferring the

不可逆

[2] Please refer to [Agarwal and Chen, 2009] and [Wang and Blei, 2011] for the detail algorithm.

[3] In practice, normalizing U by $U/\|U\|$ before mapping can obtain slightly better results. We omit this formula for simplicity.

先在单一域计算潜在向量，在计算跨域映射函数

feature vector in the other domain. In this section, we only introduce the formulation of $\boldsymbol{f}(\boldsymbol{U}'_a)$, and the one for $\boldsymbol{g}(\boldsymbol{U}_a)$ is similar.

For simplicity, suppose $\boldsymbol{U}_a$ and $\boldsymbol{U}'_a$ are both k-dimensional vectors. In practice, they are not required to be the same. Then $\boldsymbol{f}(\boldsymbol{U}'_a)$ can be split into a set of k functions, with each function, $f^i(\boldsymbol{U}'_a), i \in \{1, ..., k\}$, mapping $\boldsymbol{U}'_a$ to the i^{th} dimension of $\boldsymbol{U}_a$, denoted by $f^i(\boldsymbol{U}'_a) \approx U^i_a$. We first assume f^i to be linear, and then extend it to be non-linear by the kernel trick. The original linear presentation for f^i is defined as

$$f^i(\boldsymbol{U}'_a) = (\boldsymbol{\omega}^i)^T \cdot \boldsymbol{U}'_a + \beta^i,$$

where $\boldsymbol{\omega}^i$ is the weight vector of $\boldsymbol{U}'_a$'s dimensions. Suppose S is the user set having feedback in both domains. The error between $f^i(\boldsymbol{U}'_a)$ and U^i_a is assumed to follow a zero-mean Gaussian distribution. A zero-mean Gaussian prior is set for $\boldsymbol{\omega}^i$. Consequently, maximizing the likelihood for the mapping error in S is equivalent to finding $\{\boldsymbol{\omega}^i, \beta^i\}$ that minimizes the quadratic errors with regularization terms, as

$$\min_{\boldsymbol{\omega}^i, \beta^i} \frac{1}{2}(\boldsymbol{\omega}^i)^T \boldsymbol{\omega}^i + \gamma \frac{1}{2} \sum_{a \in S} e_a^2,$$
$$s.t.\ U^i_a = f^i(\boldsymbol{U}'_a) + e_a, a \in S.$$

Through the Karush-Kuhn-Tucker (KKT) conditions, the target is equivalent to solving the following linear system

$$\begin{bmatrix} 0 & 1_n^T \\ 1_n & \boldsymbol{K} + \frac{1}{\gamma}\boldsymbol{I} \end{bmatrix} \begin{bmatrix} \beta^i \\ \boldsymbol{\alpha}^i \end{bmatrix} = \begin{bmatrix} 0 \\ \boldsymbol{U}^i \end{bmatrix}, \quad (1)$$

where $\boldsymbol{U}^i$ is an $|S|$-dimension vector, with the a^{th} dimension denoting U^i_a, and $K_{ab} = K(\boldsymbol{U}'_a, \boldsymbol{U}'_b) = \phi(\boldsymbol{U}'_a)^T \phi(\boldsymbol{U}'_b)$ is the Kernel matrix[4]. By the kernel trick, K can be substituted by non-linear functions, where the RBF is chosen, defined as,

$$K(\boldsymbol{U}'_a, \boldsymbol{U}'_b) = \exp\left((\|\boldsymbol{U}'_a - \boldsymbol{U}'_b\|^2)/\sigma^2\right).$$

We successfully find the non-linear presentation of f^i, to map $\boldsymbol{U}'_a$ to U^i_a, formulated as

$$f^i(\boldsymbol{U}'_a) = \sum_{b \in S} \alpha^i_b K(\boldsymbol{U}'_a, \boldsymbol{U}'_b) + \beta^i,$$

where $\boldsymbol{\omega}^i$ is eliminated, and $\{\boldsymbol{\alpha}^i, \beta^i\}$ are the final parameters.

4.3 The Joint Transfer Learning Framework

The graphical model of the proposed joint framework is shown in Fig 2 (c). $\boldsymbol{\mathcal{F}}$ is an $n \times m$ implicit feedback matrix of the target domain, factorized by an $n \times k$ matrix $\boldsymbol{\mathcal{U}}$ and an $m \times k$ matrix $\boldsymbol{\mathcal{T}}$. $\boldsymbol{\mathcal{G}}$ is an $n \times l$ implicit feedback matrix of the auxiliary domain, factorized by an $n \times k$ matrix $\boldsymbol{\mathcal{U}}'$ and an $l \times k$ matrix $\boldsymbol{\mathcal{A}}$. The user feature vector $\boldsymbol{U}_a$ is influenced by the transferred information $\boldsymbol{f}(\boldsymbol{U}'_a)$. Similarly, the user feature vector $\boldsymbol{U}'_a$ is also influenced by $\boldsymbol{g}(\boldsymbol{U}_a)$.

[4]From the Mercer's theorem, K is a symmetric positive semi-definite matrix.

Algorithm 1 Parameter Estimation

Input: The cross-domain feedback matrix and $\{\boldsymbol{\theta}, \boldsymbol{\theta}'\}$
Outputs: $\{\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{U}}', \boldsymbol{\epsilon}, \boldsymbol{\epsilon}', \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\alpha}', \boldsymbol{\beta}'\}$

1: Initialize $\{\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{U}}', \boldsymbol{\epsilon}, \boldsymbol{\epsilon}', \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\alpha}', \boldsymbol{\beta}'\}$
2: **for** each iteration **do**
3: Update $\{\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{U}}', \boldsymbol{\epsilon}, \boldsymbol{\epsilon}'\}$ by the stochastic gradient descent method, according to Eq. 3
4: Update $\{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\alpha}', \boldsymbol{\beta}'\}$ by solving Eq. 1
5: **end for**

In the framework, parameters of the CTR, $\{\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{U}}', \boldsymbol{\epsilon}, \boldsymbol{\epsilon}'\}$, are jointly optimized with parameters of the mapping functions, denoted as $\{\boldsymbol{\alpha}, \boldsymbol{\beta}\}$ for $\boldsymbol{f}$, and $\{\boldsymbol{\alpha}', \boldsymbol{\beta}'\}$ for $\boldsymbol{g}$. By fixing the learned topic proportion for items in both domains, $\{\boldsymbol{\theta}, \boldsymbol{\theta}'\}$, the optimization objective is to maximize the log-likelihood of the joint model, defined as

$$L(\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{U}}', \boldsymbol{\epsilon}, \boldsymbol{\epsilon}'\ \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\alpha}', \boldsymbol{\beta}') = \lambda L_{MAP}(\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{U}}', \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\alpha}', \boldsymbol{\beta}') + (1-\lambda)\left(L_{CTR}(\boldsymbol{\mathcal{U}}, \boldsymbol{\epsilon}|\boldsymbol{\theta}, \boldsymbol{\phi}) + L_{CTR}(\boldsymbol{\mathcal{U}}', \boldsymbol{\epsilon}'|\boldsymbol{\theta}', \boldsymbol{\phi}')\right), \quad (2)$$

$$L_{MAP}(\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{U}}', \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\alpha}', \boldsymbol{\beta}') = -\sum_{a=1}^n \|\boldsymbol{f}(\boldsymbol{U}'_a) - \boldsymbol{U}_a\|^2 - \sum_{a=1}^n \|\boldsymbol{g}(\boldsymbol{U}_a) - \boldsymbol{U}'_a\|^2,$$

$$L_{CTR}(\boldsymbol{\mathcal{U}}, \boldsymbol{\epsilon}|\boldsymbol{\theta}, \boldsymbol{\phi}) = -\frac{1}{\sigma_u^2}\sum_{a=1}^n \boldsymbol{U}_a^T \boldsymbol{U}_a - \frac{1}{\sigma_t^2}\sum_{b=1}^m \boldsymbol{\epsilon}_b^T \boldsymbol{\epsilon}_b - \sum_{a,b,b' \in D} \ln\left(1 + \exp(-(\boldsymbol{U}_a^T(\boldsymbol{\theta}_b + \boldsymbol{\epsilon}_b) - \boldsymbol{U}_a^T(\boldsymbol{\theta}_{b'} + \boldsymbol{\epsilon}_{b'}))\right),$$

单个域的CTR，和两个域的映射　　用主题作为物品潜在向量

Given $\{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\alpha}', \boldsymbol{\beta}'\}$, the parameters of the mapping functions, L_{MAP} can be seen as regularization terms for optimizing the CTR model. It makes sense that $\boldsymbol{U}_a$ (or $\boldsymbol{U}'_a$) should be similar with the transferred vector $\boldsymbol{f}(\boldsymbol{U}'_a)$ (or $\boldsymbol{g}(\boldsymbol{U}_a)$). Given $\{\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{U}}', \boldsymbol{\epsilon}, \boldsymbol{\epsilon}'\}$, the parameters of the CTR model, it is equivalent to the summation of the objectives for each individual mapping function.

Parameter Estimations

The task is to find $\{\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{U}}', \boldsymbol{\epsilon}, \boldsymbol{\epsilon}', \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\alpha}', \boldsymbol{\beta}'\}$ that can maximize the joint optimization objective, defined in Eq. 2. An iterative process is conducted, as shown in Algorithm 1.

Given $\{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\alpha}', \boldsymbol{\beta}'\}$, the parameters of the mapping functions, a stochastic gradient descent method is utilized in searching $\{\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{U}}', \boldsymbol{\epsilon}, \boldsymbol{\epsilon}'\}$. In each step, we sample a triple $(a, b, b') \in D_T$ in the target domain, and a triple $(a, c, c') \in D_A$ in the auxiliary domain. The gradient to $\{\boldsymbol{U}_a, \boldsymbol{\epsilon}_b, \boldsymbol{\epsilon}_{b'}\}$ is calculated as (similar for $\{\boldsymbol{U}'_a, \boldsymbol{\epsilon}'_c, \boldsymbol{\epsilon}'_{c'}\}$)

$$\nabla_{\boldsymbol{U}_a} L_{CTR}(\boldsymbol{\mathcal{U}}, \boldsymbol{\epsilon}) = (1-\lambda)\frac{(\boldsymbol{T}_b - \boldsymbol{T}_{b'})\exp(-\boldsymbol{U}_a^T\boldsymbol{T}_b + \boldsymbol{U}_a^T\boldsymbol{T}_{b'})}{1+\exp(-\boldsymbol{U}_a^T\boldsymbol{T}_b + \boldsymbol{U}_a^T\boldsymbol{T}_{b'})} - \frac{1}{\sigma_u^2}\boldsymbol{U}_a - \lambda(\boldsymbol{U}_a - \boldsymbol{f}(\boldsymbol{U}'_a))$$

$$\nabla_{\boldsymbol{\epsilon}_b} L_{CTR}(\boldsymbol{\mathcal{U}}, \boldsymbol{\epsilon}) = (1-\lambda)\frac{\boldsymbol{U}_a \exp(-\boldsymbol{U}_a^T\boldsymbol{T}_b + \boldsymbol{U}_a^T\boldsymbol{T}_{b'})}{1+\exp(-\boldsymbol{U}_a^T\boldsymbol{T}_b + \boldsymbol{U}_a^T\boldsymbol{T}_{b'})} - \frac{1}{\sigma_t^2}\boldsymbol{\epsilon}_b,$$

$$\nabla_{\boldsymbol{\epsilon}_{b'}} L_{CTR}(\boldsymbol{\mathcal{U}}, \boldsymbol{\epsilon}) = (1-\lambda)\frac{-\boldsymbol{U}_a \exp(-\boldsymbol{U}_a^T\boldsymbol{T}_b + \boldsymbol{U}_a^T\boldsymbol{T}_{b'})}{1+\exp(-\boldsymbol{U}_a^T\boldsymbol{T}_b + \boldsymbol{U}_a^T\boldsymbol{T}_{b'})} - \frac{1}{\sigma_t^2}\boldsymbol{\epsilon}_{b'}. \quad (3)$$

Given $\{\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{U}}', \boldsymbol{\epsilon}, \boldsymbol{\epsilon}'\}$, the parameters of the CTR model, the objective in Eq. 2 is converted to

$$\min_{\alpha,\alpha',\beta,\beta'} \sum_{i=1}^{k}\sum_{a=1}^{n}\left(f^i(\boldsymbol{U}'_a) - U_a^i\right)^2 + \sum_{m=1}^{k}\sum_{b=1}^{n}\left(g^m(\boldsymbol{U}_b) - U_b'^m\right)^2,$$

Due to the independency of the $2\times k$ functions, minimizing the summation is equivalent to minimizing each individual independently. Thus the optimization is converted to solve the linear system in Eq. 1, where the three-level learning method in Suykens's book [Suykens *et al.*, 2002] is employed. In practice, to reduce the noise and the complexity, only a set of users S who have dense feedback in the two domains is utilized for learning the mapping functions.

Complexity Analysis

The complexity for the stochastic gradient descent in each iteration is $O(k)$, where k is the dimension of latent vectors and is also the number of topics. The complexity for learning the mapping function is $O(|S|^2)$, where $|S|$ is the number of cross-domain dense users. We set $k = 15$, $|S| = 1000$ empirically, which will be discussed in the experiments. From Eq. 1, learning a mapping function is equivalent to the least squares support vector machines with $k = 15$ features. 1000 instances are sufficient and efficient through verifications.

5 Experiments

5.1 Experimental Setup

The dataset is crawled from Douban, a review site in China, which contains both book reviews and movie reviews. 8,704 users are finally crawled, who have 3,769,055 visits on 9,420 movies, and 1,023,742 visits on 9,268 books. Each user has at least 18 visits in both domains. The top 20 reviews for each item are crawled, with stopwords discarded. Detailed statistics of the dataset are provided in Table 1.

Following Rendle's work [Rendle *et al.*, 2009], we utilize the area under the ROC curve (AUC) to evaluate performances of different models. The larger the value is, the better the performance is. Users' feedback is divided into two disjoint sets, S_{train} and S_{test}. The average AUC is calculated[5] as

$$AUC = \frac{1}{|U|}\sum_u \frac{1}{|E(u)|}\sum_{(a,b)\in E(u)} \delta(\boldsymbol{U}_u^T\boldsymbol{T}_a > \boldsymbol{U}_u^T\boldsymbol{T}_b)$$
$$E(u) := \{(a,b)|(u,a)\in S_{test} \wedge (u,b)\notin (S_{test}\cup S_{train})\}.$$

We implement the following baselines, including (1) Popularity, (2) BPR+MF [Rendle *et al.*, 2009], (3) BPR+CMF [Singh and Gordon, 2008], (4) BPR+Tensor [Hu *et al.*, 2013], and (5) BPR+CSVD [Pan and Yang, 2013]. The last three methods are cross-domain CF algorithms, which utilize the feedback from both domains. The proposed model is divided into four variations: (1) The proposed framework (CTR+RBF); (2) RBF kernel is replace by linear regression (CTR+Li); (3) all review topics are removed (BPR+RBF); and (4) a single-domain CF with review incorporated (CTR). We set $\frac{1}{\sigma_u^2} = \frac{1}{\sigma_{u'}^2} = \frac{1}{\sigma_t^2} = \frac{1}{\sigma_a^2} = 0.1$, $k = 15$, $\lambda = 0.5$, $\sigma^2 = 2.5$ and $\gamma = 500$. 80% of users are randomly selected for training, and the others are for testing.

[5] $\delta(x) = 1$ if $x = true$; or $\delta(x) = 0$ if $x = false$.

Table 1: Statistics of the dataset

	Book		Movie	
	user	book	user	movie
Min. #feedback	18	1	18	1
Max. #feedback	2,033	3,612	3,257	6,511
Avg. #feedback	116.7	109.6	433.0	400.1
Avg. #word	–	1,145	–	2,335
Total #word	–	10M	–	22M

Table 3: The learned topics from the two domains

Topics of the movie domain					
Politics	Youth	Wars	Romantic	Horror	Cop
history	music	human	girl	death	police
government	youth	war	marriage	ghost	gun
politics	dream	hero	love	horror	murder
British	girl	earth	lose	mother	killer
freedom	memory	Japan	beauty	doctor	crime
Topics of the book domain					
Politics	Investment	Novel	Suspense	Education	Food
soviet	customer	Harry	murder	peking	taipei
civics	invest	prejudice	Sherlock	abroad	kitchen
socialism	economics	Rowling	sanctum	code	milk
despotism	Web	Hiyawu	Higashino	plagiarism	egg
Nepoleon	sale	Rochester	crime	graduate	corn

In choosing parameters of the proposed model and the baselines, we traverse them in a common range, and select the best performance through cross validation for comparisons. For example, $k = 15$ achieves the best performance of the proposed framework, while in CSVD, $k = 35$ achieves the best performance, which is much larger. This is consistent with the discussion in [McAuley and Leskovec, 2013].

5.2 Overall Performance

To simulate cold-start users, for each test user, we randomly select x implicit ratings (x ranges from 0 to 7 respectively) to learn feature vectors, and evaluate performances of different models. Table 2 shows the overall performances. Top rows are the results when book is the target domain, and bottom rows are the results when movie is the target domain. Imp. 1 denotes the relative improvements over the best of the five baselines; and Imp. 2 denotes the relative improvements over the CTR-Li method. It is observed that the proposed framework with the review text incorporated outperforms previous baselines by around 10%. Non-linear models constantly performs better than linear models, by around 4%.

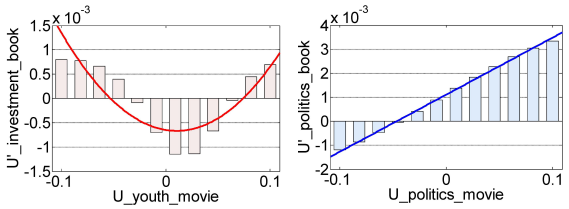
5.3 Interpretations

Table 3 shows the top words of some learned topics, from the review texts in both domains, which helps us to understand the user latent vectors. To analyze the insight transfer patterns, we study the mapping relation between cross-domain topic pairs. For example, we manually set a user latent vector in the movie domain, with other dimensions being 0, and adjust the corresponding dimension of "youth movies" in the range of $[-0.1, 0.1]$. Through the learned mapping function, we observe the mapping value in the book domain, on the dimension of "investment books". Figure 3 (a) shows the result. It is observed that as the value in "youth movies" goes up, the value in "investment books" first goes down, and then

Table 2: Overall performances of different methods for cold-start users

线性和非线性提升的性能

Target	#Train	Popularity	BPR+MF	BPR+CMF	BPR+Tensor	BPR+CSVD	BPR+RBF	CTR	CTR-Li	CTR-RBF	Imp. 1	Imp. 2
Book	0	0.6168	0.6153	0.6396	0.6460	0.6519	0.6653	0.6189	0.6879	**0.7180**	10.14%	4.38%
	1	0.6168	0.6321	0.6477	0.6597	0.6531	0.6737	0.6750	0.6978	**0.7267**	10.16%	4.14%
	2	0.6166	0.6482	0.6606	0.6685	0.6728	0.6823	0.6939	0.7035	**0.7377**	9.64%	4.85%
	3	0.6168	0.6524	0.6707	0.6844	0.6801	0.6931	0.7061	0.7141	**0.7429**	8.55%	4.03%
	4	0.6167	0.6595	0.6773	0.6876	0.6930	0.7028	0.7185	0.7176	**0.7505**	8.30%	4.59%
	5	0.6162	0.6639	0.6850	0.6964	0.7023	0.7097	0.7239	0.7208	**0.7593**	8.12%	5.34%
	6	0.6168	0.6702	0.6876	0.7028	0.7064	0.7136	0.7325	0.7307	**0.7644**	8.22%	4.61%
	7	0.6163	0.6781	0.6957	0.7091	0.7161	0.7251	0.7372	0.7420	**0.7697**	7.49%	3.74%
Movie	0	0.6413	0.6416	0.6435	0.6446	0.6480	0.6623	0.6422	0.6691	**0.7086**	9.357%	5.91%
	1	0.6412	0.6423	0.6506	0.6545	0.6528	0.6687	0.6695	0.6771	**0.7120**	8.784%	5.15%
	2	0.6411	0.6437	0.6560	0.6617	0.6596	0.6720	0.6763	0.6884	**0.7198**	8.790%	4.57%
	3	0.6410	0.6459	0.6592	0.6637	0.6623	0.6791	0.6832	0.6948	**0.7167**	7.988%	3.15%
	4	0.6409	0.6477	0.6619	0.6678	0.6679	0.6831	0.6906	0.7038	**0.7252**	8.585%	3.05%
	5	0.6408	0.6499	0.6634	0.6688	0.6732	0.6883	0.6928	0.7088	**0.7301**	8.447%	3.01%
	6	0.6409	0.6531	0.6708	0.6740	0.6788	0.6994	0.7010	0.7143	**0.7363**	8.468%	3.08%
	7	0.6406	0.6572	0.6761	0.6798	0.6849	0.7068	0.7119	0.7210	**0.7434**	8.533%	3.10%



(a) Non-linear transfer patterns (b) Linear transfer patterns

Figure 3: The interpretation of transfer patterns.

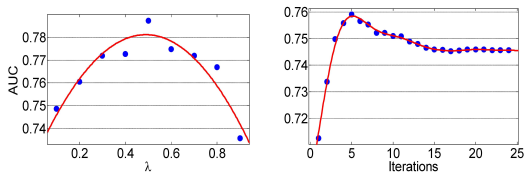
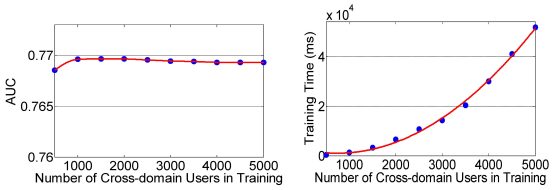


Figure 4: Parameter and convergence analysis.

rises. This exactly matches the study in the introduction, indicating that non-linear transfer patterns can be learned by the proposed framework. Figure 3 (b) shows the learned mapping relation from "politics movies" to "politics books". This time, the learned mapping relation becomes a linear one. It also makes sense, because these two topics are correlated. This analysis explains the insight advantage of the RBF kernel.

5.4 Parameter and Convergence Analysis

λ is the parameter in Eq. 2, balancing the weights of the CTR model and the mapping functions. From Fig. 4 (left), performances are not sensitive with λ being changed. Figure 4 (right) shows the convergence. An iteration means an exchange between learning the CTR parameters and the mapping function parameters. Within an exchange, tens of thousands sub-iterations are conducted for the stochastic gradient descent method of the CTR model. According to the figure, we set the iteration number to be 5 in the experiments.



(a) Performances (b) Training time

Figure 5: Changing the size of cross-domain users in training, with book as the target domain, and #Train=7.

选取太多用户，不会明显提高准确性，反而会增大训练时间

5.5 How to Choose the Cross-domain Dense Users

In choosing the size of the cross-domain users for learning mapping functions, we randomly selected 1,963 users from the training set to form a validation set, and ranked the remained 5,000 users in the descending order according to their minimum feedback number in the two domains. From the top 500 users to all 5,000 users, we add them gradually and observe the performances and the training time. We repeat this process for 5 times, and the average results are shown in Fig. 5. It is observed that performances are not sensitive with the setting being changed. If the size is large, the included sparse users slightly impairs the performance, and the complexity increases; while if it is small, there are insufficient training data. By considering both accuracy and complexity, we select the top 1000 users in practice. In previous work based on linear mappings, due to the linear time complexity, selecting dense users is not needed, and all users are utilized.

6 Conclusion

We have proposed a non-linear transfer learning framework, to incorporate the review text for improving the cross-domain recommendation. For users with sparse implicit feedback, the proposed framework outperforms previous methods without the review text by 10% in the AUC metric, and the non-linear mapping functions outperforms linear ones by 4%.

Acknowledgments

The work described in this paper was mainly supported by the National Basic Research Program of China (973 Pro-

gram, Grant No. 2013CB329605), the National Natural Science Foundation of China (No. 61300076, No. 61375045), and the Ph.D. Programs Foundation of Ministry of Education of China (No. 20131101120035).

References

[Agarwal and Chen, 2009] Deepak Agarwal and Bee-Chung Chen. Regression-based latent factor models. In *Proc. of SIGKDD'09*, pages 19–28. ACM, 2009.

[Agarwal and Chen, 2010] D. Agarwal and B.C. Chen. flda: matrix factorization through latent dirichlet allocation. In *Proc. of WSDM'10*, pages 91–100. ACM, 2010.

[Berkovsky *et al.*, 2007] Shlomo Berkovsky, Tsvi Kuflik, and Francesco Ricci. Cross-domain mediation in collaborative filtering. In *User Modeling 2007*. 2007.

[Blei *et al.*, 2003] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[Breese *et al.*, 1998] J.S. Breese, D. Heckerman, C. Kadie, et al. Empirical analysis of predictive algorithms for collaborative filtering. In *Proc. of UAI'98*, pages 43–52, 1998.

[Cremonesi and Quadrana, 2014] Paolo Cremonesi and Massimo Quadrana. Cross-domain recommendations without overlapping data: myth or reality? In *Proc. of RecSys'14*, pages 297–300. ACM, 2014.

[Cremonesi *et al.*, 2011] Paolo Cremonesi, Antonio Tripodi, and Roberto Turrin. Cross-domain recommender systems. In *Proc. of ICDMW'11*, pages 496–503. IEEE, 2011.

[Diao *et al.*, 2014] Qiming Diao, Minghui Qiu, Chao-Yuan Wu, Alexander J Smola, Jing Jiang, and Chong Wang. Jointly modeling aspects, ratings and sentiments for movie recommendation (jmars). In *Proc. of SIGKDD'14*, 2014.

[Enrich *et al.*, 2013] Manuel Enrich, Matthias Braunhofer, and Francesco Ricci. Cold-start management with cross-domain collaborative filtering and tags. In *E-Commerce and Web Technologies*, pages 101–112. Springer, 2013.

[Fernndez-Tobłas and Cantador, 2013] I. Fernndez-Tobłas and I. Cantador. Exploiting social tags in matrix factorization models for cross-domain collaborative filtering. In *Proc. of the 1st Intl. Workshop on New Trends in Content-based Recommender Systems (2013)*, 2013.

[Hu *et al.*, 2008] Yifan Hu, Yehuda Koren, and Chris Volinsky. Collaborative filtering for implicit feedback datasets. In *Proc. of ICDM'08*, pages 263–272. IEEE, 2008.

[Hu *et al.*, 2013] Liang Hu, Jian Cao, Guandong Xu, Longbing Cao, Zhiping Gu, and Can Zhu. Personalized recommendation via cross-domain triadic factorization. In *Proc. of WWW'13*, pages 595–606, 2013.

[Koren *et al.*, 2009] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[Li *et al.*, 2009] Bin Li, Qiang Yang, and Xiangyang Xue. Can movies and books collaborate? cross-domain collaborative filtering for sparsity reduction. In *Proc. of IJCAI'09*, volume 9, pages 2052–2057, 2009.

[Loni *et al.*, 2014] Babak Loni, Yue Shi, Martha Larson, and Alan Hanjalic. Cross-domain collaborative filtering with factorization machines. In *Advances in Information Retrieval*, pages 656–661. Springer, 2014.

[Ma *et al.*, 2007] Hao Ma, Irwin King, and Michael R Lyu. Effective missing data prediction for collaborative filtering. In *Proc. of SIGIR'07*, pages 39–46. ACM, 2007.

[McAuley and Leskovec, 2013] Julian McAuley and Jure Leskovec. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proc. of RecSys'13*, pages 165–172. ACM, 2013.

[Pan and Yang, 2013] Weike Pan and Qiang Yang. Transfer learning in heterogeneous collaborative filtering domains. *Artificial Intelligence*, 197:39–55, 2013.

[Rendle *et al.*, 2009] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proc. of UAI'09*, pages 452–461. AUAI, 2009.

[Salakhutdinov and Mnih, 2007] Ruslan Salakhutdinov and Andriy Mnih. Probabilistic matrix factorization. In *Proc. of NIPS'07*, pages 1257–1264, 2007.

[Schein *et al.*, 2002] Andrew I. Schein, Alexandrin Popescul, Lyle H. Ungar, and Pennock David M. Methods and metrics for cold start recommendations. In *Proc. of SIGKDD'02*. ACM, 2002.

[Shapira *et al.*, 2013] Bracha Shapira, Lior Rokach, and Shirley Freilikhman. Facebook single and cross domain data for recommendation systems. *User Modeling and User-Adapted Interaction*, 23(2-3):211–247, 2013.

[Shi *et al.*, 2011] Yue Shi, Martha Larson, and Alan Hanjalic. Tags as bridges between domains: Improving recommendation with tag-induced cross-domain collaborative filtering. In *User Modeling, Adaption and Personalization*, pages 305–316. Springer, 2011.

[Singh and Gordon, 2008] Ajit P Singh and Geoffrey J Gordon. Relational learning via collective matrix factorization. In *Proc. of SIGKDD'08*, pages 650–658. ACM, 2008.

[Suykens *et al.*, 2002] Johan AK Suykens, Tony Van Gestel, Jos De Brabanter, Bart De Moor, Joos Vandewalle, JAK Suykens, and T Van Gestel. *Least squares support vector machines*, volume 4. World Scientific, 2002.

[Tang *et al.*, 2012] Jie Tang, Sen Wu, Jimeng Sun, and Hang Su. Cross-domain collaboration recommendation. In *Proc. of SIGKDD'12*, pages 1285–1293. ACM, 2012.

[Wang and Blei, 2011] C. Wang and D.M. Blei. Collaborative topic modeling for recommending scientific articles. In *Proc. of SIGKDD'11*, pages 448–456. ACM, 2011.

[Weimer *et al.*, 2007] Markus Weimer, Alexandros Karatzoglou, Quoc V Le, and Alex J Smola. Cofi rank-maximum margin matrix factorization for collaborative ranking. In *Proc. of NIPS'07*, pages 1593–1600, 2007.

[Zhang and Koren, 2007] Yi Zhang and Jonathan Koren. Efficient bayesian hierarchical user modeling for recommendation system. In *Proc. of SIGIR'07*, pages 47–54. ACM, 2007.