A unified framework of active transfer learning for cross-system recommendation

Zhao, Lili; Pan, Sinno Jialin; Yang, Qiang

2017

Zhao, L., Pan, S. J., & Yang, Q. (2017). A unified framework of active transfer learning for cross-system recommendation. Artificial Intelligence, 245, 38-55.

https://hdl.handle.net/10356/83227

https://doi.org/10.1016/j.artint.2016.12.004

A Unified Framework of Active Transfer Learning for Cross-System Recommendation

Lili Zhao[†], Sinno Jialin Pan[‡], and Qiang Yang[†]

[†]*Hong Kong University of Science and Technology, Hong Kong*
[‡]*Nanyang Technological University, Singapore*
conquer.zhao@gmail.com, sinnopan@ntu.edu.sg, qyang@cse.ust.hk

Abstract

In the past decade, artificial intelligence (AI) techniques have been successfully applied to recommender systems employed in many e-commerce companies, such as Amazon, eBay, Netflix, etc., which aim to provide personalized recommendations on products or services. Among various AI-based recommendation techniques, collaborative filtering has proven to be one of the most promising methods. However, most collaborative-filtering-based recommender systems, especially the newly launched ones, have trouble making accurate recommendations for users. This is caused by the data sparsity issue in recommender systems, where little existing rating information is available. To address this issue, one of the most effective practices is applying transfer learning techniques by leveraging relatively rich collaborative data knowledge from related systems, which have been well running. Previous transfer learning models for recommender systems often assume that a sufficient set of entity correspondences (either user or item) across the target and auxiliary systems (a.k.a. source systems) is given in advance. This assumption does not hold in many real-world scenarios where entity correspondences across systems are usually unknown, and the cost of identifying them can be expensive. In this paper, we propose a new transfer learning framework for recommender systems, which relaxes the above assumption to facilitate flexible knowledge transfer across different systems with low cost by using an active learning principle to construct entity correspondences across systems. Specifically, for the purpose of maximizing knowledge transfer, we first iteratively select entities in the target system based on some criterion to query their correspondences in the source system. We then plug the actively constructed entity correspondences into a general transferred collaborative-filtering model to improve recommendation quality. Based on the framework, we propose three solutions by specifying three

state-of-the-art collaborative filtering methods, namely Maximum-Margin Matrix Factorization, Regularized Low-rank Matrix Factorization, and Probabilistic Matrix Factorization. We perform extensive experiments on two real-world datasets to verify the effectiveness of our proposed framework and the three specified solutions for cross-system recommendation.

Keywords: transfer learning, active learning, recommender systems

1. Introduction

With the development and explosion of Web 1.0 and 2.0 technologies, recommender systems have been part of the Internet in the past two decades to provide recommendations on items, e.g., products or services. The goal of recommender systems is to suggest personalized items that match the interests of each specific user [1].[1] To understand users' interests, a typical method is to ask users to fill their personal information and answer some predefined questions, and then summarize the information to build specific profiles for each user manually. With the built user profiles, one can recommend relevant items to each user. Instead of manually conducting surveys to build users profiles and generating recommended item list, modern recommender systems have been adopting various artificial intelligence (AI) techniques to learn users profiles, predict users' intensions, and recommend items of interest automatically. In general, commonly used AI techniques for recommender systems include collaborative filtering (CF), content-based filtering, case-based reasoning, constraint satisfaction with a domain-dependent knowledge base, etc. [2, 3]. Among these AI techniques, CF, especially matrix factorization, has proven to be one of the most promising methods, and been successfully used in many commercial recommender systems.

The goal of CF-based recommender systems is to generate item recommendations for a user by utilizing the observed preferences of other users whose historical behaviors are correlated with those of the target user. However, most CF-based recommender systems perform poorly when little collaborative information, e.g., historical ratings on items, is available. This is referred to as the data sparsity problem, which is one of the most common and major challenging problems in many newly launched recommender systems. To address the data sparsity

[1] In a broad definition, any software system that provides suggestions on items to purchase, to subscribe, to use, or to invest can be regarded as a recommender system. In this sense, computational advertising, query suggestion, etc., can be also seen as examples of recommendations.

problem, transfer learning has been proposed by exploiting auxiliary collaborative data from some related recommender system(s). In a nutshell, transfer learning is a promising paradigm of machine learning, which aims to adapt a predictive model across different domains or tasks with little additional human supervision by extracting and transferring common knowledge across the source and the target domains or tasks [4, 5]. To make a success of transfer learning, discovering commonality between the source and the target domains or tasks is crucial. A motivation behind applying transfer learning to recommender systems is that many users are active in multiple social medias (e.g., Twitter, Facebook, etc.), or purchasing products from various e-commercial websites (e.g., Amazon, Taobao, etc.), thus, a source CF model built with rich collaborative data can be compressed by identifying a same set of users across different websites as a prior to assist the training of a more precise CF model for the target recommender system [6, 7]. This approach is also known as cross-system CF.

Previous transfer-learning approaches to cross-system CF can be classified into two categories: 1) CF methods *with* cross-system entity correspondences, and 2) CF methods *without* cross-system entity correspondences. In the former category, Mehta and Hofmann [8] and Pan *et al.* [9, 10], respectively, proposed to embed the cross-system entity correspondences as constraints to jointly learn the CF models for the source and the target recommender systems with an aim to improve the performance of the target CF system. Although these approaches have shown promising results, they require the existence of fully or sufficient entity-correspondence mappings, i.e., user correspondence or item correspondence, across different systems. This strong prerequisite is often difficult to satisfy in most real-world scenarios, as some specific users or items in one system may be missing in other systems. For example, user populations of Twitter and Facebook services are overlapping, but not identical, as is the case with Amazon and eBay. In addition, even though there may exist potential entity-correspondence mappings between different systems, they may be expensive or time-consuming to be recognized as users may use different names, or an item may be named differently in different online commercial systems. As illustrated in Figure 1, we have two movie recommender systems A and B. In general, different movies may share a same name. For instance, the movie "The Island" can be referred to as a American science fiction/thriller film released in 2005, or a Russian biographical film about a fictional 20th century Eastern Orthodox monk released in 2006. In this case, for the movie "The Island" in system A, we are not sure which version of the movie "The Island" in system B is its correspondence. Therefore, to identify whether two movies are corresponding to each other, one

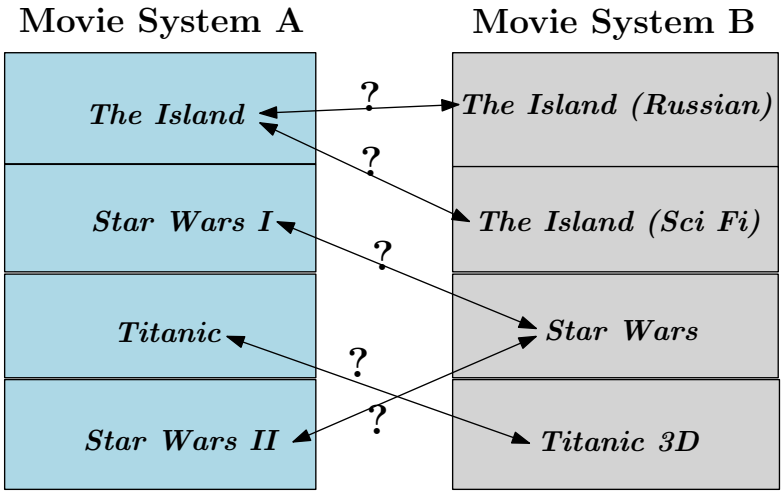


Figure 1: An example of two recommender systems with ambiguous movies names.

may need to compare their meta data or even need to watch the trailers if the meta information is missing, which can be very time-consuming.

In the latter category of approaches where no assumption is made on pre-existing cross-system mappings, researches have been focused on capturing the group-level behaviors of users for knowledge transfer. For example, Li *et al.* [6] proposed a codebook-based-transfer (CBT) method for cross-system CF. The main assumption of CBT is that specific users or items may be different across systems, but the groups of them, e.g., based on interests, ages, etc., should behave similarly. Therefore, CBT aims to first generate a set of cluster-level user-item rating patterns from the source system, which is referred to as a codebook. The codebook can then be used as a prior for learning a CF model for the target system. Li *et al.* [7] further proposed a probabilistic model for cross-system CF that shares a similar motivation with CBT. Compared to the approaches in the former category, which make use of cross-system entity correspondences as a bridge, however, these approaches are less effective for knowledge transfer across recommender systems.

In this paper, we assume that the cross-system entity correspondences are unknown in general, but that the mappings can be identified with cost. Based on this assumption, we propose a unified framework to actively construct entity-correspondence mappings across recommender systems, where a flexible transfer learning approach with partial entity-correspondence mappings between systems and a strategy for actively constructing cross-system correspondences are inte-

grated. To be specific, the proposed framework consists of two main components:

- an active learning approach to constructing entity correspondences across systems in an iterative manner, and
- an extended matrix factorization approach to cross-system CF that flexibly leverages partial entity-correspondence mappings as a bridge for knowledge transfer.

The proposed framework is general, where various extended matrix factorization methods in a transfer learning manner can be integrated. In this paper, we offer three specific solutions by extending and plugging three well-known CF methods, namely Maximum-Margin Matrix Factorization (MMMF), Regularized Low-rank Matrix Factorization (RLMF), and Probabilistic Matrix Factorization (PMF) into the framework. Specifically, we extend the three popular CF methods in the transfer learning manner such that they can flexibly transfer knowledge across domains with partial cross-domain entity correspondences, and propose specific active learning strategies on top of the extended CF methods to actively construct entity correspondences across domains. Note that matrix factorization based CF is a rapidly advancing research area, where several new techniques are proposed every year. According to a recent comparative study of CF algorithms [11], various state-of-the-art matrix factorization methods are comparable to each other in general. Some methods work better on relatively smaller datasets, while others work better on larger datasets, in terms of user or item size. Some methods work better when historical ratings of items by users are sparse, while others work better when there are sufficient historical ratings. In this work, our focus is not discussing which matrix factorization method can be adapted into our framework to achieve the best performance for knowledge transfer across different recommender systems, but providing a general active transfer learning framework, where researchers can extend their favor matrix factorization methods for different applications and datasets.

Compared to our previous work [12], the contributions of this paper are summarized as follows.

1. We generalize the MMMF-based active transfer learning method proposed in [12] to a unified framework.
2. Based on the framework, we further specify two more solutions based on other well-known CF methods, namely RLMF and PMF.
3. We conduct extensive experiments to verify the effectiveness of the proposed active transfer learning framework for cross-system CF.

The rest of this paper is organized as follows. In the following section, we start by reviewing some related work. In Section 3, we introduce the notations and preliminaries used in this paper. In Section 4, we first describe the proposed framework at a high level, and then present three specific solutions in detail. After that we show experiments that are conducted on two real-world datasets to verify the effectiveness of the proposed framework and the three specific solutions in Section 5. Finally, we conclude our work in Section 6.

2. Related Work

Recommender systems emerged as an independent research area in the mid-1990s [13, 14], and attracted more and more attention from both academia and industry since the Netflix competition[2] held between 2006 and 2009. Besides products/services recommendation for e-commerce, recommender systems have been employed in many other application areas, such as configuration for products design [15], requirement engineering [16], music recommendation [17], tag recommendation in the social web [18], etc. Among various techniques for recommender systems, AI-based methods, specially CF methods through matrix factorization, have proven to be effective and promising [19, 11].

Besides the work mentioned in Section 1, there is some other related work on applying transfer learning to cross-system CF. For instance, Pan *et al.* [20] proposed an approach known as TIF (Transfer by Integrative Factorization) to integrate the auxiliary uncertain ratings, which are distributions of ratings instead of exact point-wise scores, as constraints into the target matrix factorization problem. Cao *et al.* [21] and Zhang *et al.* [22], respectively, extended the Collective Matrix Factorization (CMF) [23] method to solve multi-domain CF problems in a multi-task learning manner. In their work, a CF model is learned for each domain by exploiting the correlations or similarities among multiple domains. Tang *et al.* [24] applied collective learning techniques for multi-domain Web search with implicit feedbacks. However, most of the existing methods require that the entities, either users or items, across different systems to be identical, and the correspondences between domains are given in advance. As we discussed in the previous section, this assumption is impractical in many real-world scenarios.

Our work is also related to some previous work on active learning for CF [25, 26, 27, 28, 29], which aims to solve the sparsity problem in CF by actively querying users to give ratings on selected items. A common assumption behind this

[2] `http://www.netflixprize.com`

direction of work is that the users quered by an active learner are always able to provide ratings on the selected items in the system. However, in many real-world scenarios, this assumption is hard to satisfy because users may only be familiar with some items in the system, and may fail to provide ratings on the actively selected items. For example, a user may only watched a few movies, and is not able to provide accurate ratings on other movies that he/she has not watched. Alternatively, in this paper, instead of actively asking users to give ratings on selected items, we propose to actively construct cross-system entity correspondence mappings as a bridge for knowledge transfer across recommender systems to solve the data sparsity problem in the target recommender system.

Another related research area is to develop a unified framework for active learning and transfer learning. Previous work in this research direction is focused on how to actively query instances in the target domain for labels in order to learn a target predictive model by leveraging source domain knowledge [30, 31, 32, 33, 34, 35, 36]. Existing methods can be classified into three categories: 1) performing transfer and active learning once, respectively, in a pipeline [37, 30, 31], 2) performing transfer and active learning alternatingly [32, 33, 34, 36], and 3) integrating transfer and active learning into a unified optimization problem [35]. Most of them are focused on classification or regression problems. Different from previous work, in this paper, our study on active transfer learning is focused on addressing the data sparsity problem for CF instead of traditional classification or regression problems. Moreover, the proposed active learning strategy aims to construct entity correspondences between systems instead of querying a "class label" for an instance. Therefore, existing methods on combining active learning and transfer learning cannot be directly applied to our problem.

3. Notations & Preliminaries

Denote by $\mathcal{D}$ the target CF task, which is associated with an extremely sparse preference matrix $\mathbf{X}^{(d)} \in \mathbb{R}^{m_d \times n_d}$, where m_d is the number of users and n_d is the number of items. Each entry $x_{uv}^{(d)}$ of $\mathbf{X}^{(d)}$ corresponds to user u's preference on item v. If $x_{uv}^{(d)} \neq 0$, it means for user u, the preference on item v is observed, otherwise unobserved. Let $\mathcal{I}^{(d)}$ be the set of all observed (u, v) pairs of $\mathbf{X}^{(d)}$. The goal is to predict users' unobserved preferences based on a few historically observed preferences. For rating-based recommender systems, preferences are represented by numerical values (e.g., $[1, 2, ..., 5]$, or one star through five stars). In cross-system CF, besides $\mathcal{D}$, suppose we have a source CF task $\mathcal{S}$ which is associated with a relatively dense preference matrix $\mathbf{X}^{(s)} \in \mathbb{R}^{m_s \times n_s}$, where m_s is

the number of users and n_s is the number of items. Similarly, let $\mathcal{I}^{(s)}$ be the set of all observed (u, v) pairs of $\mathbf{X}^{(s)}$. Furthermore, we assume that cross-system entity correspondences are unknown, but can be identified with cost. Given a budget in terms of the maximum number of cross-system entity correspondences to be constructed, our goal is to 1) actively construct entity correspondences across the source and the target systems, and 2) make use of them for knowledge transfer from the source task $\mathcal{S}$ to the target task $\mathcal{D}$. In the sequel, we denote by $\mathbf{X}_{*,i}$ the i^{th} column of the matrix $\mathbf{X}$, and superscript $^\top$ the transpose of a vector or matrix, and use the words "domain" and "system" interchangeably.

3.1. Matrix Factorization for Collaborative Filtering

Matrix factorization [38, 39, 19] is one family of state-of-the-art algorithms in CF [40]. In matrix factorization for CF, given a sparse matrix $\mathbf{X}$, one can model the users and items using low-rank factor matrices $\mathbf{U} \in \mathbb{R}^{k\times m}$ and $\mathbf{V} \in \mathbb{R}^{k\times n}$, respectively, where the u^{th} user and the v^{th} item are represented by $\mathbf{U}_{*u}$ and $\mathbf{V}_{*v}$, i.e., the u^{th} and v^{th} column of $\mathbf{U}$ and $\mathbf{V}$, respectively. The objective of most matrix factorization methods for CF can be summarized in a general minimization problem as follows,

$$\min_{\mathbf{U},\mathbf{V},\Theta} \ell\left(\mathbf{U}, \mathbf{V}, \mathbf{X}; \Theta\right) + \lambda \mathcal{R}\left(\mathbf{U}, \mathbf{V}\right), \tag{1}$$

where $\ell(\cdot)$ is a loss function of factorization on the target rating matrix $\mathbf{X}$, and Θ is a set of parameters. The second term in the objective is a regularization term on the low-rank factor matrices of users and items, and $\lambda \geq 0$ is a trade-off parameter.

Different forms of the loss function $\ell(\cdot)$ lead to different approaches. Some popular loss functions include the Hinge loss with the form $h(z) = (1 - z)_+ = \max(0, 1-z)$, the negative-log-posterior loss $\ell = -\ln p(\mathbf{U}, \mathbf{V}|\mathbf{X})$ or equivalently the sum-of-squared-errors loss $\ell = \sum_{u,v}(x_{uv} - \mathbf{U}_{*u}^\top \mathbf{V}_{*v})^2$. In the following sections, we review three popular matrix factorization methods namely Maximum-Margin Matrix Factorization (MMMF) [41], Regularized Low-rank Matrix Factorization (RLMF) [19], and Probabilistic Matrix Factorization (PMF) [42], which will be extended and plugged in our proposed framework for cross-system CF.

3.1.1. Maximum-Margin Matrix Factorization

MMMF [41] aims to learn a fully observed matrix $\mathbf{Y} \in \mathbb{R}^{m\times n}$ to approximate the target preference matrix $\mathbf{X} \in \mathbb{R}^{m\times n}$ by maximizing the predictive margin and minimizing the trace norm of $\mathbf{Y}$. Specially, in binary preference predictions, $x_{uv} \in \{-1, +1\}$, where $x_{uv} = +1$ denotes that the user u likes the item v, while

$x_{uv} = -1$ denotes dislike. By considering hard-margin matrix factorization, the goal of MMMF is to find a minimum trace norm matrix $\mathbf{Y}$ that matches the observed preferences with a margin of one, i.e., $y_{uv}x_{uv} \geq 1$ for all $(u, v) \in \mathcal{I}$, where $\mathcal{I}$ is the set of observed (u, v) pairs of $\mathbf{X}$. By introducing slack variables $\xi_{uv} \geq 0$, the hard-margin constraint can be relaxed by requiring $y_{uv}x_{uv} \geq 1 - \xi_{uv}$ for all $(u, v) \in \mathcal{I}$, and minimizing a trade-off between the trace norm and the slack. This results in the following objective,

$$\min_{\mathbf{Y}} \sum_{(u,v)\in\mathcal{I}} h\left(y_{uv}x_{uv}\right) + \lambda \left\|\mathbf{Y}\right\|_{\Sigma}, \tag{2}$$

where $h(z) = (1 - z)_+ = \max(0, 1 - z)$ is the Hinge loss, $\|\cdot\|_{\Sigma}$ denotes the trace norm, and $\lambda \geq 0$ is a trade-off parameter. As proposed in [38], the objective (2) can be extended to ordinal rating predictions, and solved efficiently. To be specific, suppose $x_{uv} \in \{1, 2, ..., R\}$, one can use $R - 1$ thresholds $\theta_1, \theta_2, ..., \theta_{R-1}$ to relate the predicted real-valued y_{uv} to the discrete-valued x_{uv} by requiring

$$\theta_{x_{uv}-1} + 1 \leq y_{uv} \leq \theta_{x_{uv}} - 1,$$

where $\theta_0 = -\infty$ and $\theta_R = \infty$. When adding slack in this case, not only the violation of the two immediate constraints $\theta_{x_{uv}-1} + 1 \leq y_{uv}$ and $y_{uv} \leq \theta_{x_{uv}} - 1$, but also the violation of all other implied threshold constraints $\theta_r + 1 \leq y_{uv}$ for $r < x_{uv}$ and $y_{uv} \leq \theta_r - 1$ for $r \geq x_{uv}$ should be penalized. The goal by doing this is to emphasize the cost of crossing multiple rating-boundaries and yield a loss function that upper bounds the mean-absolute-error (MAE). By further supposing that $\mathbf{Y}$ can be decomposed as $\mathbf{Y} = \mathbf{U}^\top\mathbf{V}$, where $\mathbf{U} \in \mathbb{R}^{k\times m}$ and $\mathbf{V} \in \mathbb{R}^{k\times n}$. The objective of MMMF for ordinal rating predictions can be written as follows,

$$\min_{\mathbf{U},\mathbf{V},\Theta} \sum_{(u,v)\in\mathcal{I}} \sum_{r=1}^{R-1} h\left(T_{uv}^r\left(\theta_{ur} - \mathbf{U}_{*u}^\top\mathbf{V}_{*v}\right)\right) + \lambda\left(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2\right), \tag{3}$$

where $T_{uv}^r = +1$ for $r \geq x_{uv}$, while $T_{uv}^r = -1$ for $r < x_{uv}$, and $\|\cdot\|_F$ denotes the Frobenius norm. The thresholds $\Theta = \{\theta_{ur}\}$'s can be learned together with $\mathbf{U}$ and $\mathbf{V}$ from the data. Note that the thresholds here are user-specific, i.e., for a same user u, the values of the corresponding thresholds θ_{ur}'s are the same, while for different users u's, the values of the corresponding thresholds θ_{ur}'s can be different. The alternating minimization approach can be applied to solve the optimization problem [38, 19]: iteratively keep two of $\mathbf{U}$, $\mathbf{V}$ and Θ fixed and optimize over the other using gradient-descent approaches, then switch and repeat.

3.1.2. Regularized Low-rank Matrix Factorization

RLMF [19] is a matrix factorization approach based on regularized Singular Value Decomposition (SVD) on sparse matrices. The objective of RLMF is to solve the following minimization problem,

$$\min_{\mathbf{U},\mathbf{V},b_u,b_v} \sum_{(u,v)\in\mathcal{I}} \left(x_{uv} - \left(\bar{r} + b_u + b_v + \mathbf{U}_{*u}^\top \mathbf{V}_{*v}\right)\right)^2 + \lambda\left(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2 + b_u^2 + b_v^2\right), \quad (4)$$

where $\bar{r}$ is the observed overall averaged rating, b_u and b_v indicate the bias of user u and item v, respectively. The second term of the objective consists of a set of regularization terms on $\mathbf{U}$, $\mathbf{V}$, b_u and b_v, respectively, and λ is a trade-off parameter to control the impact of the regularization terms. A local minimum of the objective (4) can be obtained by performing gradient descent on the objective with respect to $\mathbf{U}$, $\mathbf{V}$, b_u and b_v, alternatingly.

3.1.3. Probabilistic Matrix Factorization

PMF [42] adopts a probabilistic model with Gaussian observation noise, and aims to maximize the following conditional distribution over the observed ratings,

$$p(\mathbf{X}|\mathbf{U},\mathbf{V},\sigma^2) = \prod_{u=1}^{m}\prod_{v=1}^{n} \left[\mathcal{N}(x_{uv}|\mathbf{U}_{*u}^\top \mathbf{V}_{*v}, \sigma^2)\right]^{I_{uv}}, \quad (5)$$

where $\mathcal{N}(x_{uv}|\mathbf{U}_{*u}^\top \mathbf{V}_{*v}, \sigma^2)$ is a probability density function of a Gaussian distribution with the mean $\mu = \mathbf{U}_{*u}^\top \mathbf{V}_{*v}$ and the variance σ^2, and I_{uv} is the indicator function that is equal to 1 if the user u rates the item v, i.e., $(u,v) \in \mathcal{I}$, and equal to 0 otherwise. To bound predictions within the range of valid rating values, the logistic function $g(x) = 1/(1+exp(-x))$ is post-performed on the dot product between the user- and item-specific latent vectors, and the function $t(x) = (x-1)/(R-1)$ is used to map the ratings $\{1, 2, ..., R\}$ to the interval $[0, 1]$ so that the range of valid rating values matches the range of predictions. This results in the following conditional distribution:

$$p(\mathbf{X}|\mathbf{U},\mathbf{V},\sigma^2) = \prod_{u=1}^{m}\prod_{v=1}^{n} \left[\mathcal{N}(x_{uv}|g(\mathbf{U}_{*u}^\top \mathbf{V}_{*v}), \sigma^2)\right]^{I_{uv}}. \quad (6)$$

Moreover, usually, zero-mean spherical Gaussian priors are placed on the user and item latent vectors, respectively,

$$\begin{cases} p(\mathbf{U}|\sigma_U^2) = \prod_{u=1}^{m} \mathcal{N}(\mathbf{U}_{*u}|0, \sigma_U^2\mathbf{I}), \\ p(\mathbf{V}|\sigma_V^2) = \prod_{v=1}^{n} \mathcal{N}(\mathbf{V}_{*v}|0, \sigma_V^2\mathbf{I}). \end{cases} \quad (7)$$

In this paper, we adopt the constraint version of PMF proposed in [42], which introduces a new latent similarity constraint matrix $\mathbf{H} \in \mathbb{R}^{k \times n}$ on users. As a result, the latent vector of the user u can be represented by

$$\mathbf{U}_{*u} = \mathbf{Y}_{*u} + \frac{\sum_{h=1}^{n} I_{uh} \mathbf{H}_{*h}}{\sum_{h=1}^{n} I_{uh}}, \tag{8}$$

where I_{uh} is the indicator function that is equal to 1 if the user u rates the item h, and equal to 0 otherwise, $\mathbf{H}_{*h}$ captures the effect of a user having rated a particular item v on the prior mean of the user's latent vector, and $\mathbf{Y}_{*u}$ can be seen as the offset added to the mean of the prior distribution to get the latent vector $\mathbf{U}_{*u}$ for the user u. By plugging (8) into (6), we obtain a new conditional distribution over the observed ratings as follows,

$$p(\mathbf{X}|\mathbf{Y}, \mathbf{V}, \mathbf{H}, \sigma^2) = \prod_{u=1}^{m} \prod_{v=1}^{n} \left[\mathcal{N}\left(x_{uv} | g\left(\left(\mathbf{Y}_{*u} + \frac{\sum_{h=1}^{n} I_{uh} \mathbf{H}_{*h}}{\sum_{h=1}^{n} I_{uh}} \right)^{\top} \mathbf{V}_{*v} \right), \sigma^2 \right) \right]^{I_{uv}}, \tag{9}$$

where the Gaussian prior on $\mathbf{U}$ in (7) is replaced by the one on $\mathbf{Y}$, and an additional zero-mean spherical Gaussian prior is placed on the latent similarity constraint matrix $\mathbf{H}$:

$$p(\mathbf{H}|\sigma_H^2) = \prod_{w=1}^{n} \mathcal{N}(\mathbf{H}_{*h}|0, \sigma_H^2 \mathbf{I}). \tag{10}$$

Based on the conditional distribution in (9), it can be proven that maximizing the log-posterior $\ln p(\mathbf{Y}, \mathbf{V}, \mathbf{H}|\mathbf{X}, \sigma^2, \sigma_Y^2, \sigma_V^2, \sigma_H^2)$ over the user and item factor matrixes with the priors is equivalent to minimizing the sum-of-squared-errors objective function with quadratic regularization terms formulated as follows,

$$\begin{aligned} \min_{\mathbf{Y}, \mathbf{V}, \mathbf{H}} \sum_{(u,v) \in \mathcal{I}} & \left(x_{uv} - g\left(\left(\mathbf{Y}_{*u} + \frac{\sum_{h=1}^{n} I_{uh} \mathbf{H}_{*h}}{\sum_{h=1}^{n} I_{uh}} \right)^{\top} \mathbf{V}_{*v} \right) \right)^2 \\ & + \lambda_Y \|\mathbf{Y}\|_F^2 + \lambda_V \|\mathbf{V}\|_F^2 + \lambda_H \|\mathbf{H}\|_F^2, \end{aligned} \tag{11}$$

where $\lambda_Y = \sigma^2/\sigma_Y^2$, $\lambda_V = \sigma^2/\sigma_V^2$ and $\lambda_H = \sigma^2/\sigma_H^2$ are trade-off parameters to balance the effect among the approximation error term and the regularization terms. A local minimum of the objective function (11) can be found by performing gradient decent on the objective respect to $\mathbf{Y}$, $\mathbf{V}$ and $\mathbf{H}$, alternatingly.

4. Active Transfer Learning for Cross-System Collaborative Filtering

4.1. A Unified Framework

The overall general framework on active transfer learning for cross-system CF is described in Algorithm 1. To begin with, we apply a base matrix factorization method f, which will be specified in particular solutions, on the target collaborative data to learn a CF model for initialization. After that, we iteratively perform the following three steps:

- We choose K entities based on an entity selection function $ActiveLearn()$, which will be specified in particular solutions as well.
- We query their correspondences in the source system.
- We then apply the extended matrix factorization method f_{TL} in the transfer learning manner on both the source and target collaborative data to learn an updated CF model. Note that the entity selection function $ActiveLearn()$ is built on top of the base method f at the initial step or the extended method f_{TL} at each iteration.

In the rest of this section, we first describe the idea of the extended matrix factorization method in the transfer learning manner f_{TL} by assuming a set of cross-system correspondences be constructed as input, and then present the high-level idea on how to design $ActiveLearn()$ to actively select entities for querying cross-system correspondences.

4.1.1. High-level Idea on f_{TL}

Denote by $\mathbf{U}_{\mathcal{C}}^{(s)}$ and $\mathbf{V}_{\mathcal{C}}^{(s)}$ the factor sub-matrices of $\mathbf{U}^{(s)}$ and $\mathbf{V}^{(s)}$ for the entities in the source system, whose indices are in $\mathcal{C}$, respectively. Similarly, denote by $\mathbf{U}_{\mathcal{C}}^{(d)}$ and $\mathbf{V}_{\mathcal{C}}^{(d)}$ the factor sub-matrices for the entities in the target system, whose indices are in $\mathcal{C}$, respectively. Here $\mathcal{C}$ denotes the unified indices of the constructed corresponding entities, which can be either users or items, between the source and target systems. The general objective of the extended matrix factorization method f_{TL} with partial entity correspondences for cross-system CF can be written as follows,

$$\min_{\mathbf{U}^{(d)},\mathbf{V}^{(d)},\Theta} \ell\big(\mathbf{U}^{(d)},\mathbf{V}^{(d)},\mathbf{X}^{(d)};\Theta\big)+\lambda\mathcal{R}\big(\mathbf{U}^{(d)},\mathbf{V}^{(d)}\big)+\lambda_{\mathcal{C}}\mathcal{R}\Big(\mathbf{U}_{\mathcal{C}}^{(d)},\mathbf{V}_{\mathcal{C}}^{(d)},\mathbf{U}_{\mathcal{C}}^{(s)},\mathbf{V}_{\mathcal{C}}^{(s)}\Big), \quad (12)$$

where the last term is a regularization term that aims to use $\mathbf{U}_{\mathcal{C}}^{(s)}$ and $\mathbf{V}_{\mathcal{C}}^{(s)}$ as priors to learn more precise $\mathbf{U}_{\mathcal{C}}^{(d)}$ and $\mathbf{V}_{\mathcal{C}}^{(d)}$, which can be expanded to obtain more

Algorithm 1 Active Transfer Learning for Cross-System CF

Input: The factor matrices of users and items, $\mathbf{U}^{(s)}$ and $\mathbf{V}^{(s)}$, learned by the source-system CF model, the sparse target-system rating matrix $\mathbf{X}^{(d)}$, a base matrix factorization method f, and its extension f_{TL} in the transfer learning manner, total number of iterations T, and the number of cross-system entity correspondences to actively constructed at each iteration K,
Output: Factor matrices of users and items in the target system, $\mathbf{U}^{(d)}$ and $\mathbf{V}^{(d)}$.

Initialize:
Apply f on $\mathbf{X}^{(d)}$ to generate $\Theta_0^{(d)}$, $\mathbf{U}_0^{(d)}$ and $\mathbf{V}_0^{(d)}$.
for $t = 0$ to T **do**
Step 1: Set $\mathcal{C}^{(d)} = ActiveLearn(\Theta_t^{(d)}, \mathbf{U}_t^{(d)}, \mathbf{V}_t^{(d)}, K)$, where $\mathcal{C}^{(d)}$ is the set of the indices of the selected entities (either users or items), and $|\mathcal{C}^{(d)}| = K$.
Step 2: Query the selected target-system entities in the source system to identify their corresponding set $\mathcal{C}^{(s)}$. For simplicity, we use $\mathcal{C}$ to denote the unified indices of the constructed corresponding entities between domains.
Step 3: Perform $f_{TL}(\mathbf{U}^{(s)}, \mathbf{V}^{(s)}, \mathbf{U}_t^{(d)}, \mathbf{V}_t^{(d)}, \mathbf{X}^{(d)}, \mathcal{C})$ to update $\Theta_{t+1}^{(d)}$, $\mathbf{U}_{t+1}^{(d)}$, and $\mathbf{V}_{t+1}^{(d)}$.
end for
Return: $\mathbf{U}^{(d)} \leftarrow \mathbf{U}_T^{(d)}$, and $\mathbf{V}^{(d)} \leftarrow \mathbf{V}_T^{(d)}$.

precise $\mathbf{U}^{(d)}$ and $\mathbf{V}^{(d)}$, respectively. The associated $\lambda_{\mathcal{C}} \geq 0$ is a trade-off parameter to control the impact of the regularization term.

Intuitively, a simple way to define the regularization term is to enforce the target factor sub-matrices $\mathbf{U}_{\mathcal{C}}^{(d)}$ and $\mathbf{V}_{\mathcal{C}}^{(d)}$ in target system to be the same as the source factor sub-matrices $\mathbf{U}_{\mathcal{C}}^{(s)}$ and $\mathbf{V}_{\mathcal{C}}^{(s)}$, respectively, i.e.,

$$\mathcal{R}\left(\mathbf{U}_{\mathcal{C}}^{(d)}, \mathbf{V}_{\mathcal{C}}^{(d)}, \mathbf{U}_{\mathcal{C}}^{(s)}, \mathbf{V}_{\mathcal{C}}^{(s)}\right) = \left\|\mathbf{W}_{\mathcal{C}}^{(s)} - \mathbf{W}_{\mathcal{C}}^{(d)}\right\|_F^2, \tag{13}$$

where $\mathbf{W}_{\mathcal{C}}^{(d)} = [\mathbf{U}_{\mathcal{C}}^{(d)} \ \mathbf{V}_{\mathcal{C}}^{(d)}]$, $\mathbf{W}_{\mathcal{C}}^{(s)} = [\mathbf{U}_{\mathcal{C}}^{(s)} \ \mathbf{V}_{\mathcal{C}}^{(s)}]$ with $\mathbf{U}_{\mathcal{C}}^{(d)}, \mathbf{U}_{\mathcal{C}}^{(s)} \in \mathbb{R}^{k \times n_1}$, $\mathbf{V}_{\mathcal{C}}^{(d)}, \mathbf{V}_{\mathcal{C}}^{(s)} \in \mathbb{R}^{k \times n_2}$, and $n_1 + n_2 = |\mathcal{C}|$. The regularization term defined in (13) is based on an "identical" assumption on the factor sub-matrices $\mathbf{U}_{\mathcal{C}}$ and $\mathbf{V}_{\mathcal{C}}$: the source and the target systems should share the same factor sub-matrices $\mathbf{U}_{\mathcal{C}}$ and $\mathbf{V}_{\mathcal{C}}$, i.e., $\mathbf{U}_{\mathcal{C}}^{(s)} = \mathbf{U}_{\mathcal{C}}^{(d)} = \mathbf{U}_{\mathcal{C}}$ and $\mathbf{V}_{\mathcal{C}}^{(s)} = \mathbf{V}_{\mathcal{C}}^{(d)} = \mathbf{V}_{\mathcal{C}}$. This assumption is similar to that used in CMF, and may be too restricted to satisfy in practice.

Alternatively, we propose to use the similarities between entities estimated in

the source system as priors to constrain the similarities between the corresponding entities in the target system. The motivation is that if two entities in the source system are similar to each other, then their correspondences tend to be similar to each other in the target system as well. Therefore, we propose to use the following regularization term on the factor sub-matrices,

$$\mathcal{R}\left(\mathbf{U}_{\mathcal{C}}^{(d)}, \mathbf{V}_{\mathcal{C}}^{(d)}, \mathbf{U}_{\mathcal{C}}^{(s)}, \mathbf{V}_{\mathcal{C}}^{(s)}\right) = \text{tr}\left(\mathbf{W}_{\mathcal{C}}^{(d)} \mathbf{L}_{\mathcal{C}}^{(s)} {\mathbf{W}_{\mathcal{C}}^{(d)}}^{\top}\right), \tag{14}$$

where $\text{tr}(\cdot)$ denotes the trace of a matrix, and $\mathbf{L}_{\mathcal{C}}^{(s)} = \begin{bmatrix} \mathbf{L}_U^{(s)} & 0 \\ 0 & \mathbf{L}_V^{(s)} \end{bmatrix}$, with $\mathbf{L}_U^{(s)} = \mathbf{D}_U^{(s)} - \mathbf{A}_U^{(s)}$ and $\mathbf{L}_V^{(s)} = \mathbf{D}_V^{(s)} - \mathbf{A}_V^{(s)}$, where $\mathbf{A}_U^{(s)} = {\mathbf{U}_{\mathcal{C}}^{(s)}}^{\top} \mathbf{U}_{\mathcal{C}}^{(s)}$ and $\mathbf{A}_V^{(s)} = {\mathbf{V}_{\mathcal{C}}^{(s)}}^{\top} \mathbf{V}_{\mathcal{C}}^{(s)}$ are the similarity matrices of the corresponding users and items in the source system, respectively. The diagonal matrices $\mathbf{D}_U^{(s)}$ and $\mathbf{D}_V^{(s)}$ are defined as $[\mathbf{D}_U^{(s)}]_{ii} = \sum_j [\mathbf{A}_U^{(s)}]_{ij}$ and $[\mathbf{D}_V^{(s)}]_{ii} = \sum_j [\mathbf{A}_V^{(s)}]_{ij}$, respectively. The matrices $\mathbf{W}_{\mathcal{C}}^{(s)}$ and $\mathbf{W}_{\mathcal{C}}^{(d)}$ are as the same as defined in (13), and the matrices $\mathbf{L}_U^{(s)}$ and $\mathbf{L}_V^{(s)}$ are known as the Laplacian matrices [43]. Note that a similar regularization term has been proposed in [44]. However, their work is focused on utilizing relational information for single-domain CF, and the Laplacian matrix is constructed using links between entities in a single domain.

4.1.2. High-level Idea on $ActiveLearn()$

Based on the extended matrix factorization method introduced above, intuitively, at each iteration, we should select entities (either items or users) in the target system, whose predictions by the current CF model are of *most uncertainty*, to query their correspondences in the source system. In this way, knowledge transferred from the source system can improve the prediction accuracy on the most uncertain target-system entities, and thus improve the overall prediction accuracy for the target system. Similar ideas have been widely used in many active learning approaches to various applications [45]. However, in the context of recommender systems, users' ratings on items typically follow a power-law distribution, which is also known as the long tail problem. Specifically, regarding items, the long tail is composed of a small number of popular items with lots of users' ratings, and the rest are located in the heavy tail, which are not sold well and only have few users' ratings. Similarly, regarding users, the long tail is composed of a small number of active users who give a lots of ratings on items, and the rest are located in the heavy tail, who are inactive to give items ratings. It has been shown that matrix-factorization-based CF methods usually fail to make confident predictions on the

items (or users) for a specific user (or item), whose historical ratings are rare. Therefore, the items (or users) of the most uncertain predictions in the target recommender system tend to be in the tail with high probabilities. Furthermore, since we assume the source and the target recommender systems be similar, if the items (or users) are in the tail in the target system, then their correspondences tend to be in the tail in the source system as well. This implies that the predictions on the corresponding entities in the source system may not be precise either, resulting in limited knowledge transferred through the extended matrix factorization method. Therefore, besides focusing on prediction uncertainty, we need to take the long tail issue into consideration when designing an active entity selection strategy for building cross-system correspondences.

At a high level, we propose an active entity selection strategy for the target domain as follow. For simplicity in description, in this section and the subsequent sections, we only describe how to actively select users from the target system for querying corresponding users in the source system. Procedure on actively selecting items for correspondences construction is similar.

- We first denote by $\delta^{(d)}(u, v)$ an *certainty* measure of the prediction of a matrix-factorization-based CF model on a user-item pair (u, v) in the target system. The larger is the value, the more certain or confident is the prediction.

- With $\delta^{(d)}(u, v)$, we then define an entity-level certainty measure on a user, $\delta_u^{(d)}$, as follows,

$$\delta_u^{(d)} = \eta \frac{1}{\left|\mathcal{I}_u^{(d)}\right|} \sum_{v \in \mathcal{I}_u^{(d)}} \delta^{(d)}(u, v) + (1 - \eta) \frac{1}{\left|\widehat{\mathcal{I}}_u^{(d)}\right|} \sum_{v \in \widehat{\mathcal{I}}_u^{(d)}} \delta^{(d)}(u, v), \qquad (15)$$

  where $\mathcal{I}_u^{(d)}$ denotes the item set of observed ratings given by user u in the target system, while $\widehat{\mathcal{I}}_u^{(d)}$ denotes the item set of unobserved ratings for user u in the target system. On the right hand side of the equation, the first term is the average of the certainty of predictions on the user-item pairs for user u, whose rating are observed, and the second term is the average of the certainty of predictions on the user-item pairs for user u, whose ratings are unobserved. The tradeoff parameter $\eta \in [0, 1]$ is to balance the impact of the two terms to the overall certainty of the predictions on user u. In this paper, we simply set $\eta = 0.5$.

- At each round or iteration, in order to select K source-system users, we first select K_1 ($< K$) users who are of the least certainty (i.e., the most uncertainty) based on $\delta_u^{(d)}$ to construct $\mathcal{C}_2$, i.e., selecting K_1 users whose corresponding $\delta_u^{(d)}$'s are of smallest values. After that for the rest users $\{u_i\}$'s, we select $K-K_1$ users with largest scores according to the following function to construct $\mathcal{C}_2$,

$$\Delta^{(d)}(u_i, \mathcal{C}_1) = \frac{\sum_{u_j \in \mathcal{C}_1} \text{sim}(u_i, u_j)\delta_{u_i}^{(d)}}{\sum_{u_j \in \mathcal{C}_1} \text{sim}(u_i, u_j)}, \tag{16}$$

where $\text{sim}(u_i, u_j) = \frac{\left|\mathcal{I}_{u_i}^{(d)} \cap \mathcal{I}_{u_j}^{(d)}\right|}{\max\left(\left|\mathcal{I}_{u_i}^{(d)}\right|, \left|\mathcal{I}_{u_j}^{(d)}\right|\right)}$ is the correlation measure between the users u_i and u_j based on their rating behaviors. Finally, we set $\mathcal{C} = \mathcal{C}_1 \cap \mathcal{C}_2$, which is the set of K users to be selected. The motivation behind the scoring function (16) is that we aim to select users for constructing $\mathcal{C}_2$, who are 1) informative, i.e., with large values of $\{\delta_{u_i}^{(d)}\}$'s, and thus supposed to be "active" instead of being in the tail, and 2) of strong correlation to the pre-selected most uncertain users in $\mathcal{C}_1$, i.e., with large values of $\sum_{u_j \in \mathcal{C}_1} \text{sim}(u_i, u_j)$, and thus supposed to be helpful for generating reasonable recommendations based on the intrinsic assumption in CF.

In the following sections, we introduce three particular solutions by equipping different base matrix factorization methods and their transfer learning extensions in the framework, and present their specific active correspondences construction approaches in detail.

4.2. A Solution Equipped with Maximum-Margin Matrix Factorization

In this section, we first present a particular solution based on MMMF. We start by introducing an extended MMMF method in the transfer learning manner with partial entity correspondences between the source and the target systems, and then present an approach to actively selecting entities to construct correspondences between systems.

4.2.1. MMMF with Partial Entity Correspondence

By plugging the objective of MMMF (3) and the regularization term of cross-system entity correspondences (14) into the framework (12), we obtain the optimization problem of the extended MMMF method in the transfer learning manner

with partial entity correspondences between systems as follows,

$$\min_{\mathbf{U}^{(d)},\mathbf{V}^{(d)},\Theta} \sum_{(u,v)\in\mathcal{I}^{(d)}} \sum_{r=1}^{R-1} h\left(T_{uv}^{r}\left(\theta_{ur} - {\mathbf{U}^{(d)}}_{*u}^{\top}\mathbf{V}_{*v}^{(d)}\right)\right) + \lambda\left(\|\mathbf{U}^{(d)}\|_F^2 + \|\mathbf{V}^{(d)}\|_F^2\right) + \lambda_{\mathcal{C}}\mathrm{tr}\left(\mathbf{W}_{\mathcal{C}}^{(d)}\mathbf{L}_{\mathcal{C}}^{(s)}{\mathbf{W}_{\mathcal{C}}^{(d)}}^{\top}\right), \tag{17}$$

where $\mathbf{W}_{\mathcal{C}}^{(d)} = [\mathbf{U}_{\mathcal{C}}^{(d)}\ \mathbf{V}_{\mathcal{C}}^{(d)}]$, and $\mathbf{W}_{\mathcal{C}}^{(s)} = [\mathbf{U}_{\mathcal{C}}^{(s)}\ \mathbf{V}_{\mathcal{C}}^{(s)}]$. In the sequel, we denote by MMMF$_{TL}$ the optimization problem (17). For comparison on the impact of different regularization terms to transfer learning, we denote by MMMF$_{CMF}$ the optimization problem by replacing the last regularization term in (17) by the CMF regularization term (13) as follows,

$$\min_{\mathbf{U}^{(d)},\mathbf{V}^{(d)},\Theta} \sum_{(u,v)\in\mathcal{I}^{(d)}} \sum_{r=1}^{R-1} h\left(T_{uv}^{r}\left(\theta_{ur} - {\mathbf{U}^{(d)}}_{*u}^{\top}\mathbf{V}_{*v}^{(d)}\right)\right) + \lambda\left(\|\mathbf{U}^{(d)}\|_F^2 + \|\mathbf{V}^{(d)}\|_F^2\right) + \lambda_{\mathcal{C}}\left\|\mathbf{W}_{\mathcal{C}}^{(s)} - \mathbf{W}_{\mathcal{C}}^{(d)}\right\|_F^2. \tag{18}$$

4.2.2. Actively Constructing Entity Correspondences through MMMF

As MMMF is a margin-based matrix factorization method, in this section, we present a margin-based approach for actively constructing cross-system entity correspondences. To implement the active entity selection strategy introduced in Section 4.1.2, we need to specify the certainty measure of the prediction on a user-item pair, $\delta^{(d)}(u,v)$, and a user, $\delta_u^{(d)}$, based on the extended MMMF method. A common motivation behind most margin-based active learning approaches [46, 47, 48] is that given a margin-based model, the margin of an example denotes certainty of the prediction on the example. The larger the margin is for an example, the higher the certainty is for its prediction. In the following, we start by defining a margin on a user-item pair.

Margins on User-Item Pairs. Suppose that MMMF (3) or MMMF$_{TL}$ (17) has been applied to the collaborative data in the target system to learn the factor matrices $\mathbf{U}^{(d)}$ and $\mathbf{V}^{(d)}$. The margins of a prediction with respect to the thresholds θ_0, θ_1, ..., θ_R ($\theta_0 = -\infty$ and $\theta_R = +\infty$) in MMMF are illustrated in Figure 2. Intuitively, given a user-item pair (u,v) in the target domain, we expect the predicted rating by MMMF, ${\mathbf{U}^{(d)}}_{*,u}^{\top}\mathbf{V}_{*,v}^{(d)}$, to be in the correct interval $(\theta_{x_{uv}-1}, \theta_{x_{uv}}]$, and to be far from the boundaries (thresholds). Therefore, the margin of a user-item pair

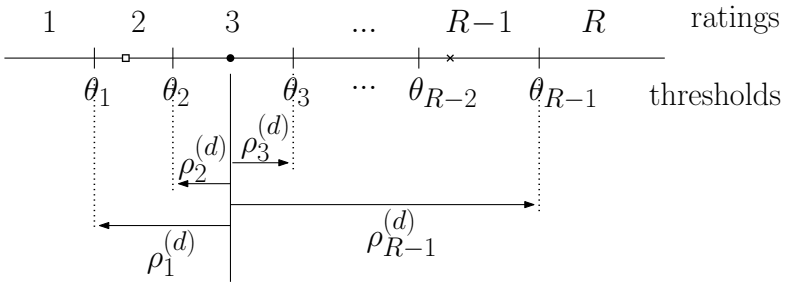


Figure 2: The margins of a user-item pair.

(u, v) can be defined as

$$\begin{cases} \rho_r^{(d)}(u,v) = {\mathbf{U}^{(d)}}^{\top}_{*,u}\mathbf{V}^{(d)}_{*,v} - \theta_r, \text{ if } x^{(d)}_{u,v} > r, \\ \\ \rho_r^{(d)}(u,v) = \theta_r - {\mathbf{U}^{(d)}}^{\top}_{*,u}\mathbf{V}^{(d)}_{*,v}, \text{ if } x^{(d)}_{u,v} \leq r. \end{cases} \tag{19}$$

Similar to other margin-based active learning methods, here we assume that, for a user-item pair (u, v) whose rating is not observed, the prediction of the current CF model is correct, i.e., $x^{(d)}_{u,v} = {\mathbf{U}^{(d)}}^{\top}_{*,u}\mathbf{V}^{(d)}_{*,v}$. Based on the above definition, for each user-item pair (u, v), there are $R-1$ margins. For instance, as shown in Figure 2, for the circle point that denotes the predicted rating of a user-item pair (u, v), the associated $R-1$ margins denoted by $\rho_1(u, v)$, $\rho_2(u, v)$, ..., and $\rho_{R-1}(u, v)$ are the distances between the point and the $R-1$ thresholds θ_1, θ_2, ..., and θ_{R-1}, respectively. Among the $R-1$ margins, the margins to the left (lower) and right (upper) boundaries of the correct interval are of the most importance, which are denoted by $\rho_L^{(d)}(u, v)$ and $\rho_U^{(d)}(u, v)$, respectively. For the circle point shown in Figure 2, $\rho_L^{(d)}(u, v) = \rho_2^{(d)}(u, v)$ and $\rho_U^{(d)}(u, v) = \rho_3^{(d)}(u, v)$. Intuitively, for a user-item pair (u, v), when the predicted rating is in the middle of the correct interval, i.e., $\rho_L^{(d)}(u, v) = \rho_U^{(d)}(u, v)$, the confidence of the prediction is the highest, because it is farthest from the two boundaries of the correct interval. Therefore, we define $\delta^{(d)}(u, v)$ as the normalized margin of a user-item pair (u, v) as follows,

$$\delta^{(d)}(u,v) = 1 - \frac{\left|\rho_L^{(d)}(u,v) - \rho_U^{(d)}(u,v)\right|}{\rho_L^{(d)}(u,v) + \rho_U^{(d)}(u,v)}. \tag{20}$$

Note that $\delta^{(d)}(u, v) \in [b, 1]$, where $b = \min\left(1 - \frac{\rho_L^{(d)}}{\rho_L^{(d)}+\rho_U^{(d)}}, 1 - \frac{\rho_U^{(d)}}{\rho_L^{(d)}+\rho_U^{(d)}}\right)$. When $\rho_L^{(d)}(u, v) = \rho_U^{(d)}(u, v)$, the margin obtains its maximum, i.e., $\delta^{(d)}(u, v) = 1$, and

when $\rho_L^{(d)}(u,v) = 0$ or $\rho_U^{(d)}(u,v) = 0$, the margin obtains its minimum, i.e., $\delta^{(d)}(u,v) = b$.

Margins on Entities. With the normalized margin or certainty measure on each user-item pair defined in (20), we are able to define the overall margin or certainty measure on a user, $\delta_u^{(d)}$, using (15). With the margin-based $\delta_u^{(d)}$, we can implement the active entity selection strategy introduced in Section 4.1.2 with the extended MMMF method. In the sequel, we denote by MG_{hy} this active entity selection approach.

4.3. A Solution Equipped with Regularized Low-rank Matrix Factorization

In this section, we present a second particular solution with RLMF. We start by introducing an extended RLMF method in the transfer learning manner with flexible entity correspondences between the source and the target systems, and then present an active entity selection approach to constructing cross-system correspondences based on the extended RLMF.

4.3.1. RLMF with Partial Entity Correspondence

By plugging the objective of RLMF (4) and the regularization term of cross-system entity correspondences (14) into the framework (12), we obtain the optimization problem of the extended RLMF method in the transfer learning manner with partial entity correspondences between systems as follows,

$$\begin{aligned}\min_{\mathbf{U}^{(d)},\mathbf{V}^{(d)},b_u,b_v} & \sum_{(u,v)\in\mathcal{I}^{(d)}} \left(x_{uv}^{(d)} - \left(\bar{r} + b_u + b_v + {\mathbf{U}_{*u}^{(d)}}^\top \mathbf{V}_{*v}^{(d)}\right)\right)^2 + \lambda(b_u^2 + b_v^2) \\ & + \lambda\left(\|\mathbf{U}^{(d)}\|_F^2 + \|\mathbf{V}^{(d)}\|_F^2\right) + \lambda_{\mathcal{C}} \text{tr}\left(\mathbf{W}_{\mathcal{C}}^{(d)} \mathbf{L}_{\mathcal{C}}^{(s)} {\mathbf{W}_{\mathcal{C}}^{(d)}}^\top\right). \end{aligned} \tag{21}$$

In the sequel, we denote by RLMF_{TL} the optimization problem (21). For comparison on the impact of different regularization terms to transfer learning, we denote by RLMF_{CMF} the optimization problem by replacing the last regularization term in (21) with the CMF regularization term (13) as follows,

$$\begin{aligned}\min_{\mathbf{U}^{(d)},\mathbf{V}^{(d)},b_u,b_v} & \sum_{(u,v)\in\mathcal{I}^{(d)}} \left(x_{uv}^{(d)} - \left(\bar{r} + b_u + b_v + {\mathbf{U}_{*u}^{(d)}}^\top \mathbf{V}_{*v}^{(d)}\right)\right)^2 + \lambda(b_u^2 + b_v^2) \\ & + \lambda\left(\|\mathbf{U}^{(d)}\|_F^2 + \|\mathbf{V}^{(d)}\|_F^2\right) + \lambda_{\mathcal{C}} \left\|\mathbf{W}_{\mathcal{C}}^{(s)} - \mathbf{W}_{\mathcal{C}}^{(d)}\right\|_F^2. \end{aligned} \tag{22}$$

4.3.2. Actively Constructing Entity Correspondences through RLMF

Different from the margin-based approach to active correspondences construction, here, we present an error-based approach with the extended RLMF method for actively constructing entity correspondences. This approach does not aim to measure how much the model is likely to change, but how much its generalization error is likely to be reduced. The idea is to iteratively build new correspondences, with which the expected generalization error of the current CF model for the target system can be reduced to the utmost extent.

Expected Errors on User-Item Pairs. Suppose that RLMF (4) or RLMF$_{TL}$ (21) is fed with collaborative data in the target system to learn a CF model that predicts y_{uv} for each user-item pair. Given b_u, b_v, $\mathbf{U}$, $\mathbf{V}$, we can then write the expected error of the CF model as follows:

$$EE = e(\mathbf{X}, \mathbf{Y}), \tag{23}$$

where $e(\cdot)$ is some loss function that measures the degree of disagreement in difference between the true ratings $\mathbf{X}$ and the model's predictions $\mathbf{Y}$. The proposed error-based active learning approach thus aims to select a set of queries $\mathcal{Q}$ at each iteration to construct K more correspondences between the source and the target system in addition to the existing set of correspondences such that the resulting new CF model obtains lower generalization error than any other set of queries $\mathcal{Q}'$ of K correspondences construction, i.e.,

$$EE_{\mathcal{C}+\mathcal{Q}} < EE_{\mathcal{C}+\mathcal{Q}'}, \tag{24}$$

where $\mathcal{C}$ is the set of cross-system correspondences used in the current CF model, which can be empty. In this paper, we adopt the sum-of-squared-errors loss $e(\mathbf{X}, \mathbf{Y}) = \sum_{u,v}(x_{uv}^{(d)} - y_{uv}^{(d)})^2$. Therefore, the error on each user-item pair (u, v) can be calculated as:

$$e^{(d)}(u, v) = \left\| y_{uv}^{(d)} - {\mathbf{U}_{*,u}^{(d)}}^{\top} \mathbf{V}_{*,v}^{(d)} - b_u - b_v \right\|_2,$$

where $y_{uv}^{(d)} = x_{uv}^{(d)}$ if $x_{uv}^{(d)}$ is observed, otherwise,

$$y_{uv}^{(d)} = \arg\min_r \left| r - {\mathbf{U}_{*,u}^{(d)}}^{\top} \mathbf{V}_{*,v}^{(d)} - b_u - b_v \right|,$$

where $r \in \{1, ..., R\}$. In this way, the *uncertainty* of a prediction on an instance can be measured by the expected error of the predictive model on the instance,

the larger is the expected error, the more *uncertain* is the prediction. However, as defined in Section 4.1.2, $\delta^{(d)}(u, v)$ is a "certainty" measure, which is supposed to be larger when the corresponding prediction is more certain. Therefore, here, we define $\delta^{(d)}(u, v)$ as the *negative* of the expected error on a user-item pair (u, v):

$$\delta^{(d)}(u, v) = -e^{(d)}(u, v). \tag{25}$$

Expected Errors on Entities. With the expected-error-based $\delta^{(d)}(u, v)$ on a user-item pair defined in (25), we can define the overall uncertainty measure on a user, $\delta_u^{(d)}$, using (15), and implement the active entity selection strategy introduced in Section 4.1.2 with the extended RLMF method. In the sequel, we denote by EE_{hy} this active entity selection approach.

4.4. A Solution Equipped with Probabilistic Matrix Factorization

In this section, we present the third solution based on PMF. We start by introducing an extended PMF method in the transfer learning manner with partial entity correspondences between the source and the target systems, and then present an active entity correspondences construction approach accordingly.

4.4.1. PMF with Partial Entity Correspondence

By plugging the objective of PMF (11) and the regularization term of cross-system entity correspondences (14) into the framework (12), we obtain the optimization problem of the extended PMF method in the transfer learning manner with partial entity correspondences between systems as follows,

$$\begin{aligned}\min_{\mathbf{Y}^{(d)},\mathbf{V}^{(d)},\mathbf{H}^{(d)}} \sum_{(u,v)\in\mathcal{I}^{(d)}} \left(x_{uv}^{(d)} - g\left(\left(\mathbf{Y}_{*u}^{(d)} + \widehat{\mathbf{H}}_{*h}^{(d)}\right)^\top \mathbf{V}_{*v}^{(d)}\right)\right)^2 + \lambda_Y \|\mathbf{Y}^{(d)}\|_F^2 \\ + \lambda_V \|\mathbf{V}^{(d)}\|_F^2 + \lambda_H \|\mathbf{H}^{(d)}\|_F^2 + \lambda_{\mathcal{C}} \text{tr}\left(\mathbf{W}_{\mathcal{C}}^{(d)} \mathbf{L}_{\mathcal{C}}^{(s)} {\mathbf{W}_{\mathcal{C}}^{(d)}}^\top\right),\end{aligned} \tag{26}$$

where

$$\widehat{\mathbf{H}}_{*h}^{(d)} = \frac{\sum_{h=1}^{n} I_{uh} \mathbf{H}_{*h}^{(d)}}{\sum_{h=1}^{n} I_{uh}}.$$

In the sequel, we denote by PMF_{TL} the optimization problem (26). For comparison on the impact of different regularization terms to transfer learning, we denote by PMF_{CMF} the optimization problem by replacing the last regularization term in

(26) by the CMF regularization term (13) as follows,

$$\min_{\mathbf{Y}^{(d)},\mathbf{V}^{(d)},\mathbf{H}^{(d)}} \sum_{(u,v)\in\mathcal{I}^{(d)}} \left(x_{uv}^{(d)} - g\left(\left(\mathbf{Y}_{*u}^{(d)} + \widehat{\mathbf{H}}_{*h}^{(d)} \right)^{\top} \mathbf{V}_{*v}^{(d)} \right) \right)^2 + \lambda_Y \|\mathbf{Y}^{(d)}\|_F^2 + \lambda_V \|\mathbf{V}^{(d)}\|_F^2 + \lambda_H \|\mathbf{H}^{(d)}\|_F^2 + \lambda_{\mathcal{C}} \left\| \mathbf{W}_{\mathcal{C}}^{(s)} - \mathbf{W}_{\mathcal{C}}^{(d)} \right\|_F^2. \quad (27)$$

4.4.2. Actively Constructing Entity Correspondences through PMF

With the probabilities on predictions generated by PMF, we present an entropy-based method for actively constructing entity correspondences across domains, which attempts to sequentially minimize the expected entropy of the predictions.

Entropy on User-Item Pairs. Suppose that PMF (11) or PMF$_{TL}$ (26) is performed on the collaborative data in the target system to learn a CF model. Given a user-item pair (u, v), the entropy of the prediction y_{uv} given by the current model can be defined as,

$$\mathcal{H}^{(d)}(u,v) = -\left(1 - z^{(d)}(u,v)\right) \log\left(1 - z^{(d)}(u,v)\right) - z^{(d)}(u,v) \log\left(z^{(d)}(u,v)\right), \quad (28)$$

where $z^{(d)}(u,v) = \mathcal{N}\left(y_{uv}^{(d)} | g(\mathbf{U}_{*u}^{(d)\top} \mathbf{V}_{*v}^{(d)}), \sigma^2\right)$, and $\mathbf{U}_{*u}^{(d)} = \mathbf{Y}_{*u}^{(d)} + \frac{\sum_{h=1}^{n} I_{uh} \mathbf{H}_{*h}^{(d)}}{\sum_{w=1}^{n} I_{uh}}$. If $x_{uv}^{(d)}$ is observed, $y_{uv}^{(d)} = x_{uv}^{(d)}$, where $x_{uv}^{(d)}$ has been transformed to $[0, 1]$ through $t(x) = \frac{x-1}{R-1}$, otherwise, $y_{uv}^{(d)} = t(r_{uv}^{(d)})$, where

$$r_{uv}^{(d)} = \arg\max_{r} \left| \mathcal{N}\left(r | g(\mathbf{U}_{*u}^{(d)\top} \mathbf{V}_{*v}^{(d)}), \sigma^2\right) \right|, \text{ and } r \in \{1, ..., R\}.$$

In this way, the *uncertainty* of a prediction on an instance can be measured the entropy of the prediction, the larger is the entropy, the more *uncertain* is the prediction. Therefore, similar to error-based certainty measure, we define $\delta^{(d)}(u, v)$ as the *negative* of the entropy of the prediction on a user-item pair (u, v):

$$\delta^{(d)}(u,v) = -\mathcal{H}^{(d)}(u,v). \quad (29)$$

Entropy on Entities. With the entropy-based $\delta^{(d)}(u, v)$ on a user-item pair defined in (29), we can define the overall uncertainty measure on a user, $\delta_u^{(d)}$, using (15), and implement the active entity selection strategy introduced in Section 4.1.2 with the extended PMF method. In the sequel, we denote by ES$_{hy}$ this active entity selection approach.

5. Experiments

5.1. Datasets and Experimental Setting

We evaluate our proposed framework on two datasets: Netflix[3] and Douban[4]. The Netflix dataset contains more than 100 millions ratings given by more than $480,000$ users on $17,770$ movies with ratings in $\{1,2,3,4,5\}$. And Douban is a popular recommendation website in China, which has over 100 millions users. It mainly provides three recommendation services, including movies, books and music with rating scale in $\{1,2,3,4,5\}$ as well.

For the Netflix dataset, we filter out movies with less than 5 ratings for our experiments. The dataset is partitioned into two parts along two disjoint sets of users with a same set of movies. One part consists of ratings given by 50% users with 1.2% rating density, which serves as the source domain. The remaining users are considered as the target domain with 0.7% rating density. For the Douban dataset, we crawl a set consisting of $12,000$ users and $100,000$ items with only movies and books. Users with less than 10 ratings are discarded. There remain $270,000$ ratings on $3,500$ books, and $1,400,000$ ratings on $8,000$ movies, given by $11,000$ users. The density of the ratings on books and movies are 0.6% and 1.5%, respectively. We consider movie ratings as the source domain and book ratings as the target domain. In this task, all users are shared but items are disjoint. Furthermore, since there are about $6,000$ movies shared by Netflix and Douban, we extract ratings on the shared movies from Netflix and Douban, respectively, and obtain $490,000$ ratings given by $120,000$ users from Douban with rating density 0.7%, and $1,600,000$ ratings given by $10,000$ users from Netflix with density 2.6%. We consider ratings on Netflix as the source domain and those on Douban as the target domain. In total, we construct three cross-system CF tasks, and denote by **Task 1: Netflix→Netflix**, **Task 2: DoubanMovie→DoubanBook** and **Task 3: Netflix→DoubanMovie**, respectively.

In the experiments, for each time, we split each target domain data into a training set of 80% preference entries and a test set of 20% preference entries, and report the average results of 10 random times. The parameters of the model, i.e., the number of latent factors k and the number of iterations T, are tuned on some hand-out data of **Task 1: Netflix→Netflix**, and fixed to all experiments.[5]

[3] http://www.netflix.com

[4] http://www.douban.com

[5] Suppose that total budget is ρ, which is the total number of correspondences to be constructed, we set the number of correspondences actively constructed in each iteration as $K = \rho/T$.

Here, $T = 10$, and $k = 5$. In all experiments, we set $K_1 = \lfloor \frac{K}{2} \rfloor$, and the regularizer weight $\lambda_{\mathcal{C}} = 0.5$. For evaluation criterion, we use Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) defined as,

$$\text{RMSE} = \sqrt{\sum_{(u,v)\in\mathcal{I}} \frac{(x_{uv} - \hat{x}_{uv})^2}{|\mathcal{I}|}},$$

$$\text{MAE} = \sum_{(u,v)\in\mathcal{I}} \frac{|x_{uv} - \hat{x}_{uv}|}{|\mathcal{I}|},$$

where x_{uv} and $\hat{x}_{uv}$ are the true and predicted ratings, respectively, and $|\mathcal{I}|$ is the number of test entries. The smaller is the value, the better is the performance.

5.2. Overall Comparison Results

In the first experiment, we qualitatively show the effectiveness of our proposed active transfer learning framework for cross-system CF compared with the following baselines:

- NoTransf without correspondences (NoTransf (w/o corr.)): to apply state-of-the-art CF models on the target-domain collaborative data directly without either active learning or transfer learning. In this paper, regarding state-of-the-art CF models for comparison, we use MMMF, RLMF, and PMF as described in Section 3.1.

- NoTransf with actively constructed correspondences (NoTransf ($x\%$ corr)): to first apply active learning to construct cross-domain entity correspondences ($x\%$ of all the available cross-system correspondences), and then align the source and target domain data to generate a unified item-user matrix. Finally, we apply state-of-the-art CF models on the unified matrix for recommendations.

- CBT: to apply the codebook-based-transfer (CBT) method on the source and target domain data for recommendations. As mentioned in Section 1, CBT does not require any entity correspondence to be constructed.

- $\mathcal{M}_{TL}$ with full correspondences ($\mathcal{M}_{TL}$ (100% corr.)): to apply the proposed transfer learning approaches on the source and target domain data with full entity correspondences for recommendations, where $\mathcal{M}$ represents RLMF, MMMF and PMF accordingly. Note that these methods, which assume all

Table 1: Overall comparison results on the three datasets in terms of RMSE. Numbers in bold font indicate the best prediction performance excluding $\mathcal{M}_{TL}$ with full correspondences.

	Methods		**Tasks**		
			Task 1	**Task 2**	**Task 3**
RMSE	CBT (w/o corr.)		0.8846 (± 0.0002)	0.8656 (± 0.0002)	0.8246 (± 0.0002)
	RLMF	NoTransf (w/o corr.)	0.8900 (± 0.0004)	0.8804 (± 0.0017)	0.8520 (± 0.0003)
		NoTransf (0.1% corr.)	0.9112 (± 0.0002)	0.8876 (± 0.0003)	0.8643 (± 0.0003)
		RLMF$_{TL}$ (0.1% corr.)	0.8658 (± 0.0001)	0.8255 (± 0.0002)	0.7624 (± 0.0004)
		RLMF$_{TL}$ (100% corr.)	0.8352 (± 0.0002)	0.7909 (± 0.0003)	0.7379 (± 0.0003)
	MMMF	NoTransf (w/o corr.)	0.8800 (± 0.0001)	0.8784 (± 0.0002)	0.8578 (± 0.0002)
		NoTransf (0.1% corr.)	0.9103 (± 0.0004)	0.8837 (± 0.0001)	0.8589 (± 0.0002)
		MMMF$_{TL}$ (0.1% corr.)	**0.8607 (± 0.0003)**	**0.8192 (± 0.0003)**	**0.7462 (± 0.0001)**
		MMMF$_{TL}$ (100% corr.)	0.8338 (± 0.0002)	0.7863 (± 0.0002)	0.7147 (± 0.0001)
	PMF	NoTransf (w/o corr.)	0.8754 (± 0.0002)	0.8780 (± 0.0003)	0.8458 (± 0.0001)
		NoTransf (0.1% corr.)	0.8803 (± 0.0002)	0.8923 (± 0.0002)	0.8527 (± 0.0003)
		PMF$_{TL}$ (0.1% corr.)	0.8702 (± 0.0002)	0.8252 (± 0.0002)	0.7662 (± 0.0006)
		PMF$_{TL}$ (100% corr.)	0.8438 (± 0.0002)	0.7879 (± 0.0005)	0.7479 (± 0.0002)

entity correspondences be available, can be considered as an upper bound of the proposed active transfer learning methods.

The overall comparison results on the three cross-domain tasks are shown in Tables 1-2. For the active learning approaches, we use MG$_{hy}$, EE$_{hy}$ and ES$_{hy}$ as proposed in Section 4.1.2 with the extended matrix factorization methods MMMF$_{TL}$, RLMF$_{TL}$, and PMF$_{TL}$, respectively. As can be observed from the rows labeled with "NoTransf (w/o corr.)" in the table, applying state-of-the-art CF models on

Table 2: Overall comparison results on the three datasets in terms of MAE. Numbers in bold font indicate the best prediction performance excluding $\mathcal{M}_{TL}$ with full correspondences.

	Methods		**Tasks**		
			Task 1	**Task 2**	**Task 3**
MAE	CBT (w/o corr.)		0.6824 (± 0.0003)	0.6642 (± 0.0002)	0.6250 (± 0.0004)
	RLMF	NoTransf (w/o corr.)	0.6976 (± 0.0003)	0.6735 (± 0.0003)	0.6545 (± 0.0002)
		NoTransf (0.1% corr.)	0.6913 (± 0.0002)	0.6784 (± 0.0001)	0.6583 (± 0.0004)
		RLMF$_{TL}$ (0.1% corr.)	0.6679 (± 0.0002)	0.6231 (± 0.0003)	0.5971 (± 0.0004)
		RLMF$_{TL}$ (100% corr.)	0.6457 (± 0.0002)	0.5942 (± 0.0002)	0.5763 (± 0.0004)
	MMMF	NoTransf (w/o corr.)	0.6812 (± 0.0003)	0.6720 (± 0.0002)	0.6583 (± 0.0004)
		NoTransf (0.1% corr.)	0.6983 (± 0.0003)	0.6784 (± 0.0002)	0.6593 (± 0.0002)
		MMMF$_{TL}$ (0.1% corr.)	**0.6602 (± 0.0002)**	**0.6174 (± 0.0004)**	**0.5503 (± 0.0002)**
		MMMF$_{TL}$ (100% corr.)	0.6343 (± 0.0005)	0.5876 (± 0.0003)	0.5387 (± 0.0002)
	PMF	NoTransf (w/o corr.)	0.6863 (± 0.0002)	0.6749 (± 0.0002)	0.6310 (± 0.0003)
		NoTransf (0.1% corr.)	0.6931 (± 0.0002)	0.6852 (± 0.0003)	0.6370 (± 0.0003)
		PMF$_{TL}$ (0.1% corr.)	0.6780 (± 0.0003)	0.6191 (± 0.0002)	0.5656 (± 0.0003)
		PMF$_{TL}$ (100% corr.)	0.6634 (± 0.0004)	0.6038 (± 0.0004)	0.5534 (± 0.0001)

the extremely sparse target domain data directly is not able to obtain precise recommendation results in terms of RMSE or MAE. The results of rows labeled with "NoTransf (0.1% corr.)" in the table suggest that aligning all the source and target data to a unified item-user matrix and then performing state-of-the-art CF models on it cannot help to boost the recommendation performance, but may even hurt the performance compared to that of applying CF models on the target domain data only. This is because the alignment makes the matrix to be factorized larger but still very sparse, resulting in a more difficult learning task. From the table

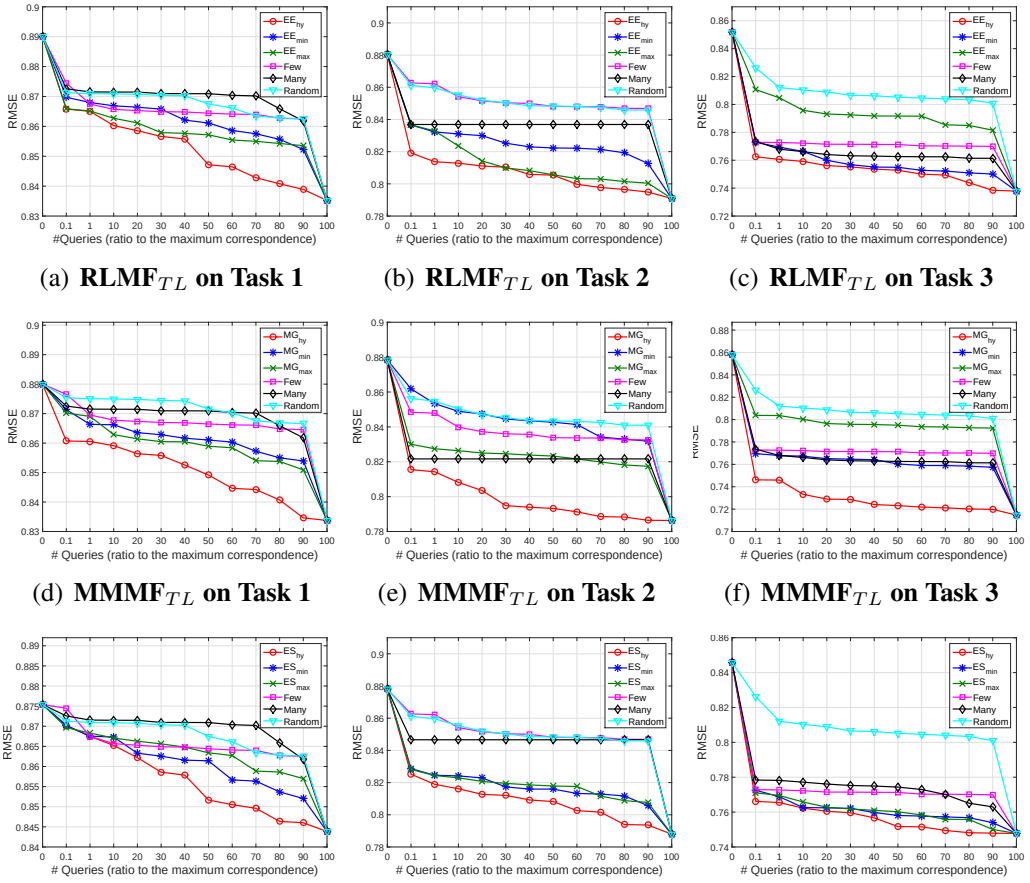
we can also observe that the transfer learning method CBT performs better than the NoTransf methods. However, our proposed active transfer learning methods RLMF$_{TL}$, MMMF$_{TL}$ and PMF$_{TL}$ with only 0.1% entity correspondences achieve better performance than CBT in terms of RMSE and MAE (around 2.2%, 4.9%, and 8.1% improvement in terms of RMSE, and 2.0%, 6.7%, and 8.6% improvement in terms of MAE over **Task 1**, **Task 2**, and **Task 3**, respectively). This verifies the conclusion that making use of cross-system entity correspondences as a bridge is useful for knowledge transfer across recommender systems. Among the three proposed methods, MMMF$_{TL}$ performs best on the three tasks. Finally, by considering the performance of active strategies with full entity correspondences as the knowledge-transfer upper bound, and the performance of base models as the lower bound, our proposed active transfer learning methods with only 0.1% entity correspondences to be cross-labeled can achieve around 34.13%, 61.41% and 79.28% in RMSE or 46.09%, 68.91% and 82.66% in MAE knowledge transfer ratio as defined in (30) on average over **Task 1**, **Task 2**, and **Task 3**, respectively.

$$\text{transfer ratio of } \mathcal{M}_{TL}(0.1\%) = \frac{\text{NoTransf (w/o corr.)} - \mathcal{M}_{TL}(0.1\%)}{\text{NoTransf (w/o corr.)} - \mathcal{M}_{TL}(100\%)}. \quad (30)$$

5.3. Experiments on Different Active Learning Strategies

In the second experiment, we aim to verify the performance of our proposed active transfer learning framework plugging with different entity selection strategies. Here, we use $\mathcal{M}_{TL}$ as the base transfer learning approach to cross-domain CF. Regarding entity selection strategies, besides the proposed approaches, MG$_{hy}$ with MMMF$_{TL}$, EE$_{hy}$ with RLMF$_{TL}$, and ES$_{hy}$ with PMF$_{TL}$, we also conduct comparison experiments on the following strategies:

- Random: at each iteration, to select K entities randomly in the target domain to query their correspondences in the source domain.
- Many: at each iteration, to select K entities with most historical ratings, i.e., the users who give most ratings on items or the items which attract most users to give ratings, in the target domain to query their correspondences in the source domain.
- Few: at each iteration, to select K entities with fewest historical ratings, i.e., the users who give fewest ratings on items or the items which attract fewest users to give ratings, in the target domain to query their correspondences in the source domain.



(a) **RLMF$_{TL}$ on Task 1** (b) **RLMF$_{TL}$ on Task 2** (c) **RLMF$_{TL}$ on Task 3**

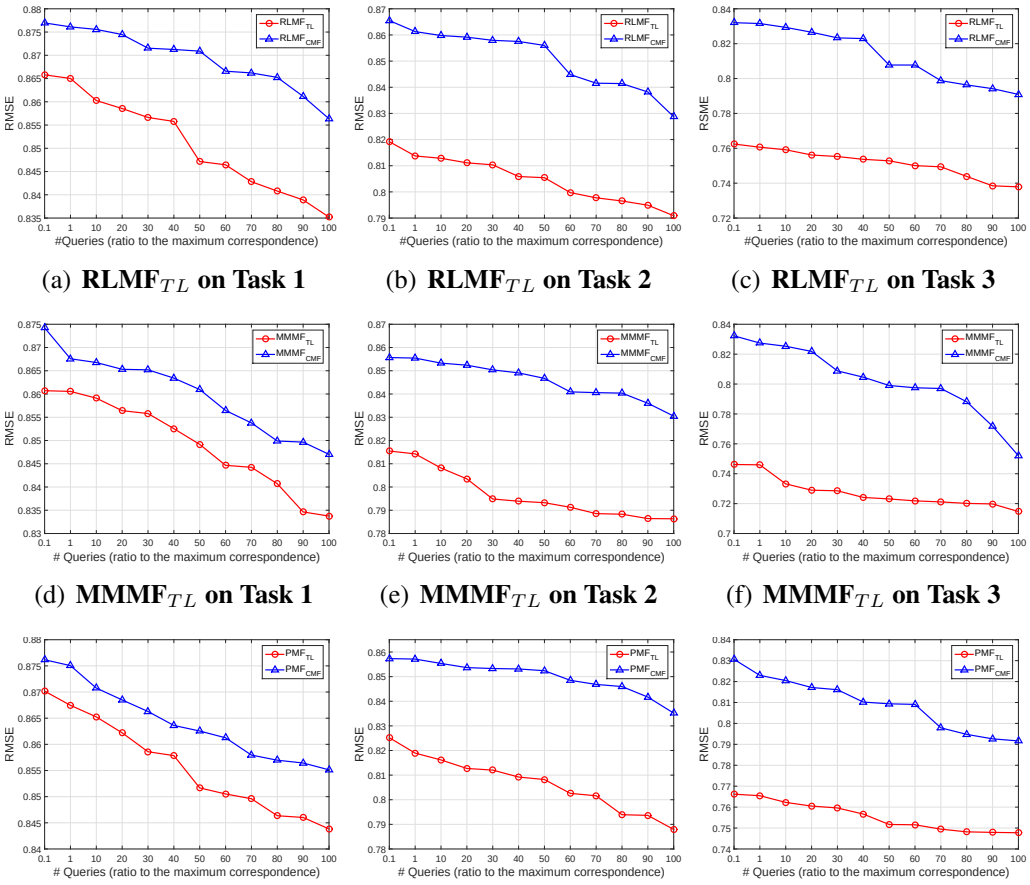
(d) **MMMF$_{TL}$ on Task 1** (e) **MMMF$_{TL}$ on Task 2** (f) **MMMF$_{TL}$ on Task 3**

(g) **PMF$_{TL}$ on Task 1** (h) **PMF$_{TL}$ on Task 2** (i) **PMF$_{TL}$ on Task 3**

Figure 3: Results on different entity selection strategies under varying proportions of entity correspondences to be labeled.

- MG$_{min}$, EE$_{min}$, ES$_{min}$: at each iteration, to select K entities whose predicted ratings are of the most **uncertainty** based on $\delta_u^{(d)}$ or $\delta_v^{(d)}$ by using MMMF$_{TL}$, RLMF$_{TL}$, and PMF$_{TL}$, respectively, in the target domain to query their correspondences in the source domain.

- MG$_{max}$, EE$_{max}$, ES$_{max}$: at each iteration, to select K entities whose predicted ratings are of the most **certainty** based on $\delta_u^{(d)}$ or $\delta_v^{(d)}$ by using MMMF$_{TL}$, RLMF$_{TL}$, and PMF$_{TL}$, respectively, in the target domain to query their correspondences in the source domain.



(a) **RLMF$_{TL}$ on Task 1** (b) **RLMF$_{TL}$ on Task 2** (c) **RLMF$_{TL}$ on Task 3**

(d) **MMMF$_{TL}$ on Task 1** (e) **MMMF$_{TL}$ on Task 2** (f) **MMMF$_{TL}$ on Task 3**

(g) **PMF$_{TL}$ on Task 1** (h) **PMF$_{TL}$ on Task 2** (i) **PMF$_{TL}$ on Task 3**

Figure 4: Results on different cross-domain regularizers under varying proportions of entity correspondences to be labeled.

Figure 3 shows the results of $\mathcal{M}_{TL}$ with different entity selection strategies under varying proportions of entity correspondences to be constructed. From the figure, we can observe that the active approaches based on the entity-level margin, error and entropy (i.e., $\mathcal{A}_{min}$, $\mathcal{A}_{max}$, and $\mathcal{A}_{hy}$, where $\mathcal{A}$ represents MG, EE, ES, respectively) perform much better than other approaches. In addition, compared with $\mathcal{A}_{min}$ and $\mathcal{A}_{max}$, the proposed $\mathcal{A}_{hy}$ can avoid selecting long-tail users in the source domain for knowledge transfer, thus performs best.

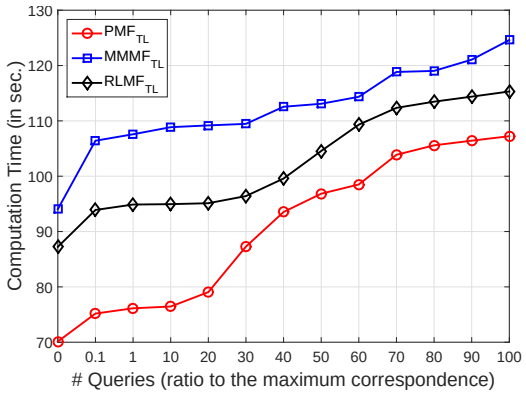


Figure 5: Computation time analysis on the proposed active transfer learning approaches.

5.4. Experiments on Different Cross-domain Regularization Terms

As mentioned in Section 4, the regularization term $\mathcal{R}(\mathbf{U}_{\mathcal{C}}^{(d)}, \mathbf{V}_{\mathcal{C}}^{(d)}, \mathbf{U}_{\mathcal{C}}^{(s)}, \mathbf{V}_{\mathcal{C}}^{(s)})$ in (12) for cross-system knowledge transfer can be substituted by different forms, e.g., (13) or (14), which results in different transfer learning approaches, $\mathcal{M}_{CMF}$ or $\mathcal{M}_{TL}$ accordingly. Therefore, in the third experiment, we use $\mathcal{A}_{hy}$ as the entity selection strategy, and compare the performance of $\mathcal{M}_{TL}$ and $\mathcal{M}_{CMF}$ in terms of RMSE. As can be seen from Figure 4, the proposed $\mathcal{M}_{TL}$ outperforms its corresponding $\mathcal{M}_{CMF}$ consistently on the three cross-system tasks under varying proportions of the labeled entity correspondences. This implies that using similarities between entities from the source domain data as priors is more safe and useful for knowledge transfer across recommender systems than using the factor matrices factorized from the source domain data as priors directly.

5.5. Computational Time Analysis

For the last experiment, we study computational time of the proposed three active transfer learning approaches. The computer used for running the computational time comparison experiments is equipped with 1.4GHz Intel Core i5, 8GB memory and 512GB SSD. Comparison results under varying proportions of entity correspondences to be constructed on **Task 1** are shown in Figure 5. Note that when the proportion of entity correspondences equals to 0, the proposed active transfer learning approaches, MMMF$_{TL}$, RLMF$_{TL}$, and PMF$_{TL}$, are reduced to the NoTransf methods without correspondences, i.e., MMMF, RLMF, and PMF using the target domain data only, respectively. As the matrix factorization on the source system data can be pre-trained, the reported computational time does

not include the time on generating $\mathbf{U}^{(s)}$ and $\mathbf{V}^{(s)}$ for the source system. Furthermore, we also ignore the time on manually labeling cross-system entity correspondences. From the figure, we can find that as $\mathcal{M}_{TL}$ aims to exploit cross-system entity correspondences to transfer knowledge from the source system to the target system, the computational time increases when the number of the constructed cross-system entity correspondences increases. However, when the proportion of the constructed entity correspondences is not larger than 20%, the computational time of $\mathcal{M}_{TL}$ is very close to its corresponding NoTransf method without correspondences, respectively. From the figure, we can also observe that among the three active transfer learning approaches, MMMF$_{TL}$'s computation cost is most expensive, while PMF$_{TL}$ is the most computationally efficient. In fact, the computational time of the active transfer learning approach depends on its base matrix factorization method. In practice, parallel or distributed matrix factorization techniques can be adopted to significantly boost the computational efficiency of the proposed solutions [49, 50, 51, 52, 53]. However, it is beyond the scope of this work.

Together with the results shown on Tables 1-2, we may conclude that among the three proposed active transfer learning approaches, if prediction accuracy is of the most priority, then MMMF$_{TL}$ is the best choice. If computational efficiency is of the most priority, then PMF$_{TL}$ is the best choice. RLMF$_{TL}$ can be considered as a trade-off solution. However, it should be emphasized again that the focus of this work is not discussing which matrix factorization method can be adapted into our framework to achieve the best performance for knowledge transfer across different recommender systems, but providing a general active transfer learning framework, where researchers can extend their favor matrix factorization methods for different applications and datasets.

6. Conclusions

In this paper, we present a novel framework on active transfer learning for cross-system recommendations. In the proposed framework, we 1) extend previous transfer learning approaches to CF in a flexible entity corresponding manner, and 2) propose an entity selection strategy to actively construct entity correspondences across different recommender systems. In particular, we develop three specific solutions based on the framework. Our experimental results show that compared with the transfer learning method which requires full entity correspondences, our proposed framework can achieve around around 34.13%, 61.41% and 79.28% in RMSE or 46.09%, 68.91% and 82.66% in MAE knowledge-transfer

ratio, while only requires 0.1% of the entities to have correspondences. For future work, we are planning to apply the proposed framework to other applications, such as cross-system link prediction in social networks.

7. Acknowledgement

Lili Zhao and Qiang Yang thank the support of China National 973 project 2014CB340304 and Hong Kong CERG projects 16211214, 16209715 and 16244616. Sinno J. Pan thanks the support of the NTU Singapore Nanyang Assistant Professorship (NAP) grant M4081532.020.

References

[1] F. J. Martin, J. Donaldson, A. Ashenfelter, M. Torrens, R. Hangartner, The big promise of recommender systems, AI Magazine 32 (3) (2011) 19–27.

[2] R. D. Burke, A. Felfernig, M. H. Göker, Recommender systems: An overview, AI Magazine 32 (3) (2011) 13–18.

[3] D. Jannach, M. Zanker, A. Felfernig, G. Friedrich, Recommender Systems - An Introduction, Cambridge University Press, 2010.

[4] S. J. Pan, Q. Yang, A survey on transfer learning, IEEE Transactions on Knowledge and Data Engineering 22 (10) (2010) 1345–1359.

[5] S. J. Pan, Transfer learning, in: C. C. Aggarwal (Ed.), Data Classification: Algorithms and Applications, CRC Press, 2014, pp. 537–570.

[6] B. Li, Q. Yang, X. Xue, Can movies and books collaborate?: cross-domain collaborative filtering for sparsity reduction, in: IJCAI, 2009, pp. 2052–2057.

[7] B. Li, Q. Yang, X. Xue, Transfer learning for collaborative filtering via a rating-matrix generative model, in: ICML, 2009, pp. 617–624.

[8] B. Mehta, T. Hofmann, Cross system personalization and collaborative filtering by learning manifold alignments, in: KI, 2006, pp. 244–259.

[9] W. Pan, E. W. Xiang, N. N. Liu, Q. Yang, Transfer learning in collaborative filtering for sparsity reduction, in: AAAI, 2010.

[10] W. Pan, Q. Yang, Transfer learning in heterogeneous collaborative filtering domains, Artifical Intellgence 197 (2013) 39–55.

[11] J. Lee, M. Sun, G. Lebanon, A comparative study of collaborative filtering algorithms, CoRR abs/1205.3193.

[12] L. Zhao, S. J. Pan, E. W. Xiang, E. Zhong, Z. Lu, Q. Yang, Active transfer learning for cross-system recommendation, in: AAAI, 2013.

[13] D. Goldberg, D. Nichols, B. M. Oki, D. Terry, Using collaborative filtering to weave an information tapestry, Communication of the ACM 35 (12) (1992) 61–70.

[14] J. S. Breese, D. Heckerman, C. Kadie, Empirical analysis of predictive algorithms for collaborative filtering, in: UAI, 1998, pp. 43–52.

[15] A. A. Falkner, A. Felfernig, A. Haag, Recommendation technologies for configurable products, AI Magazine 32 (3) (2011) 99–108.

[16] B. Mobasher, J. Cleland-Huang, Recommender systems in requirements engineering, AI Magazine 32 (3) (2011) 81–89.

[17] Ò. Celma, P. Lamere, If you like radiohead, you might like this article, AI Magazine 32 (3) (2011) 57–66.

[18] R. D. Burke, J. Gemmell, A. Hotho, R. Jäschke, Recommendation in the social web, AI Magazine 32 (3) (2011) 46–56.

[19] Y. Koren, R. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, Computer 42 (2009) 30–37.

[20] W. Pan, E. W. Xiang, Q. Yang, Transfer learning in collaborative filtering with uncertain ratings, in: AAAI, 2012.

[21] B. Cao, N. N. Liu, Q. Yang, Transfer learning for collective link prediction in multiple heterogenous domains, in: ICML, 2010, pp. 159–166.

[22] Y. Zhang, B. Cao, D.-Y. Yeung, Multi-domain collaborative filtering, in: UAI, 2010, pp. 725–732.

[23] A. P. Singh, G. J. Gordon, Relational learning via collective matrix factorization, in: KDD, 2008, pp. 650–658.

[24] J. Tang, J. Yan, L. Ji, M. Zhang, S. Guo, N. Liu, X. Wang, Z. Chen, Collaborative users' brand preference mining across multiple domains from implicit feedbacks, in: AAAI, 2011.

[25] L. Shi, Y. Zhao, J. Tang, Batch mode active learning for networked data, ACM Transaction on Intelligent Systems and Technology 3 (2) (2012) 33:1–33:25.

[26] C. E. Mello, M.-A. Aufaure, G. Zimbrao, Active learning driven by rating impact analysis, in: RecSys, 2010, pp. 341–344.

[27] I. Rish, G. Tesauro, Active collaborative prediction with maximum margin matrix factorization, in: ISAIM, 2008.

[28] R. Jin, L. Si, A bayesian approach toward active learning for collaborative filtering, in: UAI, 2004, pp. 278–285.

[29] C. Boutilier, R. S. Zemel, B. Marlin, Active collaborative filtering, in: UAI, 2003, pp. 98–106.

[30] S. Raj, J. Ghosh, M. M. Crawford, An active learning approach to knowledge transfer for hyperspectral data analysis, in: IGARSS, 2006, pp. 541–544.

[31] Y. S. Chan, H. T. Ng, Domain adaptation with active learning for word sense disambiguation, in: ACL, 2007.

[32] X. Shi, W. Fan, J. Ren, Actively transfer domain knowledge, in: ECML/PKDD (2), 2008, pp. 342–357.

[33] P. Rai, A. Saha, H. Daumé, III, S. Venkatasubramanian, Domain adaptation meets active learning, in: NAACL HLT 2010 Workshop on Active Learning for Natural Language Processing, 2010, pp. 27–32.

[34] A. Saha, P. Rai, H. D. III, S. Venkatasubramanian, S. L. DuVall, Active supervised domain adaptation, in: ECML/PKDD (3), 2011, pp. 97–112.

[35] R. Chattopadhyay, W. Fan, I. Davidson, S. Panchanathan, J. Ye, Joint transfer and batch-mode active learning, in: ICML, 2013, pp. 253–261.

[36] X. Wang, T.-K. Huang, J. Schneider, Active transfer learning under model shift, in: ICML, 2014, pp. 1305–1313.

[37] X. Liao, Y. Xue, L. Carin, Logistic regression with an auxiliary data source, in: ICML, 2005, pp. 505–512.

[38] J. D. M. Rennie, N. Srebro, Fast maximum margin matrix factorization for collaborative prediction, in: ICML, 2005, pp. 713–719.

[39] A. Paterek, Improving regularized singular value decomposition for collaborative filtering, in: KDD Cup and Workshop, 2007.

[40] R. M. Bell, Y. Koren, Lessons from the netflix prize challenge, SIGKDD Explorations 9 (2) (2007) 75–79.

[41] N. Srebro, J. D. M. Rennie, T. S. Jaakkola, Maximum-margin matrix factorization, in: NIPS 17, 2005, pp. 1329–1336.

[42] R. Salakhutdinov, A. Mnih, Probabilistic matrix factorization, in: NIPS, 2007.

[43] M. Belkin, P. Niyogi, Laplacian eigenmaps and spectral techniques for embedding and clustering, in: NIPS, 2001, pp. 585–591.

[44] W.-J. Li, D.-Y. Yeung, Relation regularized matrix factorization, in: IJCAI, 2009, pp. 1126–1131.

[45] B. Settles, Active learning literature survey, Technical Report, University of Wisconsin–Madison (2009).

[46] S. Tong, D. Koller, Support vector machine active learning with applications to text classification, Journal of Machine Learning Research 2 (2002) 45–66.

[47] D. Roth, K. Small, Margin-based active learning for structured output spaces, in: ECML, 2006, pp. 413–424.

[48] M. Balcan, A. Z. Broder, T. Zhang, Margin based active learning, in: COLT, 2007, pp. 35–50.

[49] H.-F. Yu, C.-J. Hsieh, S. Si, I. Dhillon, Scalable coordinate descent approaches to parallel matrix factorization for recommender systems, in: ICDM, 2012, pp. 765–774.

[50] H.-F. Yu, C.-J. Hsieh, S. Si, I. S. Dhillon, Parallel matrix factorization for recommender systems, Knowledge and Information Systems 41 (3) (2014) 793–819.

[51] R. Gemulla, E. Nijkamp, P. J. Haas, Y. Sismanis, Large-scale matrix factorization with distributed stochastic gradient descent, in: KDD, 2011, pp. 69–77.

[52] S. Ahn, A. Korattikara, N. Liu, S. Rajan, M. Welling, Large-scale distributed bayesian matrix factorization using stochastic gradient mcmc, in: KDD, 2015, pp. 9–18.

[53] M. Li, Z. Liu, A. J. Smola, Y. Wang, Difacto: Distributed factorization machines, in: WSDM, 2016, pp. 377–386.