A cross-domain recommender system with consistent information transfer

Qian Zhang, Dianshuang Wu, Jie Lu, Feng Liu, Guangquan Zhang

Decision Systems and e-Service Intelligence Laboratory, Centre for Artificial Intelligence, Faculty of Engineering and Information Technology, University of Technology Sydney, Australia

E-mail: qian.zhang-11@student.uts.edu.au, dianshuang.wu@uts.edu.au, jie.lu@uts.edu.au, feng.liu-2@student.uts.edu.au, guangquan.zhang@uts.edu.au

Abstract

Recommender systems provide users with personalized online product and service recommendations and are a ubiquitous part of today's online entertainment smorgasbord. However, many suffer from cold-start problems due to a lack of sufficient preference data, and this is hindering their development. Cross-domain recommender systems have been proposed as one possible solution. These systems transfer knowledge from one domain that has adequate preference information to another domain that does not. The outlook for cross-domain recommendation is promising, but existing methods cannot ensure the knowledge extracted from the source domain is consistent with the target domain, which may impact the accuracy of the recommendations. To address this challenging issue, we propose a cross-domain recommender system with consistent information transfer (CIT). Knowledge consistency is based on user and item latent groups, and domain adaptation techniques are used to map and adjust these groups in both domains to maintain consistency during the transfer learning process. Experiments were conducted on five real-world datasets in three categories: movies, books, and music. The results for nine cross-domain recommendation tasks show that CIT outperforms five benchmarks and increases the accuracy of recommendations in the target domain, especially with sparse data. Practically, our proposed method is applied into a telecom product recommender system and a business partner recommender system (SmartBizSeeker) to enhance personalized decision making for both businesses and individual customers.

Keywords: Recommender systems, cross-domain recommender system, knowledge transfer, collaborative

filtering

1. Introduction

Recommender systems, which aim to provide users with personalized services and overcome the information overload problems, have been developed for more than twenty years [1]. The mainly used recommendation techniques are roughly divided into content-based and collaborative filtering-based. Without content restriction, collaborative filtering is more widely used in areas where users express their preferences by rating items, such as movies, books, and music. Over the last two decades, collaborative filtering has been comprehensively explored from basic memory-based methods [2] to various model-based methods such as matrix factorization [3], probabilistic models [4] and deep learning models [5]. However, sparsity, or the cold-start problem, remains the most challenging outstanding issue in collaborative filtering [6]. If a system fails to provide practical support, new users will quickly lose interest and stop using it [7]. To solve the cold-start problem, traditional methods aim to find additional information, such as social network [8], trust [9] or reviews [10] from within the same domain to infer user-item relationships. Unfortunately, additional information is not often available.

However, where there is insufficient data in one domain, such as movies, but relatively rich data in another domain, such as books. Transfer learning can be used to overcome cold-start problems if the two domains are either explicitly or implicitly related [11]. Moreover, transfer learning and collaborative filtering can be combined to extract knowledge from a source domain with sufficient data to increase recommendation accuracy in a target domain. In this way, a newly launched recommender system in one domain is able to benefit from a mature recommender system in another domain. Such systems are known as a cross-domain recommender system (CDRS) [12]. Because of advantages of collaborative filtering, such as its high efficiency and its lack of content restrictions, CDRSs provide relatively high-quality recommendation together with the ability to deal with cold start problems.

CDRSs aim to use information from an alternative source domain in the target domain where sufficient preference data is unavailable. CDRSs are developed into two directions. One collectively uses preference data from both domains, while the other tries to connect the domains through other information, such as the users' social relations [13] or the items' attributes [14]. Our research focuses solely on preference data since it is not restricted by other information and universally applicable. CDRSs based on preference data can be generally divided into two classes. The first class deals with situations where users and items in the source domain are either totally or partially mapped to those in the target domain [15-17]. However, these methods cannot use data without corresponding users or items in the target domain. The second class deals with situations where there are no intersections between the two domains [18, 19]. This scenario is more widely seen in real-world applications. Sharing user ID from different data source is almost impossible due to confidential user information. Our research falls into the scope of CDRS handling preference data without intersections between two domains.

Existing CDRS methods for preference data without intersections between two domains use shared information of users and items despite a lack of direct corresponding between domains. For example, a group of well-clustered users implies similar preference information, and a group of well-clustered items implies similar content information. From such groups, a user group to item group rating pattern, defined as group-level knowledge, can be extracted and shared as a compressed form of the original user-item rating matrix. These methods partly alleviate the sparsity problem and increase the prediction accuracy of recommender systems in target domain. However, none positively transfer knowledge to the target domain in a stable manner, which reduces the accuracy of the recommendations when there is shift between domains. Some methods are prone to failure because they use the group-level knowledge matrix directly without ensuring the consistency of the user/item group information is maintained during transfer. Without collectively clustering or adjusting the group-level knowledge, it usually diverges between domains. Obviously, integrating inconsistent knowledge into the target domain causes harm, rather than helping the recommender system. By ensuring the consistency of the knowledge transferred between the domains, we aim to increase the prediction accuracy of CDRSs and overcome some general problems associated with domain shift in real-world decision-making applications.

In this paper, we investigate how to effectively transfer knowledge from the source rating matrix to help increase the prediction accuracy of the recommender system on the target rating matrix. To avoid divergence caused by domains, group level knowledge is extracted on the basis of consistent user/item group information. That is, user/item information should be consistent in each corresponding group from source and target domain. A domain adaptation technique regulates user/item group information in both domains. Then group-level knowledge is learned to maximize the overall level of fitting in both domains. Thus, a cross-domain recommender system with consistent information transfer (CIT) is proposed as a knowledge transfer method. The main contributions of this paper are:

(1). A definition for "Consistent knowledge" to answer the essential question of "what to transfer" in CDRSs. We argue that information should be consistent for each user and item group so that group-level knowledge can be shared. In this way, the requirement for when group-level knowledge can be transferred is addressed, which has not been considered by previous CDRSs.

(2). A domain adaptation method that matches and adjusts user and item latent groups to maintain the consistency of group information. The group-level knowledge learned on this basis represents the shared characteristics of both domains, which can help to ensure positive transfer between the domains.

(3). An adaptive knowledge transfer method for CDRSs, called CIT. This method lessens the reduction in accuracy caused by insufficient data in the target domain. It improves the performance of immature recommender systems by transferring knowledge from another related but different domain.

The remainder of the paper is organized as follows. Section 2 contains a review of work related to CDRSs. Section 3 formally defines the problem solved. In Section 4, we present our CIT method in three parts: an overview, the steps, and the conceptual framework of the cross-domain recommender system. Section 5 presents the empirical experiments on five real-world datasets spanning three categories of data. The results for nine tasks in terms of three data sparsity ratios show that our method is better than five existing non-transfer and cross-domain methods. Finally, the discussion, conclusion and directions for future study are provided in Section 6. Guidelines for recommender system developers along with a discussion on the potential industry applications of the proposed method are included.

2. Related Work

In this Section, related works about CDRSs are reviewed.

As mentioned in Introduction, two different types of CDRSs have been developed. Some methods that connect two domains through other information rather than preference data are as follows: FUSE [13] integrates social information with preference data by sharing implicit cluster-level tensers from multiple domains. Collective matrix factorization (CMF) [14] factorizes the source rating matrix and the target rating matrix concurrently by sharing parameters when the user or item is found in both domains. This method is especially suitable with item attribute information or information contributed by users.

On the other hand, CDRSs based on preference data can be designed in various ways according to the overlap of users and item, the form the data takes, or the tasks the system needs to handle. Methods dealing with data where user/item partially or fully corresponds in both domains usually collectively factorize two matrixes in each domain by sharing part of the factorization parameters. Cross-domain triadic factorization (CDTF) [17] models the relation of a user-item-domain to extract the interactions of items in different domains. Clustering-based matrix factorization (CBMF) [15] subsequently tried to improve CDTF by utilizing information from unobserved ratings at a cluster level. These two methods work well in situations where users have ratings in multiple domains with different sparsity, i.e., where the user information fully overlaps. A large e-commerce website housing various products or services is a good example. Rating over site-time (ROST) [20] is similar to the two methods above, but it also considers the user-interest drift in different time-windows. In this situation, users/items are partly or fully overlapped. Transfer by collective factorization (TCF) [16] explores how to use implicit binary preference data in the source domain to assist recommendations in the target domain with explicit rating data. Since the data in both domains are heterogeneous, it requires that users and items in the source and target rating matrixes have one-to-one mappings. All these methods above have their own application scenarios, but they cannot be used when data from two domains have non-overlapped users/items.

Methods that handle two domains with no intersections of users/items usually transfer knowledge between the domains on a group level. Codebook transfer (CBT) [18] extracts knowledge from the source rating matrix as a 'codebook'. In this method, the source rating matrix must be full; hence, it is filled with the mean ratings of each user. The rating matrix generative model (RMGM) [19] was extended from CBT. It avoids the full matrix limitation by relaxing the hard membership constraint on user/item groups. Our research falls within the scope of methods without any user/item overlap. However, a specific definition of "consistent knowledge" is not given in the existing literature. By default, two rating matrixes are taken from source and target domains and factorized to acquire the shared knowledge. But in our proposed CIT, we defined how two rating matrixes are consistently tri-factorized and how consistent knowledge can be extracted, which helps to improve the recommendation performance in the target domain. This makes our method different from previous works. The related works in this Section are summarized in Table 1.

Table 1 Summary of related works

	user/item overlap			data			tasks	
				preference data only		Other data needed		
	full overlap	partly overlap	non-overlap	heterogeneous	homogeneous		Two domains	Multi-domain
FUSE [13]			×			×		×
CMF [14]		×				×		×
CBMF [15]		×			×		×	
CDTF [17]		×			×			×
TCF [16]	×			×			×	
CBT [18]			×		×		×	
RMGM [19]			×		×			×
ROST [20]	×				×			×
our CIT			×		×		×	

3. Problem Formulation and Motivation

In this section, a factorization view of the recommender system in one domain is given to clearly describe the problem setting. The problem under study in this paper is then formally described. Finally, the motivation of this research is given as an example.

3.1 Recommendation Task based on Tri-factorization in One Domain

In a single domain, suppose there are M users and N items. The relationship between the users and the items is represented by the user-item rating matrix $\boldsymbol{X} \in \mathbb{R}^{M \times N}$ (bold letters represent a matrix). Any rating r_{ij} in $\boldsymbol{X}$ is subject to $r_{ij} \in \{1,2,3,4,5,?\}$ ("?" denotes a missing value). To construct the group-level knowledge matrix, users and items are clustered. The rating matrix $\boldsymbol{X}$ can be factorized into three matrixes

[21]: $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T$, where $\boldsymbol{U} \in \mathbb{R}^{M\times K}$ is the user-group membership matrix, $\boldsymbol{V} \in \mathbb{R}^{N\times L}$ is the item-group membership matrix, and $\boldsymbol{S} \in \mathbb{R}^{K\times L}$ is the group-level knowledge matrix. Each row of $\boldsymbol{U}$ and $\boldsymbol{V}$ contains the memberships of the user/item entity for all groups. $\boldsymbol{S}$ is the rating pattern of each user group to each item group.

The recommendation task requires the prediction of user ratings for items where the rating values are not known. To calculate the missing values, the user-item rating matrix is reconstructed through $\widehat{\boldsymbol{X}} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T$. Tri-factorization of $\boldsymbol{X}$ minimizes the loss function $L(\boldsymbol{X}, \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T)$, which measures the error of prediction. Since $\boldsymbol{X}$ is usually sparse, the loss function is in a weighted form as follows:

$$L(\boldsymbol{X}, \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T) = \| \boldsymbol{W} \odot (\boldsymbol{X} - \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T) \|^F \quad (1)$$

where $\odot$ denotes the element-wise product of matrixes, and $\boldsymbol{W}$ is the indicator matrix representing whether the rating in $\boldsymbol{X}$ is observed or not. Thus, in single-domain recommendation, $\boldsymbol{\Theta} = \{\boldsymbol{U}, \boldsymbol{S}, \boldsymbol{V}\}$ are the parameters the recommender system uses to predict the ratings and provide a recommendation. The tri-factorization is

$$\min L(\boldsymbol{X}, \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T)$$

$$s.t.\ \boldsymbol{U} > 0, \boldsymbol{S} > 0, \boldsymbol{V} > 0$$

3.2 Cross-domain Transfer Learning Recommender System

As mentioned in the Introduction, users and items are usually denoted by de-identified user and item IDs, it is often difficult to find an explicit correlation between the two domains. In this problem setting, the users/items have no correspondence across the domains and are treated as completely different users/items. We assume that explicit rating data are available for both the source and target domains. Formally, the problem is defined as:

Definition 1 (Cross-domain Transfer Learning Recommender System). Given a source rating matrix $\boldsymbol{X}_s \in \mathbb{R}^{M_s\times N_s}$ and a target rating matrix $\boldsymbol{X}_t \in \mathbb{R}^{M_t\times N_t}$, a cross-domain transfer learning recommender system aims to help recommendation tasks in the target domain predict the rating $\widehat{\boldsymbol{X}}_t = \boldsymbol{U}_t\boldsymbol{S}_t\boldsymbol{V}_t^T$ using knowledge in the source rating matrix $\boldsymbol{X}_s$ and $\boldsymbol{\Theta}_s = \{\boldsymbol{U}_s, \boldsymbol{S}_s, \boldsymbol{V}_s\}$, where $P_s \cap P_t = \emptyset$ and $Q_s \cap Q_t = \emptyset$. P_s and Q_s represent the user set and item set in the source domain, while P_t and Q_t represent the user set and item set in the target domain.

3.3 Motivation for developing CIT

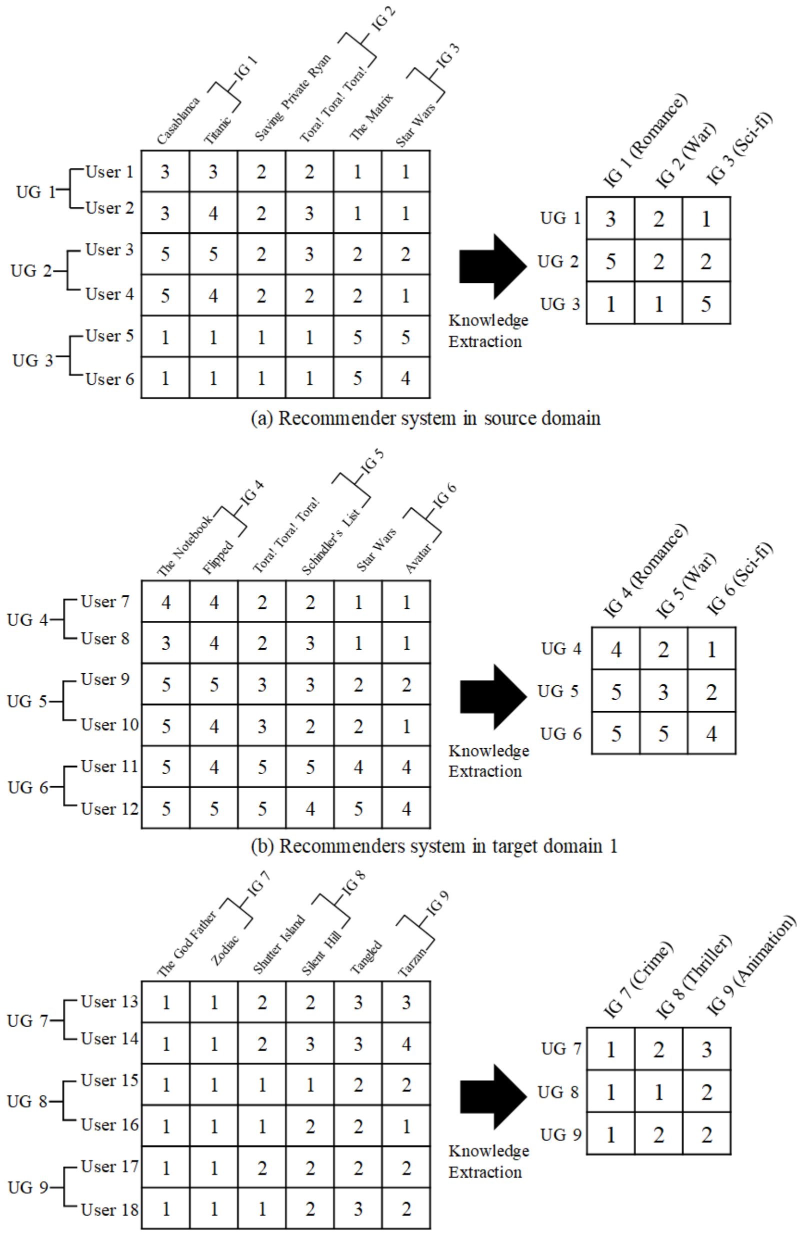
A CDRS for movies serves as a good example for describing this problem. Consider three movie rating websites. Two sites focus on classic movies (the source domain and target domain 1); the other only contains second-rate movies (target domain 2). Fig. 1 illustrates three scenarios.

Scenario 1: Users 1-4 in Fig. 1 (a) and users 7-10 in Fig. 1 (b). Although the chosen movies have different origins, all the movie subsets from the source domain and target domain 1 are quite similar. Users 1-4 in the source domain and users 7-10 in the target domain 1 also have similar movie preferences; hence, the user and item groups contain similar information in the source and target domains. In this first scenario, using the group-level knowledge directly in the target domain is effective even though there is no group-matching module.

Scenario 2: Users 5, 6 in Fig. 1 (a) and users 11, 12 in Fig. 1 (b). UG6 in target domain 1 has completely different information to UG3 in the source domain. Because the group-level knowledge is inconsistent (here, due to the user preference information), directly transferring that knowledge from source domain to target domain will impair the performance of the CDRS.

Scenario 3: Users 1-6 in Fig. 1 (a) and users 13-18 in Fig. 1 (c). As in scenario 2, UGs 7-9 have completely different group information from UGs 1-3 in the source domain, as is the case with IG 1-3 and IG 7-9. Here, both the user preference information and the item content information are inconsistent. As a result, using knowledge extracted from the source domain in target domain 2 may produce even poorer recommendations than from a recommender system that was built solely from target domain 2.

These scenarios reflect the knowledge inconsistency problem that existing CDRSs are unable to deal with. Using knowledge from another domain without mapping and adjustment only helps to produce a more accurate prediction if there is no significant divergence between the source domain and target domain. The CIT method, described in the following section, helps to solve the problem.



(c) Recommender system in target domain 2

Fig. 1. An example for CDRS.

(a)-(c) Recommender systems for a source domain, target domain 1 and target domain 2. The left side shows the schematic rating matrixes; the right side shows the group-level knowledge matrixes. These represent possible groups of users and items from the left-side rating matrixes. The possible user/item group semantic meanings are annotated as UG - user group and IG - item group.

4. A Cross-domain Recommender System with Consistent Information Transfer

This section introduces our CIT method beginning with an overview of the entire procedure. Each of the five steps of the method are then presented in detail followed by the system architecture to support decision-making for individuals and businesses.

4.1 CIT Method Overview

The proposed CIT method uses a domain adaptation technique to ensure that knowledge extracted from the source domain is consistent with the target domain and that knowledge transfer is positive. The procedure consists of five steps, as shown in Fig. 2. 1). Users/items from the source and target domains are clustered separately into groups. 2). Domain adaptation techniques are used to generate consistent user/item latent groups in the source and target domains. 3). Consistent knowledge is extracted from the latent groups. 4). Group representations in the target domain are adjusted to retain their domain-specific characteristics. 5). A recommender system for the target domain is built. We use a specific algorithm for each step, but other clustering or domain adaptation algorithms could be substituted.

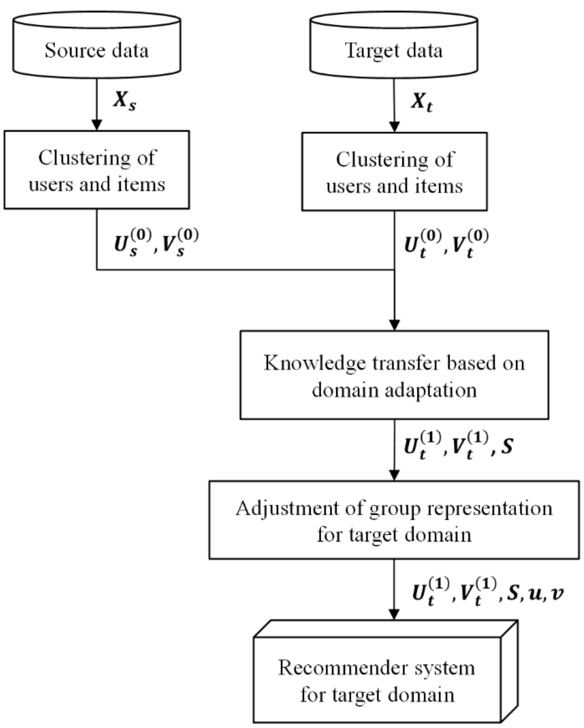


Fig. 2. The CIT method procedure

Note: The notations in the figure correspond to the equations that follow in this section.

4.2 CIT Method

Our proposed CIT method consists of five steps.

4.2.1 Step 1: Clustering of users and items in both domains

This step clusters users and items into groups. Clustering users and items appropriately is a crucial issue. Intuitively, users may have various preferences and items may have diverse content. Therefore, it is usually more appropriate to allow both users and items to fall into multiple groups with different memberships. Thus, in this paper, a flexible mixture model (FMM) [22] is used to cluster the users and items separately. The same clustering procedure is used for both the source domain and the target domain; however, for simplicity, we have only provided the description for one domain.

Suppose users are clustered into K user groups $\{Z_u^{(1)}, \dots, Z_u^{(K)}\}$, while items are clustered into L item groups $\{Z_v^{(1)}, \dots, Z_v^{(L)}\}$. Z_u and Z_v are two latent variables that denote the user and item groups respectively. $P(Z_u|u)$ is the conditional probability of a user belonging to a user group, denoting the group membership of the user; $P(Z_v|v)$ is the conditional probability of an item belonging to an item group, denoting its group membership. Each user group has a rating preference for each item group. r is the variable representing the preference of user groups to item groups. $P(r|Z_u, Z_v)$ is the conditional probability of r given user group Z_u and item group Z_v. The rating for a coupled user-item pair is:

$$R(u,v) = \sum_r r \sum_{Z_u, Z_v} P(r|Z_u, Z_v)\, P(Z_u|u) P(Z_v|v) \quad (2)$$

Equation (2) can be rewritten into matrix form:

$$\boldsymbol{X} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T \quad (3)$$

where $\boldsymbol{U} \in \mathbb{R}^{M\times K}$ and $\boldsymbol{V} \in \mathbb{R}^{N\times L}$ are the user and item group membership matrix. $\boldsymbol{U}_{ij}$ represents the membership of user u_i for user group $Z_u^{(j)}$. $\boldsymbol{U}_{i*}$ is the ith row of matrix $\boldsymbol{U}$ representing membership of user u_i to each group. $\boldsymbol{U}_{*j}$ is the jth column of matrix $\boldsymbol{U}$ representing the membership of each user to user group $Z_u^{(j)}$. The same goes for items. $\boldsymbol{S} \in \mathbb{R}^{K\times L}$ is the group-level knowledge matrix. $\boldsymbol{S}_{ij}$ represents the preference of user group $Z_u^{(i)}$ for item group $Z_v^{(j)}$.

After clustering, the user group and item group membership matrixes $\boldsymbol{U}_s^{(\boldsymbol{0})}, \boldsymbol{V}_s^{(\boldsymbol{0})}$ are acquired for the source domain and $\boldsymbol{U}_t^{(\boldsymbol{0})}, \boldsymbol{V}_t^{(\boldsymbol{0})}$ for the target domain.

$$\boldsymbol{U}_s^{(\boldsymbol{0})} = P\left(Z_{u_s}\middle|u_s\right), \boldsymbol{V}_s^{(\boldsymbol{0})} = P\left(Z_{v_s}\middle|v_s\right) \quad (4)$$

$$\boldsymbol{U}_t^{(\boldsymbol{0})} = P\left(Z_{u_t}\middle|u_t\right), \boldsymbol{V}_t^{(\boldsymbol{0})} = P\left(Z_{v_t}\middle|v_t\right) \quad (5)$$

where $P(Z_u|u) = \frac{P(u|Z_u)P(Z_u)}{\sum_{Z_u} P(u|Z_u)P(Z_u)}$ and $P(Z_v|v) = \frac{P(v|Z_v)P(Z_v)}{\sum_{Z_v} P(v|Z_v)P(Z_v)}$. Five parameters $P(u|Z_u)$, $P(v|Z_v)$, $P(r|Z_u, Z_v)$, $P(Z_u)$ and $P(Z_v)$ are learnt from the FMM (for details, see [22]).

4.2.2 Step 2: Domain adaptation of the user and item groups

This step ensures information consistency between the user/item group membership matrixes of two domains. The original user group membership matrixes $\boldsymbol{U}_s^{(0)}, \boldsymbol{U}_t^{(0)}$ and item group membership matrixes $\boldsymbol{V}_s^{(0)}, \boldsymbol{V}_t^{(0)}$ from the source and target domains are used as the starting point.

In one domain (say, the source domain), each column $\boldsymbol{U}_{s\ *j}^{(0)}$ represents the memberships of all users in a user group $\boldsymbol{j}$. Thus, it is reasonable to use the marginal probability distribution of column $\boldsymbol{U}_{s\ *j}^{(0)}$ to represent the characteristics of the user group information from user group $\boldsymbol{j}$**.** This is also applied to the other three matrixes $\boldsymbol{V}_s^{(0)}, \boldsymbol{U}_t^{(0)}, \boldsymbol{V}_t^{(0)}$. The disparity of the marginal probability distributions of user/item group membership matrixes in both domains is used to measure the divergence of the user/item group information. If the marginal probability distributions of the memberships of the two user/item groups are the same, these two user/item groups are regarded as having the same characteristics and the same physical meanings – information in the two user/item groups is consistent. This provides a method to measure the similarity between latent user/item groups in both domains. According to the basic assumption of recommender systems, i.e., "similar users like similar items", the preferences of similar user groups to similar item groups can be shared. Therefore, if the user/item group information of two domains is consistent, this group-level knowledge can be shared by both domains. The following formal definition of consistent user/item information and consistent knowledge determines which knowledge is transferrable.

Definition 2 (Information-consistent Tri-factorization). Given a source rating matrix $\boldsymbol{X}_s \in \mathbb{R}^{M_s \times N_s}$ and a target rating matrix, $\boldsymbol{X}_t \in \mathbb{R}^{M_t \times N_t}$, $\boldsymbol{X}_s$ and $\boldsymbol{X}_t$ can be factorized based on nonnegative tri-factorization:

$$\boldsymbol{X}_s = \boldsymbol{U}_s^{(0)} \boldsymbol{S}_s^{(0)} \left(\boldsymbol{V}_s^{(0)}\right)^T \qquad (6)$$

$$\boldsymbol{X}_t = \boldsymbol{U}_t^{(0)} \boldsymbol{S}_t^{(0)} \left(\boldsymbol{V}_t^{(0)}\right)^T \qquad (7)$$

If both tri-factorizations satisfy the following equations, then they are information-consistent tri-factorizations.

$$P\left(\boldsymbol{U}_s^{(\mathbf{0})}\right) = P\left(\boldsymbol{U}_t^{(\mathbf{0})}\right) \quad (8)$$

$$P\left(\boldsymbol{V}_s^{(\mathbf{0})}\right) = P\left(\boldsymbol{V}_t^{(\mathbf{0})}\right) \quad (9)$$

where $P\left(\boldsymbol{U}_s^{(\mathbf{0})}\right)$ and $P\left(\boldsymbol{V}_s^{(\mathbf{0})}\right)$ represent the marginal probability distributions of $\boldsymbol{U}_s^{(\mathbf{0})}$ and $\boldsymbol{V}_s^{(\mathbf{0})}$, respectively. We say that the user group information from $\boldsymbol{U}_s^{(\mathbf{0})}$ and $\boldsymbol{U}_t^{(\mathbf{0})}$ is consistent, and the item group information from $\boldsymbol{V}_s^{(\mathbf{0})}$ and $\boldsymbol{V}_t^{(\mathbf{0})}$ is consistent. That is, the user/item groups from source and target domains are consistent. $\boldsymbol{S}_s^{(\mathbf{0})}$ is the "consistent knowledge" of the two matrixes $\boldsymbol{X}_s$ and $\boldsymbol{X}_t$.

According to this definition, if the marginal probability distributions of user/item groups from source and target domains are the same, the group-level knowledge matrix can be shared, so that the consistent knowledge $\boldsymbol{S}_s^{(\mathbf{0})}$ can be directly used for the target rating matrix (let $\boldsymbol{S}_t^{(\mathbf{0})} = \boldsymbol{S}_s^{(\mathbf{0})}$). If the marginal probability distributions of the user/item group membership matrixes in both domains are not the same, we need to find other tri-factorization results that satisfy the conditions in Definition 2. Looking for a solution by trying different kinds of existing matrix factorization techniques is unattainable and time-consuming. Instead, we seek the solution by aligning consistent latent user groups and item groups through domain adaptation techniques. By adjusting the marginal probability distributions of user and item groups from the source and target domains comparatively, the similarities between the latent user and item groups are maximized. Consistent knowledge can then be extracted from the source rating matrix which can be directly used to help predict ratings in the target rating matrix.

To align consistent latent user and item groups, we need to find a projection to adjust the user/item group information of both rating matrixes so that the following equations are achieved:

$$P\left(\boldsymbol{\Psi}_s\left(\boldsymbol{U}_s^{(\mathbf{0})}, \boldsymbol{U}_t^{(\mathbf{0})}\right)\right) = P\left(\boldsymbol{\Psi}_t\left(\boldsymbol{U}_s^{(\mathbf{0})}, \boldsymbol{U}_t^{(\mathbf{0})}\right)\right) \quad (10)$$

$$P\left(\boldsymbol{\Phi}_s\left(\boldsymbol{V}_s^{(\mathbf{0})}, \boldsymbol{V}_t^{(\mathbf{0})}\right)\right) = P\left(\boldsymbol{\Phi}_t\left(\boldsymbol{V}_s^{(\mathbf{0})}, \boldsymbol{V}_t^{(\mathbf{0})}\right)\right) \quad (11)$$

It is apparent that $\boldsymbol{\Psi}_s$, $\boldsymbol{\Psi}_t$, $\boldsymbol{\Phi}_s$ and $\boldsymbol{\Phi}_t$ are the keys to ensuring that the latent groups remain consistent in both domains. We need to find maps that can force different distributions to become the same after mapping. A geodesic flow kernel (GFK) [23] is a domain adaptation strategy for learning robust features that is flexible against mismatch across domains and can be used to find a space for data in two domains to project into, so that the data distributions of the two domains in the projected space are similar. After projecting a GFK, a new representation is learned that satisfies the condition in Definition 2. Thus, we use a GFK to map $\boldsymbol{U}_s^{(0)}, \boldsymbol{U}_t^{(0)}, \boldsymbol{V}_s^{(0)}$ and $\boldsymbol{V}_t^{(0)}$ to $\boldsymbol{U}_s^{(1)}, \boldsymbol{U}_t^{(1)}, \boldsymbol{V}_s^{(1)}$ and $\boldsymbol{V}_t^{(1)}$. Based on the details of GFK, $\boldsymbol{\Psi}_s$, $\boldsymbol{\Psi}_t$, $\boldsymbol{\Phi}_s$, and $\boldsymbol{\Phi}_t$ can be written as follows:

$$\boldsymbol{\Psi}_s\left(\boldsymbol{U}_s^{(0)}, \boldsymbol{U}_t^{(0)}\right) = \boldsymbol{\Psi}_G\left(\boldsymbol{U}_s^{(0)}, \boldsymbol{U}_t^{(0)}\right) \times \boldsymbol{f}_{zs}\left(\boldsymbol{U}_s^{(0)}\right) \tag{12}$$

$$\boldsymbol{\Psi}_t\left(\boldsymbol{U}_s^{(0)}, \boldsymbol{U}_t^{(0)}\right) = \boldsymbol{\Psi}_G\left(\boldsymbol{U}_s^{(0)}, \boldsymbol{U}_t^{(0)}\right) \times \boldsymbol{f}_{zs}\left(\boldsymbol{U}_t^{(0)}\right) \tag{13}$$

$$\boldsymbol{\Phi}_s\left(\boldsymbol{V}_s^{(0)}, \boldsymbol{V}_t^{(0)}\right) = \boldsymbol{\Phi}_G\left(\boldsymbol{V}_s^{(0)}, \boldsymbol{V}_t^{(0)}\right) \times \boldsymbol{f}_{zs}\left(\boldsymbol{V}_s^{(0)}\right) \tag{14}$$

$$\boldsymbol{\Phi}_t\left(\boldsymbol{V}_s^{(0)}, \boldsymbol{V}_t^{(0)}\right) = \boldsymbol{\Phi}_G\left(\boldsymbol{V}_s^{(0)}, \boldsymbol{V}_t^{(0)}\right) \times \boldsymbol{f}_{zs}\left(\boldsymbol{V}_t^{(0)}\right) \tag{15}$$

where $\boldsymbol{\Psi}_G\left(\boldsymbol{U}_s^{(0)}, \boldsymbol{U}_t^{(0)}\right)$ and $\boldsymbol{\Phi}_G\left(\boldsymbol{V}_s^{(0)}, \boldsymbol{V}_t^{(0)}\right)$ are the operators of the GFK method and $\boldsymbol{f}_{zs}(\cdot)$ is the function of Z-score. More details on $\boldsymbol{\Psi}_G\left(\boldsymbol{U}_s^{(0)}, \boldsymbol{U}_t^{(0)}\right)$ and $\boldsymbol{\Phi}_G\left(\boldsymbol{V}_s^{(0)}, \boldsymbol{V}_t^{(0)}\right)$ can be found in Appendix A.

Then, the adapted latent user groups of the two rating matrixes can be obtained, which are expressed as

$$\boldsymbol{U}_s^{(1)} = \boldsymbol{\Psi}_s\left(\boldsymbol{U}_s^{(0)}, \boldsymbol{U}_t^{(0)}\right) \tag{16}$$

$$\boldsymbol{U}_t^{(1)} = \boldsymbol{\Psi}_t\left(\boldsymbol{U}_s^{(0)}, \boldsymbol{U}_t^{(0)}\right) \tag{17}$$

The same goes for the item groups: $\boldsymbol{V}_s^{(1)} = \boldsymbol{\Phi}_s\left(\boldsymbol{V}_s^{(0)}, \boldsymbol{V}_t^{(0)}\right)$, $\boldsymbol{V}_t^{(1)} = \boldsymbol{\Phi}_t\left(\boldsymbol{V}_s^{(0)}, \boldsymbol{V}_t^{(0)}\right)$. $\boldsymbol{U}_s^{(1)}, \boldsymbol{U}_t^{(1)}$ are user group membership matrixes unified to the same domain-invariant feature space for the source and target domains, while $\boldsymbol{V}_s^{(1)}$ and $\boldsymbol{V}_t^{(1)}$ are unified item group membership matrixes.

Here, an example best illustrates the domain adaptation process of the user and item group information. Consider a source domain and a target domain that both have 1000 non-overlapped users. In each domain, the users are clustered into six user groups, with inconsistent user group information between the source and target domains. The probability distributions of the first user group (the first column of $\boldsymbol{U}_s^{(0)}$ and $\boldsymbol{U}_t^{(0)}$) is shown in Fig. 3 (a); each is quite different. To force consistency, the information for every user group in each domain is adjusted, after which the user group information of the adapted matrixes $\boldsymbol{U}_s^{(1)}$ and $\boldsymbol{U}_t^{(1)}$ is almost the same, as shown in Fig. 3 (b).

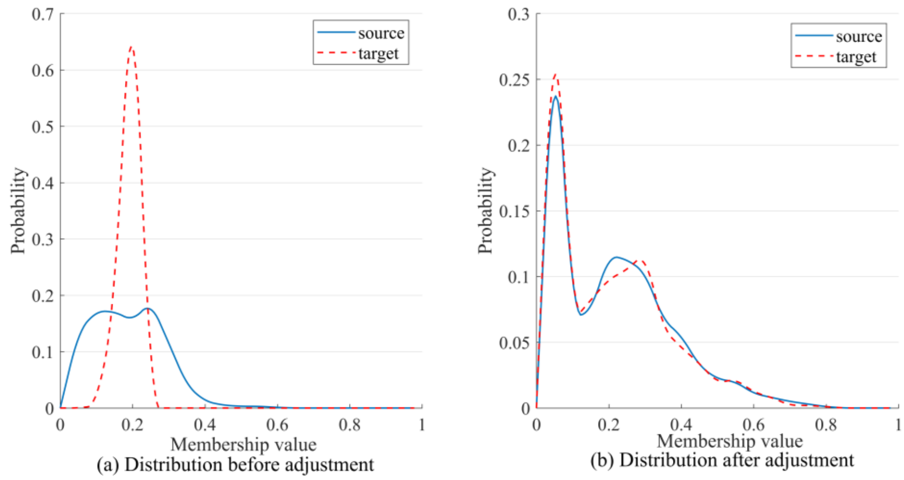


Fig. 3. An example of user group information adjustment in two domains

Note: (a) Marginal probability distribution of the first column in $\boldsymbol{U}_s^{(0)}$ and $\boldsymbol{U}_t^{(0)}$, (b) Marginal probability distribution of the first column in $\boldsymbol{U}_s^{(1)}$ and $\boldsymbol{U}_t^{(1)}$.

4.2.3 Step 3: Consistent knowledge extraction

After the domain adaptation, $\boldsymbol{U}_s^{(1)}$, $\boldsymbol{U}_t^{(1)}$ are consistent, and $\boldsymbol{V}_s^{(1)}$, $\boldsymbol{V}_t^{(1)}$ are consistent. Once we have obtained consistent group representations that are meaningful across both rating matrixes, the model trained on the source rating matrix and the target rating matrix can be brought together. On this basis, the recommender systems learned from the source and target domains will share the same group-level knowledge matrix $\boldsymbol{S}$.

Consistent knowledge $\boldsymbol{S}$ is obtained by maximizing the approximation of the available data in both the source rating matrix and the target rating matrix by approximating $\boldsymbol{X}_s \approx \boldsymbol{U}_s^{(1)}\boldsymbol{S}\left(\boldsymbol{V}_s^{(1)}\right)^T$ together with $\boldsymbol{X}_t \approx \boldsymbol{U}_t^{(1)}\boldsymbol{S}\left(\boldsymbol{V}_t^{(1)}\right)^T$. To qualify the approximation, one useful and simple measure is to use a Frobenius norm between the original rating matrix and the approximation. We have the following cost function:

$$J_s(\boldsymbol{S}) = \frac{1}{M_s N_s}\left\|\boldsymbol{W}_s \odot \left(\boldsymbol{X}_s - \boldsymbol{U}_s^{(1)}\boldsymbol{S}\left(\boldsymbol{V}_s^{(1)}\right)^T\right)\right\|_F + \frac{1}{M_t N_t}\left\|\boldsymbol{W}_t \odot \left(\boldsymbol{X}_t - \boldsymbol{U}_t^{(1)}\boldsymbol{S}\left(\boldsymbol{V}_t^{(1)}\right)^T\right)\right\|_F + \frac{1}{2KL}\lambda\|\boldsymbol{S}\|_F \quad (18)$$

where $\boldsymbol{W}_s$ is a binary weighting matrix for $\boldsymbol{X}_s$, $[\boldsymbol{W}_s]_{ij} = 1$, if $[\boldsymbol{X}_s]_{ij} \neq 0$ and $[\boldsymbol{W}_s]_{ij} = 0$, otherwise. The same applies to $\boldsymbol{W}_t$ for $\boldsymbol{X}_t$. $\odot$ is an entry-wise product, λ is the parameter for regularization.

Since the physical meaning of $\boldsymbol{S}$ is the preference that the user groups give to the item groups, it should be in range of (0,5]. Regularization to constrain the range of $\boldsymbol{S}$ is added to the cost function. Finally, consistent knowledge is learned through the following optimization problem:

$$\min J_s(\boldsymbol{S})$$

$$s.t.\ \boldsymbol{S} > 0$$

Gradient descent is a general algorithm for optimization, which leads to the update rule: $s_{ab} \leftarrow s_{ab} + \eta_{ab}\frac{\partial J_s}{\partial s_{ab}}$. For this problem, we need to constrain the non-negativity of S. The partial derivative of the cost function has a special form, so we can use tricks to set the learning rate $\eta_{ab} = {(\boldsymbol{S})_{ab}}\Big/{\left(A + B + \frac{\lambda S}{2KL}\right)_{ab}}$ to guarantee that $\boldsymbol{S}$ is nonnegative, where $A = \frac{1}{M_s N_s}\left(\boldsymbol{U}_s^{(1)}\right)^T\left(\boldsymbol{W}_s \odot \left(\boldsymbol{U}_s^{(1)}\boldsymbol{S}\left(\boldsymbol{V}_s^{(1)}\right)^T\right)\right)\boldsymbol{V}_s^{(1)}$, $B = \frac{1}{M_t N_t}\left(\boldsymbol{U}_t^{(1)}\right)^T\left(\boldsymbol{W}_t \odot \left(\boldsymbol{U}_t^{(1)}\boldsymbol{S}\left(\boldsymbol{V}_t^{(1)}\right)^T\right)\right)\boldsymbol{V}_t^{(1)}$. The objective function is non-increasing under the following update rule:

$$s_{ab} \leftarrow s_{ab}\frac{\left(\frac{1}{M_s N_s}\left(\boldsymbol{U}_s^{(1)}\right)^T \boldsymbol{X}_s \boldsymbol{V}_s^{(1)} + \frac{1}{M_t N_t}\left(\boldsymbol{U}_t^{(1)}\right)^T \boldsymbol{X}_t \boldsymbol{V}_t^{(1)}\right)_{ab}}{\left(A + B + \frac{\lambda S}{2KL}\right)_{ab}} \tag{19}$$

The learning process is summarized in Algorithm 1.

Algorithm 1: Consistent Knowledge Extraction

Input: $\boldsymbol{X}_s$, the source rating matrix
$\boldsymbol{X}_t$, the target rating matrix
$\boldsymbol{U}_s^{(1)}, \boldsymbol{V}_s^{(1)}$, user and item membership matrix of source domain
$\boldsymbol{U}_t^{(1)}, \boldsymbol{V}_t^{(1)}$, user and item membership matrix of target domain
($\boldsymbol{U}_s^{(1)}, \boldsymbol{V}_s^{(1)}, \boldsymbol{U}_t^{(1)}, \boldsymbol{V}_t^{(1)}$ are obtained from GFK algorithm)

Output: $\boldsymbol{S}$, the consistent knowledge

```
INITILIZE S ∈ ℝ^{K×L}, J_s^(min) ← 0, J_s ← 0
WHILE J_s = 0 OR J_s − J_s^(min) > ε DO
    FOR each element s_ab in S DO
        UPDATE s_ab as in equation (19)
    ENDFOR
    UPDATE J_s as in equation (18)
    IF J_s^(min) > J_s
        J_s^(min) = J_s
    ENDIF
ENDWHILE
RETURN S
```

4.2.4 Step 4: Group representation regulation

The domain adaptation technique GFK is designed for unsupervised transfer learning where no label is available in the target domain. In this problem setting, some domain-specific characteristics are embedded in the small amount of available data in the target rating matrix. To reveal these idiosyncrasies of the target domain, we amend the group representations of the target rating matrix to make the model fit better to the task in target rating matrix. It is imperative that we find maps $f_u : \boldsymbol{U}_t^{(1)} \mapsto \mathbb{R}^{M_t \times K}$ and $f_v : \boldsymbol{V}_t^{(1)} \mapsto \mathbb{R}^{N_t \times L}$ to make $\boldsymbol{U}_t^{(1)}$ and $\boldsymbol{V}_t^{(1)}$ more suitable for the target rating matrix. At the same time, the adjustment should not impair the consistency of user groups and item groups between two domains. According to Definition 2, f_u and f_v should satisfy the following equation:

$$P\left(\boldsymbol{S} \,\middle|\, f_u\left(\boldsymbol{U}_t^{(1)}\right), f_v\left(\boldsymbol{V}_t^{(1)}\right)\right) = P\left(\boldsymbol{S} \,\middle|\, \boldsymbol{U}_t^{(1)}, \boldsymbol{V}_t^{(1)}\right) \tag{20}$$

Equation (20) ensures that the probability of each element in $\boldsymbol{S}$ will not change after mapping $\boldsymbol{U}_t^{(1)}$ and $\boldsymbol{V}_t^{(1)}$ using f_u and f_v. Here, we choose $f_u\left(\boldsymbol{U}_t^{(1)}\right) = \boldsymbol{U}_t^{(1)}\boldsymbol{u}$ and $f_v\left(\boldsymbol{V}_t^{(1)}\right) = \boldsymbol{V}_t^{(1)}\boldsymbol{v}$, where $\boldsymbol{u} \geq \boldsymbol{0}$ and $\boldsymbol{v} \geq \boldsymbol{0}$. These two maps satisfy equation (20). For further details of why f_u and f_v are chosen like this, see Appendix B.

Learning f_u and f_v is an optimization problem. The cost function is:

$$J_r(\boldsymbol{u}, \boldsymbol{v}) = \left\| \boldsymbol{W}_t \odot \left(\boldsymbol{X}_t - \boldsymbol{U}_t^{(1)} \boldsymbol{u} \boldsymbol{S} \left(\boldsymbol{V}_t^{(1)} \boldsymbol{v} \right)^T \right) \right\|_F \tag{21}$$

The tuning factors can be learned through optimizing

$$\min J_r(\boldsymbol{u}, \boldsymbol{v})$$

$$s.t.\ \boldsymbol{u} \geq 0, \boldsymbol{v} \geq 0$$

Similarly, the cost function is non-increasing under the following update rules:

$$u_{ab} \leftarrow u_{ab} \frac{\left(\left(\boldsymbol{U}_t^{(1)}\right)^T \boldsymbol{X}_t \boldsymbol{V}_t^{(1)} \boldsymbol{v} \boldsymbol{S}^T\right)_{ab}}{\left(\left(\boldsymbol{U}_t^{(1)}\right)^T \left(\boldsymbol{W}_t \odot \left(\boldsymbol{U}_t^{(1)} \boldsymbol{u} \boldsymbol{S} \left(\boldsymbol{V}_t^{(1)} \boldsymbol{v}\right)^T\right)\right) \boldsymbol{V}_t^{(1)} \boldsymbol{v} \boldsymbol{S}^T\right)_{ab}} \tag{22}$$

$$v_{cd} \leftarrow v_{cd} \frac{\left(\left(\boldsymbol{V}_t^{(1)}\right)^T \boldsymbol{X}_t^T \boldsymbol{U}_t^{(1)} \boldsymbol{v} \boldsymbol{S}\right)_{cd}}{\left(\left(\boldsymbol{V}_t^{(1)}\right)^T \left(\boldsymbol{W}_t^T \odot \left(\boldsymbol{V}_t^{(1)} \boldsymbol{v} \boldsymbol{S}^T \boldsymbol{u}^T \left(\boldsymbol{U}_t^{(1)}\right)^T\right)\right) \boldsymbol{U}_t^{(1)} \boldsymbol{v} \boldsymbol{S}\right)_{cd}} \tag{23}$$

Finally, the optimization problem is solved by alternatively estimating $\boldsymbol{u}, \boldsymbol{v}$. How $\boldsymbol{u}, \boldsymbol{v}$ is learned is summarized in Algorithm 2.

Algorithm 2: Group Representation Regulation

Input: $\boldsymbol{X}_t$, the target rating matrix
$\boldsymbol{S}$, the consistent knowledge
$\boldsymbol{U}_t^{(1)}, \boldsymbol{V}_t^{(1)}$, user and item membership matrix of target domain
($\boldsymbol{U}_t^{(1)}, \boldsymbol{V}_t^{(1)}$ are obtained from GFK algorithm)
Output: $\boldsymbol{u}$, user tuning factor
$\boldsymbol{v}$, item tuning factor

INITIALIZE $\boldsymbol{u} \in \mathbb{R}^{K\times K}$, $\boldsymbol{v} \in \mathbb{R}^{L\times L}$, $J_r^{(min)} \leftarrow 0$, $J_r \leftarrow 0$
WHILE $J_r = 0$ **OR** $J_r - J_r^{(min)} > \varepsilon$ **DO**
  FOR each element $\boldsymbol{u}_{ab}$ in $\boldsymbol{u}$ **DO**
    UPDATE $\boldsymbol{u}_{ab}$ as in equation (22)
  ENDFOR
  FOR each element $\boldsymbol{v}_{cd}$ in $\boldsymbol{v}$ **DO**
    UPDATE $\boldsymbol{v}_{cd}$ as in equation (23)
  ENDFOR
  UPDATE $J_r^{(i)}$ as in equation (21)
  IF $J_r^{(min)} > J_r$
    $J_r^{(min)} = J_r$
  ENDIF
ENDWHILE
RETURN $\boldsymbol{u}, \boldsymbol{v}$

4.2.5 Step 5: Recommendation in target domain

The recommendation in target domain is given by equation (24).

$$\begin{cases} \widehat{\boldsymbol{X}}_t = \left(\boldsymbol{U}_t^{(1)}\boldsymbol{u}\right)\boldsymbol{S}\left(\boldsymbol{V}_t^{(1)}\boldsymbol{v}\right)^T \\ \boldsymbol{U}_t^{(1)} = \boldsymbol{\Psi}_G\left(\boldsymbol{U}_s^{(0)}, \boldsymbol{U}_t^{(0)}\right) \times \boldsymbol{f}_{zs}\left(\boldsymbol{U}_t^{(0)}\right) \\ \boldsymbol{V}_t^{(1)} = \boldsymbol{\Phi}_G\left(\boldsymbol{V}_s^{(0)}, \boldsymbol{V}_t^{(0)}\right) \times \boldsymbol{f}_{zs}\left(\boldsymbol{V}_t^{(0)}\right) \end{cases} \tag{24}$$

where $\widehat{\boldsymbol{X}}_t$ is the reconstructed user-item rating matrix for prediction, $\boldsymbol{u}, \boldsymbol{v}$ are user and item tuning factors for target domain, $\boldsymbol{S}$ is the consistent knowledge, $\boldsymbol{U}_s^{(0)}, \boldsymbol{U}_t^{(0)}$ are user group membership matrixes, and $\boldsymbol{V}_s^{(0)}, \boldsymbol{V}_t^{(0)}$ are item group membership matrixes for the source domain and the target domain before domain adaptation. $\boldsymbol{U}_t^{(1)}, \boldsymbol{V}_t^{(1)}$ are user and item group membership matrixes for the target domain after domain adaptation. $\boldsymbol{\Psi}_G(\cdot)$ and $\boldsymbol{\Phi}_G(\cdot)$ are GFK operators to map group membership matrixes to a domain-invariant feature space, and $\boldsymbol{f}_{zs}(\cdot)$ is the Z-score function.

4.3 Architecture of a Cross-domain Recommender System

In the proposed CIT method, group-level knowledge from a source domain and a target domain can be combined and augmented compared with what can be acquired independently from only the target domain. In this section, we introduce how to use the proposed CIT method when developing a recommender system to support decision making for businesses and individual customers.

A conceptual framework for a cross-domain recommender system that applies the proposed method is shown in Fig. 4. When businesses launch a new product or service, a sufficient amount of data has not always been collected to populate the target domain. It is often easier to acquire data from another mature service – the source domain. Accordingly, a cross-domain recommendation engine can be built, based on our method, to provide better predictions of a user's preferences for items. This assists decision making for both businesses and individual customers.

For businesses, the CDRS could be used to support product development and marketing decisions. For example, businesses could predict user preferences for more accurate cross-selling or identify potential user groups to market specific products to. They could also develop product bundles based on user preference prediction. For individual customers, our proposed CDRS could be used to facilitate targeted product searches. By ranking products according to predicted preference, customers may be able to locate the most desirable products more quickly and effectively.

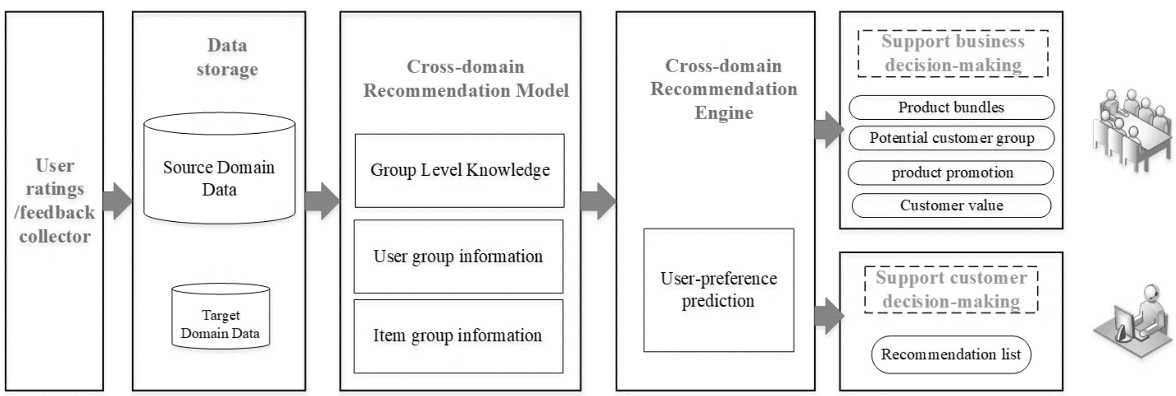


Fig. 4. Conceptual framework of a cross-domain recommender system

5. Experiments and Analysis

Our empirical experiments are presented in this section. First, the datasets and evaluation metrics are introduced, followed by the experimental settings and the baseline methods. The results of the experiments are presented along with an analysis of the parameters.

5.1 Dataset and Evaluation Metrics

In testing the CIT method, it was important to choose data from different but similar domains. Previous research has considered movies, books, and music as appropriate categories for CDRS experiment tests. For a fair comparison, we have chosen the same categories and many of the same datasets for our experiments.

Our tests comprise nine cross-domain recommendation tasks, including movie-to-movie and book-to-movie recommendations, common in prior research, as well as some new tasks extending to the music category that are less commonly tested. The baseline methods include three non-transfer methods and two cross-domain methods. Five real-world datasets were used: Movielens 20M [1], Netflix [2], LibraryThing [3], Amazon Book [4] and YahooMusic [5]. Each is publicly available and has been used to test recommender systems in a variety of scenarios for recommender systems in single domain. But tests on these dataset in this novel cross-domain setting are lacking. The statistical information for these datasets is provided in Table II.

Table II Statistical information on the original datasets

	Movielens 20M	Netflix	Library Thing	Amazon Book	Yahoo Music_1	Yahoo Music_2
#user	138493	480189	7279	8026324	200000	200000
#item	26744	17770	37232	2330066	136736	136736
#rating	20000263	100480507	749401	22507155	78344627	78742463
sparsity	0.54%	1.18%	0.28%	0.0001%	0.29%	0.29%
range	0.5-5	1-5	0.5-5	1-5	1-5	1-5
average	3.5255	3.6043	3.8709	4.2958	3.1613	3.1634
STD	1.0520	1.0852	0.9387	1.1115	1.5991	1.6046

In the Amazon Book dataset, we found that more than 6 million among 8 million users gave all their reviewed items the same rating. This phenomenon is very uncommon and rarely happens in real-world. As such, it was determined that these users could provide no effective contribution to the construction of a recommender system and were removed. For the Movielens20M and LibraryThing datasets, we normalized the ratings to a range of {1,2,3,4,5}. Movielens20M, LibraryThing, and YahooMusic_1 were used as the source domain, while Netflix, AmazonBook, YahooMusic_2 were used as the target domain. Across all the datasets, 2000 items that had been rated more than 10 times were randomly chosen. We then filtered out the users who had given less than a total of 20 ratings. The next section describes how the users were chosen.

For the source domain data, we randomly selected 4000 users to be regular customers of the site. The sparsity ratio of source domain data was controlled at 2%. Two source domain datasets with different statistical properties were chosen to test the performance of different algorithms. For the target domain data,

[1] http://grouplens.org/datasets/movielens/20m/
[2] http://netflixprize.com/index.html
[3] https://www.librarything.com
[4] http://jmcauley.ucsd.edu/data/amazon/
[5] http://webscope.sandbox.yahoo.com./

we randomly selected 2000 users to be regular customers of the site, and another 2000 users to be new customers. In terms of regular customers, three sparsity ratios were used to compare different algorithms in different circumstances. For new users, five observed ratings were given, and the rest of the ratings were used for evaluation. In the end, the rating matrixes for both the source and target domains were all 4000 × 2000 matrixes. The details of the final datasets are summarized in Table III.

Mean absolute error (MAE) and root mean square error (RMSE) were used as the evaluation metrics:

$$MAE = \sum_{(u,v,r_{u,v}) \in Y} \frac{|\hat{r_{u,v}} - r_{u,v}|}{|Y|}$$

$$RMSE = \sqrt{\sum_{(u,v,r_{u,v}) \in Y} \frac{(\hat{r_{u,v}} - r_{u,v})^2}{|Y|}}$$

where Y is the test set, and $|Y|$ is the number of test ratings.

Table III Description of data subsets in three categories

Data_type	Data_name	Data_source	Domain	Sparsity	Average
Movie	movie_s1	Movielens20M	source	2.00%	3.66
	movie_s2	Movielens20M	source	2.00%	2.63
	movie_t1	Netflix	target	0.50%	2.68
	movie_t2	Netflix	target	1.00%	2.67
	movie_t3	Netflix	target	1.50%	2.67
Book	book_s1	LibraryThing	source	2.00%	4.02
	book_s2	LibraryThing	source	2.00%	3.72
	book_t1	Amazon	target	0.50%	3.52
	book_t2	Amazon	target	0.75%	3.53
	book_t3	Amazon	target	0.94%	3.53
Music	music_s1	YahooMusic_1	source	2.00%	4.13
	music_s2	YahooMusic_1	source	2.00%	2.73
	music_t1	YahooMusic_2	target	0.50%	2.26
	music_t2	YahooMusic_2	target	1.00%	2.26
	music_t3	YahooMusic_2	target	1.50%	2.25

5.2 Experimental Settings and Baselines

Three non-transfer learning methods and two cross-domain methods were chosen as comparisons for the proposed method. The non-transfer learning methods were: Pearson's correlation coefficient (PCC) [2], FMM [22] and SVD [3]. The cross-domain methods were: CBT [18] and RMGM [19]. PCC uses user-based CF, and the number of neighborhoods was set at 50. For SVD, the latent feature number was fixed at 40, the regularization factor was set to 0.015, and the learning rate was set to 0.003. For FMM, CBT, and RMGM, the user group number and item group number were both set to 40. For the proposed method, CIT, the user group number and the item group number were both set to 40, and the regularization factor was set to 0.5.

Further analysis of the parameters is provided in Sub-section 5.4.

For each target domain, three configurations of sparsity were settled; thus, nine cross-domain recommendation tasks each under three sparsity ratios were conducted for comparison between the baselines and the proposed method. Since the algorithms (except for PCC) need to initialize the factorized matrix randomly, we ran 20 random initializations and report the averaged results and standard deviations.

5.3 Results

Comparison results are given in Table IV, V and VI. The proposed method, CIT, had the lowest MAE and RMSE among all the six methods in most of the cross-domain recommendation tasks. Compared with the non-transfer learning methods, we find that our method is more effective at extracting knowledge from the source domain to apply in the target domain. This is especially significant when the statistical properties of the source rating matrix are different from those in the target rating matrix. This indicates that our method gains its benefits by keeping the user and item group information in both domains consistent. The CIT method is able to extract knowledge even when the statistical properties of the source rating matrix diverge from the target rating matrix, while CBT and RMGM may need some restricted conditions of source data.

Comparing the six methods and given the results of all nine tasks with different sparsity ratios, we can make the following observations:

(1) For non-transfer learning methods, the FMM method shows superior performance compared to the memory-based method PCC and the famous matrix factorization method SVD from the Netflix competition. PCC and SVD are not very good at handling the cold-start problem. When the number of available ratings for users in target domain is limited, they fail to give good recommendations.

(2) CBT is not stable and positive transfer is not guaranteed. When the statistical properties of the source rating matrix is similar to that of the target rating matrix (say movie_s2 to movie_t1/2/3), CBT is better than the non-transfer baselines. Since CBT fills the source rating matrix with the users'

Table IV Prediction performance on a movie target domain

method	source data	MAE			RMSE		
		Sparsity			Sparsity		
		0.50%	1.00%	1.50%	0.50%	1.00%	1.50%

PCC	-	1.2609	1.2710	1.1981	1.5671	1.5789	1.4839
FMM	-	1.0164±0.0027	1.0069±0.0033	1.0029±0.0028	1.2283±0.0036	1.2143±0.0045	1.2064±0.0037
SVD	-	1.0230±0.0013	1.0227±0.0012	1.0391±0.0077	1.2372±0.0015	1.2382±0.0012	1.2544±0.0096
CBT	movie_s1	1.2868±0.0034	1.2845±0.0072	1.2836±0.0038	1.5318±0.0043	1.5290±0.0092	1.5277±0.0048
	movie_s2	1.0205±0.0007	1.0194±0.0016	1.0192±0.0010	**1.1964***±0.0003	1.1962±0.0008	1.1958±0.0004
	book_s1	1.4493±0.0075	1.4477±0.0071	1.4441±0.0066	1.7627±0.0114	1.7604±0.0107	1.7551±0.0100
	book_s2	1.3272±0.0118	1.3248±0.0071	1.3253±0.0104	1.5871±0.0159	1.5839±0.0093	1.5849±0.0134
	music_s1	1.4917±0.0189	1.4935±0.0164	1.4923±0.0146	1.8115±0.0187	1.8131±0.0155	1.8112±0.0141
	music_s2	1.0144±0.0023	1.0134±0.0019	1.0141±0.0020	1.2027±0.0027	1.2032±0.0020	1.2018±0.0030
RMGM	movie_s1	1.0347±0.0065	1.0252±0.0050	1.0214±0.0034	1.2515±0.0079	1.2402±0.0067	1.2345±0.0047
	movie_s2	1.0038±0.0022	0.9994±0.0025	0.9977±0.0028	1.2104±0.0031	1.2025±0.0033	1.1992±0.0038
	book_s1	1.0464±0.0048	1.0369±0.0046	1.0309±0.0052	1.2711±0.0060	1.2583±0.0064	1.2501±0.0069
	book_s2	1.0396±0.0033	1.0326±0.0038	1.0261±0.0043	1.2616±0.0043	1.2523±0.0048	1.2433±0.0057
	music_s1	1.0498±0.0055	1.0387±0.0056	1.0299±0.0058	1.2734±0.0078	1.2595±0.0079	1.2463±0.0076
	music_s2	1.0591±0.0061	1.0512±0.0039	1.0489±0.0046	1.2813±0.0076	1.2711±0.0052	1.2655±0.0058
CIT	movie_s1	**1.0002***±0.0025	**0.9906***±0.0027	**0.9888***±0.0025	**1.1846***±0.0027	**1.1881***±0.0034	**1.1846***±0.0027
	movie_s2	**0.9995***±0.0028	**0.9911***±0.0023	**0.9873***±0.0022	1.1987±0.0040	**1.1887***±0.0029	**1.1828***±0.0034
	book_s1	**0.9992***±0.0022	**0.9908***±0.0019	**0.9886***±0.0023	**1.1978***±0.0034	**1.1882***±0.0026	**1.1843***±0.0028
	book_s2	**0.9993***±0.0032	**0.9907***±0.0022	**0.9889***±0.0022	**1.1985***±0.0041	**1.1886***±0.0026	**1.1853***±0.0032
	music_s1	**0.9996***±0.0033	**0.9914***±0.0022	**0.9883***±0.0018	**1.1985***±0.0045	**1.1885***±0.0029	**1.1839***±0.0025
	music_s2	**1.0004***±0.0025	**0.9931***±0.0021	**0.9892***±0.0023	**1.1997***±0.0031	**1.1886***±0.0028	**1.1848***±0.0033

Table V Prediction performance on a book target domain

method	source data	MAE			RMSE		
		Sparsity			Sparsity		
		0.50%	0.75%	0.94%	0.50%	0.75%	0.94%
PCC	-	1.2625	1.2654	1.2340	1.5737	1.5739	1.5305
FMM	-	1.0645±0.0028	1.0256±0.0022	**1.0211***±0.0029	1.3152±0.0035	1.2645±0.0033	1.2582±0.0045
SVD	-	1.0591±0.0025	1.0288±0.0021	1.1702±0.0034	1.3220±0.0028	1.2826±0.0026	1.5032±0.0047
CBT	movie_s1	1.0859±0.0009	1.0856±0.0005	1.0851±0.0006	1.3233±0.0037	1.3219±0.0023	1.3220±0.0024
	movie_s2	1.2345±0.0118	1.2334±0.0072	1.2287±0.0078	1.4271±0.0105	1.4260±0.0064	1.4215±0.0070
	book_s1	1.0896±0.0012	1.0893±0.0009	1.0895±0.0012	1.4330±0.0030	1.4310±0.0038	1.4312±0.0038
	book_s2	1.0813±0.0010	1.0814±0.0009	1.0809±0.0008	1.3569±0.0059	1.3547±0.0046	1.3525±0.0041
	music_s1	1.1128±0.0082	1.1127±0.0098	1.1105±0.0080	1.4616±0.0096	1.4618±0.0091	1.4598±0.0073
	music_s2	1.1881±0.0129	1.1906±0.0147	1.1935±0.0119	1.3895±0.0099	1.3910±0.0116	1.3930±0.0095
RMGM	movie_s1	1.0673±0.0046	1.0460±0.0051	1.0425±0.0047	1.3057±0.0066	1.2786±0.0065	1.2750±0.0070
	movie_s2	1.0594±0.0037	1.0329±0.0036	1.0277±0.0037	1.2933±0.0039	1.2614±0.0043	1.2558±0.0036
	book_s1	1.0726±0.0041	1.0440±0.0039	1.0409±0.0046	1.3339±0.0052	1.2962±0.0050	1.2914±0.0057
	book_s2	1.0649±0.0037	1.0424±0.0037	1.0376±0.0029	1.3172±0.0049	1.2883±0.0050	1.2807±0.0036
	music_s1	1.0817±0.0055	1.0588±0.0063	1.0539±0.0053	1.3432±0.0074	1.3143±0.0103	1.3082±0.0081
	music_s2	1.1028±0.0076	1.0832±0.0072	1.0713±0.0068	1.3430±0.0094	1.3196±0.0086	1.3092±0.0084
CIT	movie_s1	**1.0464***±0.0045	**1.0246**±0.0031	1.0243±0.0028	**1.2685***±0.0041	**1.2464***±0.0041	**1.2458***±0.0046
	movie_s2	**1.0456***±0.0036	**1.0249**±0.0032	1.0245±0.0024	**1.2688***±0.0040	**1.2474***±0.0035	**1.2458***±0.0022
	book_s1	**1.0465***±0.0031	1.0257±0.0028	1.0247±0.0030	**1.2705***±0.0041	**1.2468***±0.0040	**1.2458***±0.0039
	book_s2	**1.0474***±0.0045	**1.0254**±0.0026	1.0236±0.0031	**1.2707***±0.0050	**1.2476***±0.0034	**1.2448***±0.0043
	music_s1	**1.0467***±0.0040	**1.0249***±0.0024	1.0238±0.0030	**1.2711***±0.0047	**1.2465***±0.0039	**1.2442***±0.0033
	music_s2	**1.0457***±0.0030	1.0265±0.0030	1.0238±0.0032	**1.2690***±0.0030	**1.2482***±0.0036	**1.2456***±0.0028

Table VI Prediction performance on a music target domain

method	source data	MAE			RMSE		
		Sparsity			Sparsity		
		0.50%	1.00%	1.50%	0.50%	1.00%	1.500025

PCC	-	1.4403	1.3617	1.3262	1.8421	1.7080	1.6489
FMM	-	1.2619±0.0023	1.2460±0.0027	1.2448±0.0028	1.5009±0.0035	1.4754±0.0057	1.4685±0.0045
SVD	-	1.2675±0.0009	1.2603±0.0009	1.2566±0.0014	1.4972±0.0011	1.4916±0.0015	1.4876±0.0015
CBT	movie_s1	1.3776±0.0030	1.3759±0.0044	1.3764±0.0029	1.6168±0.0046	1.6136±0.0069	1.6149±0.0048
	movie_s2	1.2726±0.0021	1.2734±0.0025	1.2728±0.0024	1.4663±0.0017	1.4644±0.0021	1.4634±0.0021
	book_s1	1.4666±0.0038	1.4665±0.0038	1.4656±0.0064	1.7929±0.0076	1.7926±0.0076	1.7908±0.0127
	book_s2	1.3986±0.0045	1.3973±0.0050	1.4005±0.0035	1.6598±0.0071	1.6581±0.0085	1.6634±0.0068
	music_s1	1.4971±0.0131	1.4934±0.0102	1.5012±0.0049	1.8343±0.0139	1.8310±0.0102	1.8387±0.0050
	music_s2	1.2597±0.0059	1.2598±0.0054	1.2568±0.0051	1.4604±0.0036	1.4604±0.0035	1.4603±0.0041
RMGM	movie_s1	1.2699±0.0038	1.2576±0.0048	1.2539±0.0036	1.5099±0.0062	1.4952±0.0075	1.4897±0.0056
	movie_s2	1.2482±0.0023	1.2401±0.0027	1.2406±0.0029	1.4819±0.0051	1.4698±0.0048	1.4707±0.0051
	book_s1	1.2832±0.0049	1.2690±0.0030	1.2623±0.0054	1.5374±0.0069	1.5180±0.0058	1.5094±0.0088
	book_s2	1.2757±0.0037	1.2620±0.0047	1.2575±0.0042	1.5285±0.0065	1.5096±0.0076	1.5026±0.0070
	music_s1	1.2901±0.0051	1.2767±0.0063	1.2683±0.0083	1.5497±0.0097	1.5317±0.0095	1.5199±0.0117
	music_s2	1.2881±0.0037	1.2842±0.0035	1.2799±0.0057	1.5385±0.0066	1.5328±0.0079	1.5264±0.0069
CIT	movie_s1	**1.2450***±0.0021	**1.2375***±0.0019	**1.2344***±0.0015	**1.4516***±0.0030	**1.4451***±0.0029	**1.4400***±0.0026
	movie_s2	**1.2452***±0.0019	**1.2375***±0.0018	**1.2345***±0.0020	**1.4513***±0.0026	**1.4439***±0.0030	**1.4409***±0.0038
	book_s1	**1.2449***±0.0019	**1.2385***±0.0017	**1.2350***±0.0021	**1.4511***±0.0020	**1.4448***±0.0022	**1.4411***±0.0040
	book_s2	**1.2455***±0.0015	**1.2377***±0.0016	**1.2344***±0.0017	**1.4523***±0.0026	**1.4444***±0.0029	**1.4403***±0.0023
	music_s1	**1.2449***±0.0021	**1.2379***±0.0021	**1.2349***±0.0026	**1.4511***±0.0027	**1.4445***±0.0027	**1.4404***±0.0039
	music_s2	**1.2453***±0.0023	**1.2375***±0.0018	**1.2344***±0.0021	**1.4513***±0.0025	**1.4438***±0.0021	**1.4402***±0.0031

average ratings, the average is crucial to this method and gains more advantages on two datasets when their average ratings are close. However, many results, like movie_s1 to movie_t1/2/3, suggest that CBT grapples with negative transfer issues. Referring to the statistical properties in Table III, the performance of CBT is directly related to the average of ratings. When the average rating of the source rating matrix deviates from that of the target domain, the performance of CBT is greatly impaired.

(3) RMGM shows similar performance to CBT but is more stable. The rating matrixes from the source and target domains are diagonally joined in RMGM. It is necessary for the two matrixes to have similar statistical properties to extract common knowledge, but RMGM fails to note whether or not the two matrixes are similar. RMGM's results suggest that discrepancies in the average will disturb the extraction of common knowledge, thus weakening transfer learning. We can see that positive transfer cannot be assured without a similarity guarantee of the rating matrixes for the source and target domains.

(4) The proposed CIT performs better than all the other baseline methods in almost all tasks, whether or not the datasets are in the same category. CIT ensures a steady improvement compared to non-transfer learning methods. Unlike the other two cross-domain methods, CIT is also suitable for datasets with different statistical properties. The adaptation knowledge transfer in CIT ensures that

the knowledge extracted from the source rating matrix is suitable for assisting recommendation in the target domain.

(5) Negative transfer was always observed for CBT and RMGM when the average rating in the source domain was different from that of the target domain. This leads to a fundamental question in transfer learning: 'When to transfer?' This is an area seldom studied in CDRS. Instead of determining when to transfer, our proposed method reduces the difference between the source and target domains by preserving consistent user and item group information. In the scope of this paper, we did not see any negative transfer learning in our proposed method.

To confirm that the improvement of our CIT method over other methods was significant, we conducted a significance analysis on all pairs of experiments for each of the nine tasks in all three sparsity ratios using Friedman's test. Most of the resulting P-values were much smaller than the significance level α ($\alpha = 0.05$). Statistically significant results are marked with an asterisk (*) in Tables IV, V and VI. Only one result was not a statistically significant improvement – the book target domain with a sparsity of 1.50% compared to the FMM non-transfer method in terms of MAE. However, CIT's performance improvement in the same scenario was significant at a data sparsity of 0.50%, suggesting that cross-domain transfer may not be required as data richness in target domain increases.

To better understand the effectiveness of transfer learning on each individual task, we calculated the average MAEs and RMSEs for each cross-domain recommendation task. The results are presented in Tables VII and VIII. The results for the nine tasks show that the proposed CIT method achieves the best performance in terms of both MAE and RMSE of the six methods.

Fig. 5 compares the results for all the methods. Since the rating average is different between the source and target domains, the overall performance of cross-domain methods CBT and RMGM was not as good as the non-transfer learning method FMM. RMGM is relatively stable and is mostly better than SVD, while CBT fluctuates and is worse than most of the other methods. We can see that the overall performance in the music category is worse than that in the movie and book categories, indicating that the rating matrix in the music category has different characteristics; however, our proposed method was still able to extract useful knowledge to help increase the prediction accuracy.

Table VII Prediction result of average MAE

Task	non-transfer			cross-domain		
	PCC	FMM	SVD	CBT	RMGM	CIT
m2m				1.1523	1.0137	**0.9929**
b2m	1.2433	1.0087	1.0283	1.3864	1.0354	**0.9929**
mu2m				1.2532	1.0463	**0.9937**
m2b				1.1589	1.0460	**1.0317**
b2b	1.2540	1.0371	1.0860	1.0853	1.0504	**1.0322**
mu2b				1.1514	1.0753	**1.0319**
m2mu				1.3248	1.2517	**1.2390**
b2mu	1.3761	1.2509	1.2615	1.4325	1.2683	**1.2393**
mu2mu				1.3784	1.2812	**1.2392**

Table VIII Prediction result of average RMSE

Task	non-transfer			cross-domain		
	PCC	FMM	SVD	CBT	RMGM	CIT
m2m				1.3628	1.2231	**1.1879**
b2m	1.5433	1.2163	1.2433	1.6724	1.2561	**1.1905**
mu2m				1.5073	1.2662	**1.1907**
m2b				1.3736	1.2783	**1.2538**
b2b	1.5594	1.2793	1.3693	1.3932	1.3013	**1.2544**
mu2b				1.4261	1.3229	**1.2541**
m2mu				1.5399	1.4862	**1.4455**
b2mu	1.7330	1.4816	1.4921	1.7263	1.5176	**1.4457**
mu2mu				1.6475	1.5332	**1.4452**

5.4 Parameter Analysis

In this section, we test how the parameters affect the performance of CIT. There are three parameters in the proposed CIT: K, L and λ. K is the number of user groups and L is the number of item groups. λ is the regularization factor for consistent knowledge extraction. For simplicity, only the result for the movie to movie task has been included. Datasets with three sparsity ratios were used to test all three parameters. Both MAE and RMSE were used as evaluation metrics. As the results for MAE were similar to RMSE, only the results for RMSE have been included.

To analyze the parameter λ, K and L were fixed at 40. In Fig. 6, we can see that RMSEs were not influenced significantly when λ was varied from 0.1 to 1.0. As for K and L, the number of user groups and the number of item groups did affect the RMSE, with a similar influence as described in previous papers: the higher the

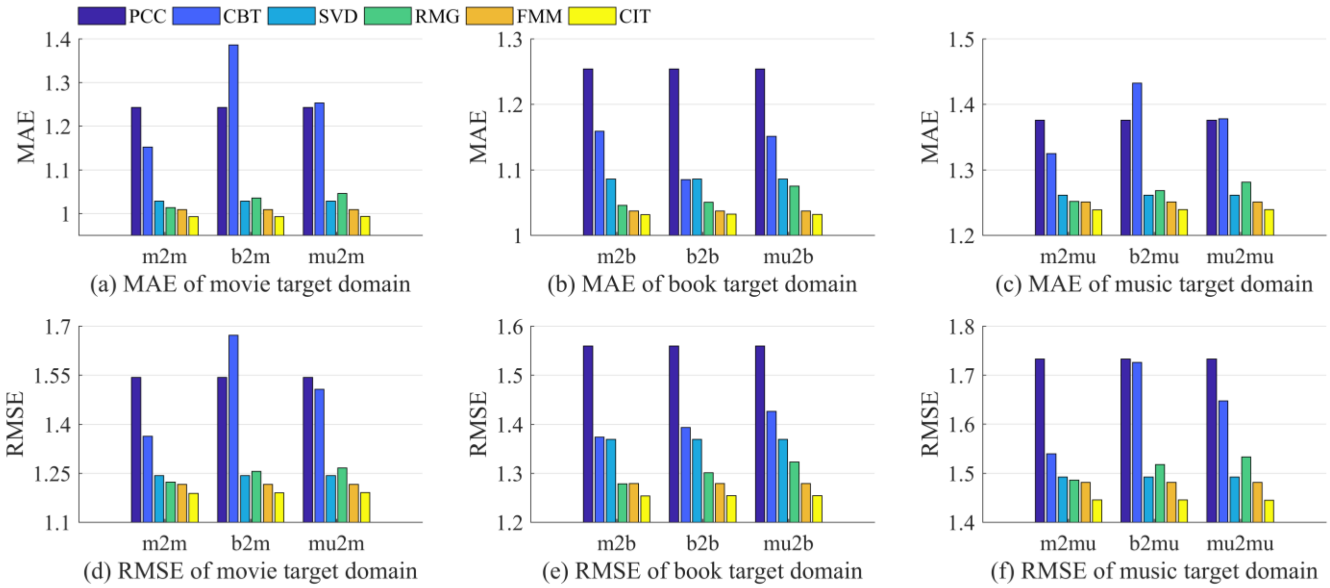
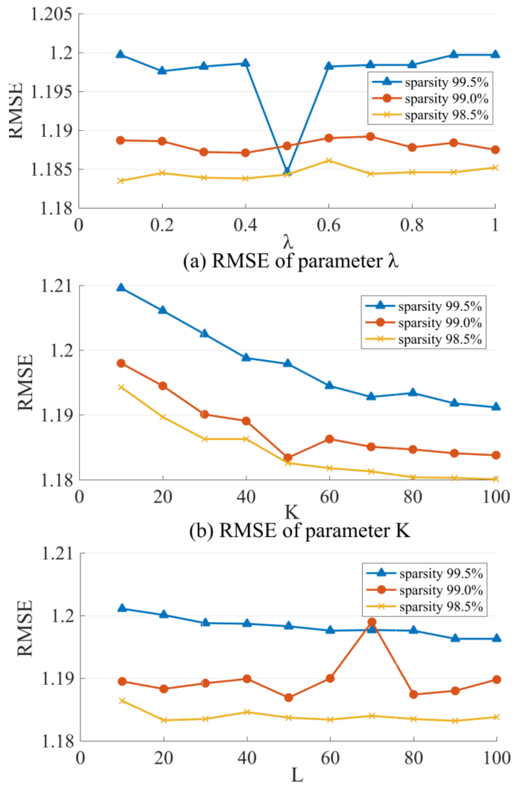


Fig. 5. Prediction result for all methods

number, the higher the accuracy. In the range of 10 to 100, the influence of K and L is not significant. However, it took more time to run the algorithm when higher K and L values were chosen. This phenomenon was especially remarkable when K and L were larger than 100. To trade-off between an acceptable running speed for the algorithm and relative accuracy on RMSE, $K = 40$ and $L = 40$ were chosen for all experiments.

6. Discussion and Conclusion

Making decisions from an overwhelming volume of information is a crucial problem for both businesses and individual customers. And when a business begins operating in a new area, most existing recommender systems are not able to provide much guidance. The cross-domain recommendation method presented in this paper is intended to help businesses and individual customers with decision-making in unchartered waters.



(a) RMSE of parameter λ

(b) RMSE of parameter K

(c) RMSE of parameter L

Fig. 6. Results of RMSE with different parameter settings

6.1 Guidelines for Recommender System Developers

Recommender system developers will find the following guidelines useful:

Guideline #1: The CIT method should be used when two domains have different sparsity ratios. One domain should have a relatively sufficient amount of data; the other should be relatively sparse. There is no need to ensure user/item correspondence between the two domains.

Guideline #2: The CIT has been specially developed for two domains with divergent statistical properties (average and variance) and is appropriate for any divergence condition.

Guideline #3: If the users in target domain have no ratings at all, the CIT method is not suitable. If the sparsity ratio is more than 2.5%, developers should carefully consider whether or not to use the CIT method.

Guideline #4: The range of ratings should be normalized before using the CIT method.

6.2 Practical Applications

The proposed cross-domain recommendation method can be used to solve cold-start problems – a significant issue in the development and application of recommender systems. Developers can use this method to effectively transfer knowledge from a source domain with sufficient data to enhance recommendation models in a target domain. Our proposed method can be used when developing a recommender system to help businesses determine marketing strategies and to attract customers. The method can also provide end users with more effective decision-making support at the initial stage of a recommender system when very little data is available in the target domain. The improved recommendations the system provides will in turn help attract users, making the system grow more feasible and useful over time. Some examples of practical applications are provided below.

Our proposed method is used in the telecom product/service recommender system [24]. Telecommunications companies often introduce new product/service categories, such as new kinds of mobile plans. To attract customers to their new revenue lines, it is important to generate accurate recommendations, and that requires new and specific recommendation models. However, creating an effective recommendation model with very little user and sales data can be challenging when a new product category is first introduced. Through the proposed method, sales data from a similar product category can be used as the source domain to enhance the recommendation model.

Our proposed method is also used in Smart BizSeeker, a B2B recommender system [25]. Smart BizSeeker aims to recommend appropriate business partners to businesses in Australia. It also suffers from the cold-start problem, as initially there is very little rating data between businesses. However, similar B2B websites, such as Alibaba[6], contain a great deal of business rating data, which provides an opportunity to enhance Smart BizSeeker's recommendation model. The proposed cross-domain recommendation method effectively transfers knowledge from the rating data of other B2B websites to Smart BizSeeker to alleviate the cold-start problem.

Our method can also solve cold-start problems in G2B and G2C recommender systems [26] with a relevant source domain that contains sufficient rating data.

[6] https://www.alibaba.com/

6.3 Conclusion and Further Study

Developing CDRS is an efficient way to deal with the cold-start problem in recommender systems. However, using cross-domain recommendation without considering domain shift is little better than gambling [27]. If the knowledge extracted from the source domain just happens to fit the target domain, the quality of recommendations may not suffer. However, if the knowledge does not, the likely result is inaccurate, poor quality recommendations. In this paper, we proposed the CIT method to transfer consistent knowledge learned from a source domain to assist recommendations in a target domain with insufficient rating data. Unlike previous research on knowledge transfer recommender systems, our work investigates what knowledge to transfer and how to effectively transfer that knowledge from the source domain to the target domain. We put forward a tri-factorization method for a cross-domain knowledge transfer recommender system to acquire consistent knowledge. One advantage of the CIT method is that user and item groups are aligned using domain adaptation techniques to ensure consistent user/item group information in both domains. Another advantage is that the method does not require corresponding users and items across domains. Experiments were conducted on five real-world datasets spanning three categories of data and nine cross-domain recommendation tasks. The results show that the proposed CIT method achieves better performance than five other methods in both single and cross-domain settings. The CIT performs particularly well, comparatively, when there is wide deviation in the rating averages between domains.

Cold-start problems are frequent in real-world applications, giving CDRSs great practical significance. However, there are many research gaps to be filled including: the types of situations that benefit from transfer learning; the sparsity levels of the data required for the target and source domains; and how to choose the most optimal source domain to assist transfer learning. If these questions are solved, CDRS can be better applied to markets and industry. Our future work will focus on developing a combined framework that containing more scenarios. To date, our work has only taken explicit rating data into consideration; more information, such as user feedback, item attributes, and implicit data needs to be considered. In addition, new customers in our experimental scenarios each have five ratings, and future work will explore 'pure' cold-start problems where new users have no ratings at all.

Appendix A. GFK operators Ψ_G and Φ_G

$U_s^{(1)}, V_s^{(1)}, U_t^{(1)}, V_t^{(1)}$ are obtained through maps Ψ_s, Ψ_t, Φ_s and Φ_t. According to equations (12)-(15), the cores of these maps are GFK operators Ψ_G and Φ_G. We refer readers to [23] for the details. Here we briefly introduce how the user membership matrixes are unified to the same domain-invariant feature space. The matrixes of items are the same as the users.

Let $P_s, P_t \in \mathbb{R}^{K\times d}$ denote the two sets of bases for the subspaces of the source user membership matrix $U_s^{(0)}$ and the target user membership matrix $U_t^{(0)}$, where K is the dimensionality of the matrixes, i.e., the number of user groups, and d is the dimension of the subspace. The subspaces can be obtained by principle component analysis (PCA) or other methods. R_s is the orthogonal component to P_s. By performing generalized SVD,

$$P_t^T P_t = U_1 \Gamma V^T, R_s^T P_t = -U_2 \Sigma V^T \tag{A.1}$$

where Γ and $\Sigma \in \mathbb{R}^{d\times d}$ are diagonal matrixes. The diagonal elements of Γ and Σ are $cos\,\theta_i$ and $sin\,\theta_i$, where $i = 1,2,\dots,d$. θ_i are the angles between subspaces P_s and P_t.

To ensure the consistency of the user groups between both domains, the GFK operator is used to map the original user group membership matrixes to a domain-invariant space:

$$\Psi_G\left(U_s^{(0)}, U_t^{(0)}\right) = U_s^{(0)} L \tag{A.2}$$

where L is G's square root, $L^T L = G$,

$$G = [P_s U_1 \quad R_s U_2] \begin{bmatrix} \Lambda_1 & \Lambda_2 \\ \Lambda_2 & \Lambda_3 \end{bmatrix} \begin{bmatrix} U_1^T P_s^T \\ U_2^T R_s^T \end{bmatrix} \tag{A.3}$$

where Λ_1 to Λ_3 are diagonal matrixes whose diagonal elements are $\lambda_1 = 1 + \frac{sin(2\theta_i)}{2\theta_i}$, $\lambda_2 = \frac{cos(2\theta_i)-1}{2\theta_i}$, $\lambda_3 = 1 - \frac{sin(2\theta_i)}{2\theta_i}$.

According to equations (16) and (17), $U_s^{(1)}, V_s^{(1)}, U_t^{(1)}$ and $V_t^{(1)}$ are obtained.

Appendix B. Proof of f_u and f_v ensuring consistency

A definition for maps like f_u and f_v is given as follows.

Definition 3 (Distribution Consistency Maps). Given a source rating matrix $\boldsymbol{X}_s \in \mathbb{R}^{M_s \times N_s}$ and a target rating matrix $\boldsymbol{X}_t \in \mathbb{R}^{M_t \times N_t}$, the tri-factorizations of $\boldsymbol{X}_s$ and $\boldsymbol{X}_t$ are group-consistent and they share consistent knowledge $\boldsymbol{S}$ such that

$$\boldsymbol{X}_s = \boldsymbol{U}_s^{(1)} \boldsymbol{S} \left(\boldsymbol{V}_s^{(1)}\right)^T \quad \text{(B.1)}$$

$$\boldsymbol{X}_t = \boldsymbol{U}_t^{(1)} \boldsymbol{S} \left(\boldsymbol{V}_t^{(1)}\right)^T \quad \text{(B.2)}$$

where $\boldsymbol{U}_s^{(1)}, \boldsymbol{U}_t^{(1)}$ are user group membership matrixes unified to the same domain-invariant feature space for the source and target domains, while $\boldsymbol{V}_s^{(1)}$ and $\boldsymbol{V}_t^{(1)}$ are unified item group membership matrixes. If maps f_u and f_v satisfy equation (20), we call f_u and f_v distribution consistency maps (DCM) for the two rating matrixes.

For a demonstration of a DCM map, we refer readers to some theoretical results in [28] for reliable unsupervised knowledge transfer including a linear monotonic map (LMM) and its related theorem. LMM is a map: $f(\boldsymbol{\mathcal{X}}) = \boldsymbol{\mathcal{X}}\boldsymbol{u}$, $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{m \times n}$, $\boldsymbol{u} \in \mathbb{R}^{n \times 1}$. A theorem for reliable unsupervised knowledge transfer is then given in [28], proving that LMM can ensure the process of unsupervised knowledge transfer is reliable. As in our situation, we give the theorem and proof as follows:

Theorem 1. Given a source rating matrix $\boldsymbol{X}_s \in \mathbb{R}^{M_s \times N_s}$ and a target rating matrix $\boldsymbol{X}_t \in \mathbb{R}^{M_t \times N_t}$, the tri-factorizations of $\boldsymbol{X}_s$ and $\boldsymbol{X}_t$ are group-consistent and they share a consistent knowledge $\boldsymbol{S}$ as in equations (B.1) and (B.2). When $\boldsymbol{u} \geq \boldsymbol{0}$ and $\boldsymbol{v} \geq \boldsymbol{0}$, $f_u\left(\boldsymbol{U}_t^{(1)}\right) = \boldsymbol{U}_t^{(1)}\boldsymbol{u}$ and $f_v\left(\boldsymbol{V}_t^{(1)}\right) = \boldsymbol{V}_t^{(1)}\boldsymbol{v}$, they are DCMs for two rating matrixes.

Proof. When $\boldsymbol{u} \geq \boldsymbol{0}$ and $\boldsymbol{v} \geq \boldsymbol{0}$, $f_u\left(\boldsymbol{U}_t^{(1)}\right) = \boldsymbol{U}_t^{(1)}\boldsymbol{u}$ and $f_v\left(\boldsymbol{V}_t^{(1)}\right) = \boldsymbol{V}_t^{(1)}\boldsymbol{v}$ can satisfy the following equation:

$$P\left(\boldsymbol{S} \middle| \boldsymbol{U}_t^{(1)}\boldsymbol{u}, \boldsymbol{V}_t^{(1)}\right) = P\left(\boldsymbol{S} \middle| \boldsymbol{U}_t^{(1)}\boldsymbol{I}, \boldsymbol{V}_t^{(1)}\right)$$

Then, to fix $f_u\left(\boldsymbol{U}_t^{(1)}\right)$, we use the following equation:

$$P\left(\boldsymbol{S} \,\middle|\, \boldsymbol{U}_t^{(1)}\boldsymbol{u}, \boldsymbol{V}_t^{(1)}\boldsymbol{v}\right) = P\left(\boldsymbol{S} \,\middle|\, \boldsymbol{U}_t^{(1)}\boldsymbol{u}, \boldsymbol{V}_t^{(1)}\boldsymbol{I}\right)$$

So, we then have

$$P\left(\boldsymbol{S} \,\middle|\, f_u\left(\boldsymbol{U}_t^{(1)}\right), f_v\left(\boldsymbol{V}_t^{(1)}\right)\right) = P\left(\boldsymbol{S} \,\middle|\, \boldsymbol{U}_t^{(1)}, \boldsymbol{V}_t^{(1)}\right)$$

Based on Definition 3, $f_u\left(\boldsymbol{U}_t^{(1)}\right) = \boldsymbol{U}_t^{(1)}\boldsymbol{u}$ and $f_v\left(\boldsymbol{V}_t^{(1)}\right) = \boldsymbol{V}_t^{(1)}\boldsymbol{v}$ are DCMs. □

Hence, the LMM is proven to be a DCM, which means we can let f_u and f_v have the following expressions:

$$f_u\left(\boldsymbol{U}_t^{(1)}\right) = \boldsymbol{U}_t^{(1)}\boldsymbol{u}, \boldsymbol{u} \geq \boldsymbol{0} \quad \text{(B.3)}$$

$$f_v\left(\boldsymbol{V}_t^{(1)}\right) = \boldsymbol{V}_t^{(1)}\boldsymbol{v}, \boldsymbol{v} \geq \boldsymbol{0} \quad \text{(B.4)}$$

where $\boldsymbol{u} \in \mathbb{R}^{K \times K}$ is user tuning factor and $\boldsymbol{v} \in \mathbb{R}^{L \times L}$ is item tuning factor.

Acknowledgement

This work was supported by the Australian Research Council (ARC) under Discovery Grant [DP150101645].

References

[1] J. Lu, D. Wu, M. Mao, W. Wang, and G. Zhang, "Recommender system application developments: a survey," *Decision Support Systems*, vol. 74, pp. 12-32, 2015.

[2] M. Deshpande and G. Karypis, "Item-based top-n recommendation algorithms," *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 143-177, 2004.

[3] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 22, no. 8, pp. 30-37, 2009.

[4] J. Liu, C. Wu, and W. Liu, "Bayesian probabilistic matrix factorization with social relations and item contents for recommendation," *Decision Support Systems*, vol. 55, no. 3, pp. 838-850, 2013.

[5] H. Wang, N. Wang, and D.-Y. Yeung, "Collaborative deep learning for recommender systems," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015, pp. 1235-1244: ACM.

[6] H.-N. Kim, A. El-Saddik, and G.-S. Jo, "Collaborative error-reflected models for cold-start recommender systems," *Decision Support Systems*, vol. 51, no. 3, pp. 519-531, 2011.

[7] V. Y. Yoon, R. E. Hostler, Z. Guo, and T. Guimaraes, "Assessing the moderating effect of consumer product knowledge and online shopping experience on using recommendation agents for customer loyalty," *Decision Support Systems*, vol. 55, no. 4, pp. 883-893, 2013.

[8] X. Li, M. Wang, and T.-P. Liang, "A multi-theoretical kernel-based approach to social network-based recommendation," *Decision Support Systems*, vol. 65, pp. 95-104, 2014.

[9] Y.-M. Li, C.-T. Wu, and C.-Y. Lai, "A social recommender mechanism for e-commerce: Combining similarity, trust, and relationship," *Decision Support Systems*, vol. 55, no. 3, pp. 740-752, 2013.

[10] J. McAuley and A. Yang, "Addressing complex and subjective product-related queries with customer reviews," in *Proceedings of the 25th International Conference on World Wide Web*, 2016, pp. 625-635: International World Wide Web Conferences Steering Committee.

[11] J. Lu, V. Behbood, P. Hao, H. Zuo, S. Xue, and G. Zhang, "Transfer learning using computational intelligence: a survey," *Knowledge-Based Systems*, vol. 80, pp. 14-23, 2015.

[12] W. Pan, "A survey of transfer learning for collaborative recommendation with auxiliary data," *Neurocomputing*, vol. 177, pp. 447-453, 2016.

[13] W. Chen, W. Hsu, and M. L. Lee, "Making recommendations from multiple domains," in *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2013, pp. 892-900: ACM.

[14] A. P. Singh and G. J. Gordon, "Relational learning via collective matrix factorization," in *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2008, pp. 650-658: ACM.

[15] N. Mirbakhsh and C. X. Ling, "Improving top-n recommendation for cold-start users via cross-domain information," *ACM Transactions on Knowledge Discovery from Data*, vol. 9, no. 4, p. 33, 2015.

[16] W. Pan and Q. Yang, "Transfer learning in heterogeneous collaborative filtering domains," *Artificial Intelligence*, vol. 197, pp. 39-55, 2013.

[17] L. Hu, J. Cao, G. Xu, L. Cao, Z. Gu, and C. Zhu, "Personalized recommendation via cross-domain triadic factorization," in *Proceedings of the 22nd International Conference on World Wide Web*, 2013, pp. 595-606: ACM.

[18] B. Li, Q. Yang, and X. Xue, "Can movies and books collaborate? Cross-domain collaborative filtering for sparsity reduction," in *Proceedings of the 21th International Joint Conference on Artificial Intelligence*, 2009, vol. 9, pp. 2052-2057.

[19] B. Li, Q. Yang, and X. Xue, "Transfer learning for collaborative filtering via a rating-matrix generative model," in *Proceedings of the 26th Annual International Conference on Machine Learning*, 2009, pp. 617-624: ACM.

[20] B. Li, X. Zhu, R. Li, and C. Zhang, "Rating knowledge sharing in cross-domain collaborative filtering," *IEEE Transactions on Cybernetics*, vol. 45, no. 5, pp. 1068-1082, 2015.

[21] C. Ding, T. Li, W. Peng, and H. Park, "Orthogonal nonnegative matrix tri-factorizations for clustering," in *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2006, pp. 126-135: ACM.

[22] L. Si and R. Jin, "Flexible mixture model for collaborative filtering," in *Proceedings of the 20th International Conference on Machine Learning*, 2003, vol. 3, pp. 704-711.

[23] B. Gong, K. Grauman, and F. Sha, "Learning kernels for unsupervised domain adaptation with applications to visual object recognition," *International Journal of Computer Vision*, vol. 109, no. 1-2, pp. 3-27, 2014.

[24] Z. Zhang, H. Lin, K. Liu, D. Wu, G. Zhang, and J. Lu, "A hybrid fuzzy-based personalized recommender system for telecom products/services," *Information Sciences*, vol. 235, pp. 117-129, 2013.

[25] D. Wu, G. Zhang, and J. Lu, "A fuzzy preference tree-based recommender system for personalized business-to-business e-services," *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 1, pp. 29-43, 2015.

[26] M. Al-Hassan, H. Lu, and J. Lu, "A semantic enhanced hybrid recommendation approach: a case study of e-Government tourism service recommendation system," *Decision Support Systems*, vol. 72, pp. 97-109, 2015.

[27] P. Cremonesi and M. Quadrana, "Cross-domain recommendations without overlapping data: myth or reality?," in *Proceedings of the 8th ACM Conference on Recommender systems*, 2014, pp. 297-300: ACM.

[28] F. Liu, G. Zhang, H. Lu, and J. Lu, "Heterogeneous Unsupervised Cross-domain Transfer Learning," *arXiv:1701.02511 [cs.LG]*, pp. 1–47, 2017. https://arxiv.org/abs/1701.02511.