

DDTCDR: Deep Dual Transfer Cross Domain Recommendation

Pan Li

New York University
44th West 4th Street, NY, US
pli2@stern.nyu.edu

Alexander Tuzhilin

New York University
44th West 4th Street, NY, US
atuzhili@stern.nyu.edu

ABSTRACT

Cross domain recommender systems have been increasingly valuable for helping consumers identify the most satisfying items from different categories. However, previously proposed cross-domain models did not take into account bidirectional latent relations between users and items. In addition, they do not explicitly model information of user and item features, while utilizing only user ratings information for recommendations. To address these concerns, in this paper we propose a novel approach to cross-domain recommendations based on the mechanism of dual learning that transfers information between two related domains in an iterative manner until the learning process stabilizes. We develop a novel latent orthogonal mapping to extract user preferences over multiple domains while preserving relations between users across different latent spaces. Combining with autoencoder approach to extract the latent essence of feature information, we propose Deep Dual Transfer Cross Domain Recommendation (DDTCDR) model to provide recommendations in respective domains. We test the proposed method on a large dataset containing three domains of movies, book and music items and demonstrate that it consistently and significantly outperforms several state-of-the-art baselines and also classical transfer learning approaches.

CCS CONCEPTS

• Information systems → Recommender systems.

KEYWORDS

Cross Domain Recommendation, Deep Learning, Transfer Learning, Dual Learning, Autoencoder

ACM Reference Format:

Pan Li and Alexander Tuzhilin. 2020. DDTCDR: Deep Dual Transfer Cross Domain Recommendation. In *The Thirteenth ACM International Conference on Web Search and Data Mining (WSDM '20)*, February 3–7, 2020, Houston, TX, USA. ACM, New York, NY, USA, 9 pages. <https://doi.org/10.1145/3336191.3371793>

1 INTRODUCTION

Recommender systems have become the key component for online marketplaces to achieve great success in understanding user

preferences. Collaborative filtering (CF) [31] approaches, especially matrix factorization (MF) [9] methods constitute the corner stone to help user discover satisfying products. However, these models suffer the cold start and the data sparsity problems [1, 32], for we only have access to a small fraction of past transactions, which makes it hard to model user preferences accurately and efficiently.

To address these problems, researchers propose to use cross domain recommendation [3] through transfer learning [25] approaches that learn user preferences in the source domain and transfer them to the target domain. For instance, if a user watches a certain movie, we will recommend the original novel on which the movie is based to that user. Most of the methods along this line focus on the unidirectional transfer learning from the source domain to the target domain and achieve great recommendation performance [3]. In addition, it is beneficial for us to also transfer user preferences in the other direction via dual transfer learning [21, 37, 44]. For example, once we know the type of books that the user would like to read, we can recommend movies on related topics to form a loop for better recommendations in both domains [11].

However, previous dual transfer models only focus on explicit information of users and items, without considering latent and complex relations between them. Besides, they do not include user and item features during the recommendation process, thus limiting the recommender system to provide satisfying results. Latent embedding methods, on the other hand, constitute a powerful tool to extract latent user preferences from the data record and model user and item features efficiently [43]. Therefore, it is crucially important to design a model that utilize latent embedding approach to conduct dual transfer learning for cross domain recommendations.

In this paper, we apply the idea of deep dual transfer learning to cross-domain recommendations using latent embeddings of user and item features. We assume that if two users have similar preferences in a certain domain, their preferences should also be similar across other domains as well. We address this assumption by proposing a unifying mechanism that extracts the essence of preference information in each domain and improves the recommendations for both domains simultaneously through better understanding of user preferences.

Specifically, we propose Deep Dual Transfer Cross Domain Recommendation (DDTCDR) model that learns latent orthogonal mappings across domains and provides cross domain recommendations leveraging user preferences from all domains. Furthermore, we empirically demonstrate that by iteratively updating the dual recommendation model, we simultaneously improve recommendation performance over both domains and surpass all baseline models. Compared to previous approaches, the proposed model has the following advantages:

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

WSDM '20, February 3–7, 2020, Houston, TX, USA

© 2020 Association for Computing Machinery.

ACM ISBN 978-1-4503-6822-3/20/02...\$15.00

<https://doi.org/10.1145/3336191.3371793>

- It transfers latent representations of features and user preferences instead of explicit information between the source domain and the target domain to capture latent and complex interactions as well as modeling feature information.
- It utilizes deep dual transfer learning mechanism to enable bidirectional transfer of user preferences that improves recommendation performance in both domains simultaneously over time.
- It learns the latent *orthogonal* mapping function across the two domains, that is capable of preserving similarities of user preferences and computing the inverse mapping function efficiently.

In this paper, we make the following contributions:

- We propose to apply the combination of dual transfer learning mechanism and latent embedding approach to the problem of cross domain recommendations.
- We empirically demonstrate that the proposed model outperforms the state-of-the-art approaches and improves recommendation accuracy across multiple domains and experimental settings.
- We theoretically demonstrate the convergence condition for the simplified case of our model and empirically show that the proposed model stabilizes and converges after several iterations.
- We illustrate that the proposed model can be easily extended for multiple-domain recommendation applications.

2 RELATED WORK

The proposed model stems from two research directions: cross domain recommendations and deep-learning based recommendations. Also, we discuss the literature on dual transfer learning, and how they motivate the combination of dual transfer learning mechanism and latent embedding methods for cross domain recommendations.

2.1 Cross Domain and Transfer Learning-based Recommendations

Cross domain recommendation approach [3] constitutes a powerful tool to deal with the data sparsity problem. Typical cross domain recommendation models are extended from single-domain recommendation models, including CMF [34], CDCF [7, 11], CDFM [23], Canonical Correlation Analysis [27–30] and Dual Regularization [40]. These approaches assume that different patterns characterize the way that users interact with items of a certain domain and allow interaction information from an auxiliary domain to inform recommendation in a target domain.

The idea of information fusion also motivates the use of transfer learning [25] that transfers extracted information from the source domain to the target domain. Specifically, researchers [6, 17] propose to learn the user preference in the source domain and transfer the preference information into target domain for better understanding of user preferences. These models have achieved great success in addressing the cold-start problem and enhancing recommendation performance.

However, these models do not fundamentally address the relationship between different domains, for they do not improve recommendation performance of both domains simultaneously, thus

might not release the full potential of utilizing the cross-domain user interaction information. Also they do not explicitly model user and item features during recommendation process. In this paper, we propose to use a novel dual transfer learning mechanism combining with autoencoder to overcome these issues and significantly improve recommendation performance.

2.2 Dual Transfer Learning

Transfer learning [25] deals with the situation where the data obtained from different resources are distributed differently. It assumes the existence of common knowledge structure that defines the domain relatedness, and incorporate this structure in the learning process by discovering a shared latent feature space in which the data distributions across domains are close to each other. The existing transfer learning methods for cross-domain recommendation include Collaborative DualPLSA [45], Joint Subspace Nonnegative Matrix Factorization [20] and JDA [22] that learn the latent factors and associations spanning a shared subspace where the marginal distributions across domains are close.

In addition, to exploit the duality between these two distributions and to enhance the transfer capability, researchers propose the dual transfer learning mechanism [21, 37, 44] that simultaneously learns the marginal and conditional distributions. Recently, researchers manage to achieve great performance on machine translation with dual-learning mechanism [4, 39, 42]. All these successful applications address the importance of exploiting the duality for mutual reinforcement. However, none of them apply the dual transfer learning mechanism into cross-domain recommendation problems, where the duality lies in the symmetrical correlation between source domain and target domain user preferences. In this paper, we utilize a novel dual-learning mechanism and significantly improve recommendation performance.

2.3 Deep Learning-based Recommendations

Recently, deep learning has been revolutionizing the recommendation architectures dramatically and brings more opportunities to improve the performance of existing recommender systems. To capture the latent relationship between users and items, researchers propose to use deep learning based recommender systems [38, 43], especially embedding methods [5, 12–15] and autoencoding methods [16, 18, 33, 41] to extract the latent essence of user-item interactions for the better understanding of user preferences.

However, user preferences in different domains are learned separately without exploiting the duality for mutual reinforcement, for researchers have not addressed the combination of deep learning methods with dual transfer learning mechanism in recommender systems, which learns user preferences from different domains simultaneously and further improves the recommendation performance. To this end, we propose a dual transfer collaborative filtering model that captures latent interactions across different domains. The effectiveness of dual transfer learning over the existing methods is demonstrated by extensive experimental evaluation.

3 METHOD

In this section, we present the proposed Deep Dual Transfer Cross Domain Recommendation (DDTCDR) model. First, we construct

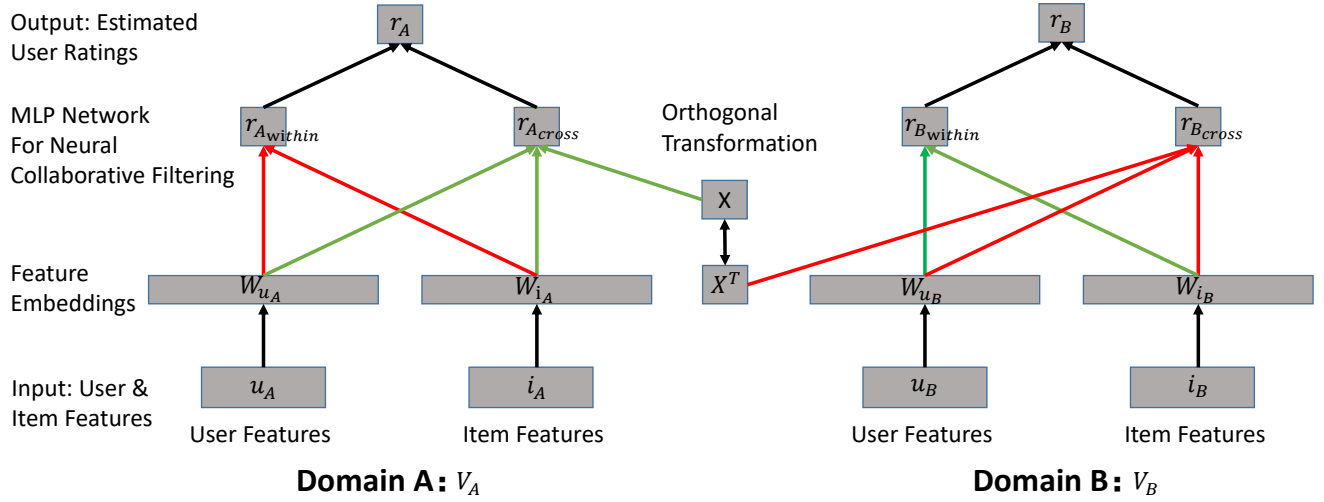


Figure 1: Model Framework: Red and blue lines represent the recommendation model for domain A and B respectively. We obtain the estimated ratings by taking the linear combination of within-domain and cross-domain user preferences and back-propagate the loss to update the two models and orthogonal mappings simultaneously.

Algorithm 1 Dual Neural Collaborative Filtering

- 1: **Input:** Domain V_A and V_B , autoencoder AE_A and AE_B , transfer rate α , learning rates γ_A and γ_B , initial recommendation models RS_A and RS_B , initial mapping function X
 - 2: **repeat**
 - 3: Sample user-item records d_A and d_B from V_A and V_B respectively
 - 4: Unpack records d_A, d_B as user features u_A, u_B , item features i_A, i_B and ratings r_A, r_B
 - 5: Generate feature embeddings from autoencoder as $W_{u_A} = AE_A(u_A)$, $W_{u_B} = AE_B(u_B)$, $W_{i_A} = AE_A(i_A)$, $W_{i_B} = AE_B(i_B)$
 - 6: Estimate the ratings in domain A via $r'_A = (1 - \alpha)RS_A(W_{u_A}, W_{i_A}) + \alpha RS_B(X * W_{u_A}, W_{i_A})$
 - 7: Estimate the ratings in domain B via $r'_B = (1 - \alpha)RS_B(W_{u_B}, W_{i_B}) + \alpha RS_A(X^T * W_{u_B}, W_{i_B})$
 - 8: Compute MSE loss $\hat{r}_A = r_A - r'_A$, $\hat{r}_B = r_B - r'_B$
 - 9: Backpropagate \hat{r}_A, \hat{r}_B and update RS_A, RS_B ;
 - 10: Backpropagate orthogonal constraint on X ; Orthogonalize X
 - 11: **until** convergence
-

feature embeddings using the autoencoding technique from the user and item features. Then we design a specific type of latent orthogonal mapping function for transferring feature embeddings across two different domains through dual transfer learning mechanism. Moreover, we theoretically demonstrate convergence of the proposed model under certain assumptions. The complete architecture of DDTCDR system is illustrated in Figure 1. As shown in Figure 1, we take the user and item features as input and map them into the feature embeddings, from which we compute the within-domain and cross-domain user preferences and provide recommendations. Important mathematical notations used in our model are listed in Table 1. We explain the details in the following sections.

3.1 Feature Embeddings

To effectively extract latent user preferences and efficiently model features of users and items, we present an autoencoder framework that learns the latent representations of user and item features and transforms the heterogeneous and discrete feature vectors into continuous feature embeddings. We denote the feature information for

user a as $u_a = \{u_{a_1}, u_{a_2}, \dots, u_{a_m}\}$ and the feature information

Table 1: Mathematical Notations.

Symbol	Description
u	User Features
i	Item Features
W_u	User Feature Embeddings
W_i	Item Feature Embeddings
γ	Learning Rate
r	Estimated Ratings
r_{within}	Within-Domain Estimated Ratings
r_{cross}	Cross-Domain Estimated Ratings
X	Latent Orthogonal Mapping
X^T	Transpose of Latent Orthogonal Mapping
RS	Domain-Specific Recommender System
AE	Domain-Specific Autoencoder
α	Hyperparameter in Hybrid Utility Function

for item b as $i_b = \{i_{b_1}, i_{b_2}, \dots, i_{b_n}\}$, where m and n stand for the dimensionality of user and item feature vectors respectively. The goal is to train two separate neural networks: encoder that maps feature vectors into latent embeddings, and decoder that reconstructs feature vectors from latent embeddings. Due to effectiveness and efficiency of the training process, we formulate both the encoder and the decoder as multi-layer perceptron (MLP). MLP learns the hidden representations by optimization reconstruction loss L :

$$L = \|u_a - MLP_{dec}(MLP_{enc}(u_a))\| \quad (1)$$

where MLP_{enc} and MLP_{dec} represents the MLP network for encoder and decoder respectively. Note that, in this step we train the autoencoder separately for users and items in different domains to avoid the information leak between domains.

3.2 Latent Orthogonal Mapping

In this section we introduce the latent orthogonal mapping function for transferring user preferences from the source domain to the target domain. The fundamental assumption of our proposed approach is that, if two users have similar preferences in the source domain, their preferences should also be similar in the target domain. This implies that we need to preserve similarities between users during transfer learning across domains. In particular, we propose to use a latent orthogonal matrix to transfer information across domains for two reasons. First, it preserves similarities between user embeddings across different latent spaces since orthogonal transformation preserves inner product of vectors. Second, it automatically derives the inverse mapping matrix as X^T , because $Y = X^T(XY)$ holds for any given orthogonal mapping matrix X , thus making inverse mapping matrix equivalent to its transpose. This simplifies the learning procedure and reduces the perplexity of the recommendation model.

Using the latent orthogonal mapping, we could transfer user preferences from one domain to the other. In the next section, we will introduce model to utilize this latent orthogonal mapping for constructing dual learning mechanism for cross domain recommendations.

3.3 Deep Dual Transfer Learning

As an important tool to provide recommendations, matrix factorization methods associate user-item pairs with a shared latent space, and use the latent feature vectors to represent users and items. In order to model latent interactions between users and items as well as feature information, it is natural to generalize matrix factorization methods using latent embedding approaches. However, in the cross domain recommendation application, we may not achieve the optimal performance by uniformly applying neural network model to all domains of users and items due to the cold-start and sparsity problems.

To address these problems and improve performance of recommendations, we propose to combine dual transfer learning mechanism with cross domain recommendations, where we transfer user preferences between the source domain and the target domain simultaneously. Consider two different domains D_A and D_B that contain user-item interactions as well as user and item features. In real business practices, user group in D_A often overlaps with D_B ,

meaning that they have purchased items in both domains, while there is no overlap of item between two domains for each item only belongs to one single domain. One crucial assumption we make here is that, if two users have similar preferences in D_A , they are supposed to have similar preferences in D_B as well, indicating that we can learn and improve user preferences in both domains simultaneously over time. Therefore to obtain better understanding of user preferences in D_A , we also utilize external user preference information from D_B and combine them together. Similarly, we could get better recommendation performance in D_B if we utilize user preference information from D_A at the same step. To leverage duality of the two transfer learning based recommendation models and to improve effectiveness of both tasks simultaneously, we conduct dual transfer learning recommendations for the two models *together* and learn the latent mapping function accordingly.

Specifically, we propose to model user preferences r using two components: *within-domain preference* r_{within} that captures user interactions and predict user behaviors in the target domain and *cross-domain preference* r_{cross} that utilizes user actions from the source domain. We also introduce transfer rate α as a hyperparameter, which represents the relative importance of the two components in prediction of user preferences. We propose to estimate user ratings in domain pairs (A, B) as follows:

$$r'_A = (1 - \alpha)RS_A(W_{u_A}, W_{i_A}) + \alpha RS_B(X * W_{u_A}, W_{i_A}) \quad (2)$$

$$r'_B = (1 - \alpha)RS_B(W_{u_B}, W_{i_B}) + \alpha RS_A(X^T * W_{u_B}, W_{i_B}) \quad (3)$$

where $W_{u_A}, W_{i_A}, W_{u_B}, W_{i_B}$ represents user and item embeddings and RS_A, RS_B stand for the neural recommendation model for domain A and B respectively. The first term in each equation computes within-domain user preferences from user and item features in the same domain, while the second term denotes cross-domain user preferences obtained by the latent orthogonal mapping function X to capture heterogeneity of different domains. We use weighted linear combination for rating estimations in equations (4) and (5). When $\alpha = 0$ or there is no user overlap between two domains, the dual learning model degenerates to two separate single-domain recommendation models; when $\alpha = 0.5$ and $W = \mathbf{I}$, the dual learning model degenerates to one universal recommendation model across two domains. For users that only appear in one domain, the hyperparameter α is selected as 0. For users that appear in both domains, α normally should take a positive value between 0 and 0.2 based on the experimental results conducted in Section 6, which indicates that within-domain preferences play the major role in the understanding of user behavior.

As shown in Figure 1 and equations (4) and (5), dual transfer learning entails the transfer loop across two domains and the learning process goes through the loop iteratively. It is important to study the convergence property of the model, which we discuss in the next section.

3.4 Convergence Analysis

In this section, we present the convergence theorem of matrix factorization with dual transfer learning mechanism. We denote the rating matrices as V_A, V_B for the two domains A and B respectively. The goal is to find the approximate non-negative factorization matrices that simultaneously minimizes the reconstruction loss. We

conduct multiplicative update algorithms [19] for the two reconstruction loss iteratively given α , X , V_A and V_B .

$$\min_{W_A, W_B, H_A, H_B} |V_A - (1 - \alpha)W_A H_A - \alpha X W_B H_B| + |V_B - (1 - \alpha)W_B H_B - \alpha X^T W_A H_A| \quad (4)$$

PROPOSITION 1. *Convergence of the iterative optimization of dual matrix factorization for rating matrices V_A and V_B in (6) is guaranteed.*

PROOF. Combining the two parts of objective functions and eliminating $W_A H_A$ term, we will get

$$(1 - \alpha)V_B - \alpha X^T V_A = (1 - 2\alpha)W_B H_B \quad (5)$$

Similarly, when we eliminate $W_B H_B$ term, we will get

$$(1 - \alpha)V_A - \alpha X V_B = (1 - 2\alpha)W_A H_A \quad (6)$$

Based on the analysis in [10] for the classical single-domain matrix factorization, to show that the repeated iteration of update rules is guaranteed to converge to a locally optimal solution, it is sufficient to show that

$$\begin{cases} 2\alpha - 1 < 0 & (a) \\ (1 - \alpha)V_B - \alpha X^T V_A \geq \mathbf{0} & (b) \\ (1 - \alpha)V_A - \alpha X V_B \geq \mathbf{0} & (c) \end{cases}$$

where $\mathbf{0}$ stands for the zero matrix. Condition (a) is an intuitive condition which indicates that the information from the target domain dominantly determines the user preference, while information transferred from the source domain only serves as the regularizer in the learning period. To fulfill the seemingly complicated condition (b) and (c), we recall that V_A and V_B are ratings matrices, which are non-negative and bounded by the rating scale k . We design two "positive perturbation" $V'_A = V_A + mk\mathbf{1}$ and $V'_B = V_B + mk\mathbf{1}$ where m is the rank of the mapping matrix X . Condition (a) is independent of the specific rating matrices, and we could check that V'_A and V'_B satisfies condition (b) and (c). Thus, the matrix factorization of V'_A and V'_B is guaranteed convergence; to reconstruct the original matrix V_A and V_B , we only need to deduce the "positive perturbation" term m , so the original problem is also guaranteed convergence. \square

In this paper, to capture latent interactions between users and items, we use the neural network based matrix factorization approach, therefore it still remains unclear if this convergence happens to the proposed model. However, our hypothesis is that even in our case we will experience similar convergence process guaranteed in the classical matrix factorization case, as stated in the proposition. We test the hypothesis in Section 6.

3.5 Extension to Multiple Domains

In previous sections, we describe the proposed DDTCDR model with special focus on the idea of combining dual transfer learning mechanism with latent embedding approaches. We point out that the proposed model not only works for cross domain recommendations between two domains, but can easily extend to recommendations between multiple domains as well. Consider N different domains $D_1, D_2, D_3, \dots, D_N$. To provide recommendations for domain D_k ,

Domain	Book	Movie	Music
# Users	804,285	959,502	45,962
# Items	182,653	79,866	183,114
# Ratings	223,007,805	51,269,130	2,536,273
Density	0.0157%	0.0669%	0.0301%

Table 2: Descriptive Statistics for the Dataset

we estimate the ratings similarly as the hybrid combination of within-domain estimation and cross-domain estimation:

$$r'_k = (1 - \alpha)RS_k(W_{u_k}, W_{i_k}) + \frac{\alpha}{n - 1} \sum_{j=1; j \neq k}^n RS_j(X_{jk} * W_{u_k}, W_{i_k}) \quad (7)$$

where X_{jk} represents the latent orthogonal transfer matrix between domain D_j and D_k respectively. Therefore, the proposed model is capable of providing recommendations for multiple-domain application effectively and efficiently.

4 EXPERIMENT

To validate the performance of the proposed model, we compare the cross-domain recommendation accuracy with several state-of-the-art methods. In addition, we conduct multiple experiments to study sensitivity of hyperparameters of our proposed model.

4.1 Dataset

We use a large-scale anonymized dataset obtained from a European online recommendation service for carrying our experiments. It allows users to rate and review a range of items from various domains, while each domain was treated as an independent sub-site with separate within-domain recommendations. Consequently, the combination of explicit user feedback and diverse domains makes the dataset unique and valuable for cross-domain recommendations. We select the subset that includes three largest domains - books, movies and music, three domains being linked together through a common user ID identifying the same user. We normalize the scale of ratings between 0 and 1. Note that each user might purchase items from different domains, while each item only belongs to one single domain. The basic statistics about this dataset are shown in Table 2.

4.2 User and Item Features

Besides the records of interactions between users and items, the online platform also collects user information by asking them online questions when they are using the recommender services. Typical examples include questions about preferences between two items, ideas about recent events, life styles, demographic information, etc.. From all these questions, we select eight most-popular questions and use the corresponding answers from users as user features. Meanwhile, although the online platform does not collect directly item features, we obtain the metadata through Google Books API¹, IMDB API² and Spotify API³. To ensure the correctness of the collected item features, we validate the retrieved results with the

¹<https://developers.google.com/books/>

²<http://www.omdbapi.com/>

³<https://developer.spotify.com/documentation/web-api/>

Category	Feature Group	Dimensionality	Type
User Features	Gender	2	one-hot
	Age	$\sim 10^2$	numerical
	Movie Taste	12	one-hot
	Residence	12	one-hot
	Preferred Category	9	one-hot
	Recommendation Usage	5	one-hot
	Marital Status	3	one-hot
Book Features	Personality	6	one-hot
	Category	8	one-hot
	Title	$\sim 10^5$	one-hot
	Author	$\sim 10^4$	multi-hot
	Publisher	$\sim 10^2$	one-hot
	Language	4	one-hot
	Country	4	one-hot
Movie Features	Price	$\sim 10^2$	numeric
	Date	$\sim 10^3$	date
	Genre	6	one-hot
	Title	$\sim 10^5$	one-hot
	Director	$\sim 10^3$	multi-hot
	Writer	$\sim 10^3$	multi-hot
	Runtime	$\sim 10^3$	numeric
Music Features	Country	4	one-hot
	Rating	$\sim 10^2$	numeric
	Votes	$\sim 10^4$	numeric
	Listener	$\sim 10^3$	numeric
	Playcount	$\sim 10^3$	numeric
	Artist	$\sim 10^4$	one-hot
	Album	$\sim 10^4$	one-hot
	Tag	8	one-hot
	Release	$\sim 10^3$	date
	Duration	$\sim 10^3$	numeric
	Title	$\sim 10^5$	one-hot

Table 3: Statistics of feature sets used in the purposed model

timestamp included in the dataset. We describe the entire feature set used in our experiments in Table 3.

4.3 Baseline Models

To conduct experiments and evaluations of our model, we utilize the record-stratified 5-fold cross validation and evaluate the recommendation performance based on RMSE, MAE, Precision and Recall metrics [26]. We compare the performance with a group of state-of-the-art methods.

- **CCFNet[17]** Cross-domain Content-boosted Collaborative Filtering neural NETWORK (CCCFNet) utilizes factorization to tie CF and content-based filtering together with a unified multi-view neural network.
- **CDFM[23]** Cross Domain Factorization Machine (CDFM) proposes an extension of FMs that incorporates domain information in this pattern, which assumes that user interaction patterns differ sufficiently to make it advantageous to model domains separately.
- **CoNet[6]** Collaborative Cross Networks (CoNet) enables knowledge transfer across domains by cross connections between base networks.
- **NCF[5]** Neural Collaborative Filtering (NCF) is a neural network architecture to model latent features of users and items using collaborative filtering method. The NCF models are trained separately for each domain without transferring any information.

- **CMF[34]** Collective Matrix Factorization (CMF) simultaneously factor several matrices, sharing parameters among factors when a user participates in multiple domains.

Furthermore, we select hyperparamter α for our study as 0.03 using Bayesian Optimization. We use one-layer MLP for constructing encoder and decoder separately. The size of feature embedding is 8. For each baseline method, we set the same hyperparameters as our proposed model if applicable. The results are reported in the next section.

5 RESULTS

5.1 Cross-Domain Recommendation Performance

Since we have three different domains, i.e. books, movies and music, this results in three domain pairs for evaluation. The performance comparison results of DDTCDR with the baselines are reported using the experimental settings in Section 4.

As shown in Table 4, 5 and 6, the proposed DDTCDR model significantly and consistently outperforms all other baselines in terms of RMSE, MAE, Precision and Recall metrics across all the three domain pairs. As Table 3 shows, RMSE measures for book vs. movie domains obtained using the proposed model are 0.2213 and 0.2213, which outperform the second-best baselines by 3.98% and 2.44%; MAE measures are 0.1708 and 0.1704, which outperform the second-best baselines by 9.54% and 9.80%. We observe similar improvements in book vs. music domains where DDTCDR outperforms the second-best baselines by 4.07%, 8.87%, 2.14%, 4.74% and improvements in movie vs. music domains by 3.75%, 9.77%, 1.89%, 4.24% respectively. To summarize, all these results show that the dual transfer learning mechanism works well in practice, and the proposed DDTCDR model achieves significant cross-domain recommendation performance improvements. It is also worth noticing that DDTCDR only requires the same level of time complexity for training compared with other baseline models.

Furthermore, we observe that the improvements of proposed model on the book and movie domains are relatively greater than that of the music domain. One possible reason is the size of the dataset, for the book and movie domains are significantly larger than the music domain. We plan to explore this point further as a topic of future research.

5.2 Convergence

In Section 3.4, we proof convergence of the dual transfer learning method for classical matrix factorization problem. Note that however, this proposition is not applicable to the DDTCDR model because the optimization process is different. Therefore, we test the convergence empirically in our study, and conjecture should still happen even though our method does not satisfy the condition. We conduct the convergence study to empirically demonstrate the convergence.

In particular, we train the model iteratively for 100 epochs until the change of loss function is less than $1e-5$. We plot the training loss over time in Figure 2, 3 and 4 for three domain pairs. The key observation is that, DDTCDR starts with relatively higher loss due to the noisy initialization. As times goes by, DDTCDR manages to

Algorithm	Book				Movie			
	RMSE	MAE	Precision@5	Recall@5	RMSE	MAE	Precision@5	Recall@5
DDTCDR	0.2213*	0.1708*	0.8595*	0.9594*	0.2213*	0.1714*	0.8925*	0.9871*
Improved %	(+3.98%)	(+9.54%)	(+2.77%)	(+6.30%)	(+2.44%)	(+9.80%)	(+2.75%)	(+2.74%)
NCF	0.2315	0.1887	0.8357	0.8924	0.2276	0.1895	0.8644	0.9589
CCFNet	0.2639	0.1841	0.8102	0.8872	0.2476	0.1939	0.8545	0.9300
CDFM	0.2494	0.2165	0.7978	0.8610	0.2289	0.1901	0.8498	0.9312
CMF	0.2921	0.2478	0.7972	0.8523	0.2738	0.2293	0.8324	0.9012
CoNet	0.2305	0.1892	0.8328	0.8990	0.2298	0.1903	0.8680	0.9601

Table 4: Comparison of recommendation performance in Book-Movie Dual Recommendation: Improved Percentage versus the second best baselines

Algorithm	Book				Music			
	RMSE	MAE	Precision@5	Recall@5	RMSE	MAE	Precision@5	Recall@5
DDTCDR	0.2209*	0.1704*	0.8570*	0.9602*	0.2753*	0.2302*	0.8392*	0.8928*
Improved %	(+4.07%)	(+8.87%)	(+3.97%)	(+3.15%)	(+2.14%)	(+4.74%)	(+5.51%)	(+5.35%)
NCF	0.2315	0.1887	0.8230	0.9294	0.2828	0.2423	0.7930	0.8450
CCFNet	0.2630	0.1842	0.8150	0.9108	0.3090	0.2422	0.7902	0.8388
CDFM	0.2489	0.2155	0.8104	0.9102	0.3252	0.2463	0.7895	0.8365
CMF	0.2921	0.2478	0.8072	0.8978	0.3478	0.2698	0.7820	0.8324
CoNet	0.2307	0.1897	0.8230	0.9300	0.2801	0.2410	0.7912	0.8428

Table 5: Comparison of recommendation performance in Book-Music Dual Recommendation: Improved Percentage versus the second best baselines

Algorithm	Movie				Music			
	RMSE	MAE	Precision@5	Recall@5	RMSE	MAE	Precision@5	Recall@5
DDTCDR	0.2174*	0.1720*	0.8926*	0.9869*	0.2758*	0.2311*	0.8370*	0.8902*
Improved %	(+3.75%)	(+9.77%)	(+5.32%)	(+3.68%)	(+1.89%)	(+4.24%)	(+4.30%)	(+4.38%)
NCF	0.2276	0.1895	0.8428	0.9495	0.2828	0.2423	0.7970	0.8501
CCFNet	0.2468	0.1932	0.8398	0.9310	0.3090	0.2433	0.7952	0.8498
CDFM	0.2289	0.1895	0.8306	0.9382	0.3252	0.2467	0.7880	0.8460
CMF	0.2738	0.2293	0.8278	0.9222	0.3478	0.2698	0.7796	0.8400
CoNet	0.2302	0.1908	0.8450	0.9508	0.2811	0.2428	0.8010	0.8512

Table 6: Comparison of recommendation performance in Movie-Music Dual Recommendation: Improved Percentage versus the second best baselines

stabilize quickly and significantly outperforms NCF only after 10 epochs.

5.3 Sensitivity

To validate that the improvement gained in our model is not sensitive to the specific experimental setting, we conduct multiple experiments to examine the change of recommendation performance corresponding to different hyperparameter settings. As Figure 5, 6 and 7 show, we could observe certain amount of performance fluctuation with respect to transfer rate α . Note that the cross-domain recommendation results (when α take positive and small value) are consistently better than that of single-domain recommendations (when $\alpha = 0$). Also, we verify the recommendation performance using different types of autoencoders, including AE [35], VAE [8],

AAE [24], WAE [36] and HVAE [2], and the improvement of recommendation performance is consistent across these settings as shown in Table 7. The choice of particular autoencoder is not relevant to the recommendation performance.

6 CONCLUSION

In this paper, we propose a novel dual transfer learning based model that significantly improves recommendation performance across different domains. We accomplish this by transferring latent information from one domain to the other through embeddings and iteratively going through the transfer learning loop until the models stabilize for both domains. We also prove that this convergence is guaranteed under certain conditions and empirically validate the hypothesis for our model across different experimental settings.

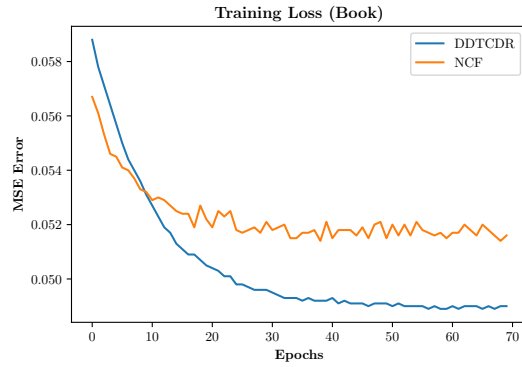


Figure 2: Book Domain Epoch-Loss

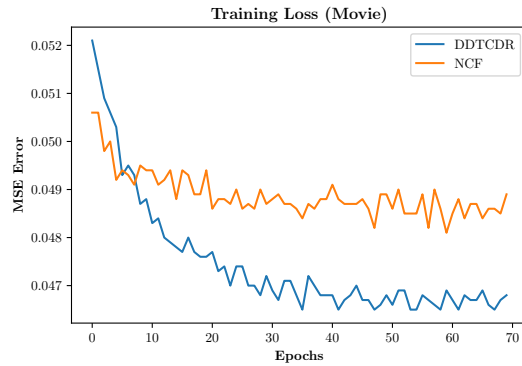


Figure 3: Movie Domain Epoch-Loss

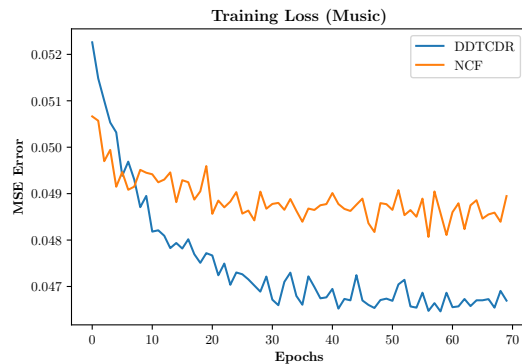


Figure 4: Music Domain Epoch-Loss

Note that, the proposed approach provides several benefits, including that it (a) transfers information about latent interactions instead of explicit features from the source domain to the target domain; (b) utilizes the dual transfer learning mechanism to enable the bidirectional training that improves performance measures for both domains simultaneously; (c) learns the latent orthogonal mapping function across two domains, that (i) preserves similarity of

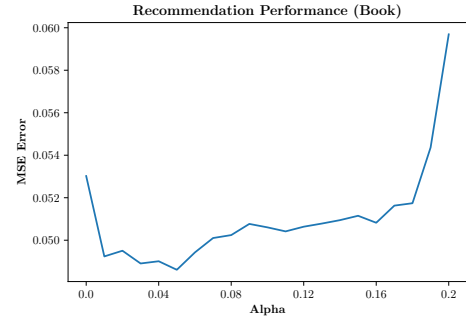


Figure 5: Recommendation Performance in Book Domain with Different Alpha Values

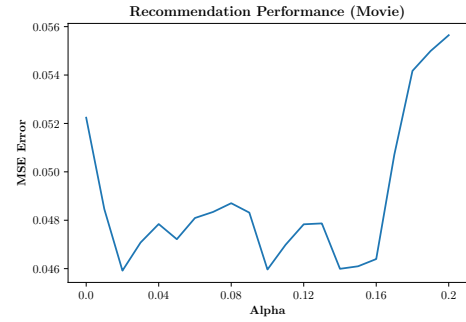


Figure 6: Recommendation Performance in Movie Domain with Different Alpha Values

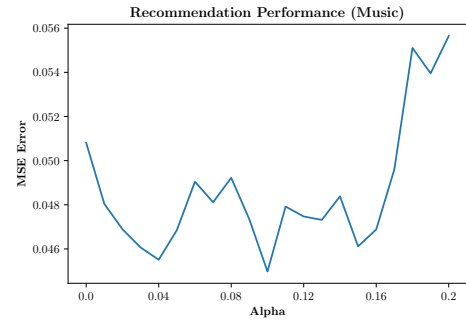


Figure 7: Recommendation Performance in Music Domain with Different Alpha Values

user preferences and thus enables proper transfer learning and (ii) computes the inverse mapping function efficiently.

As the future work, we plan to extend dual learning mechanism to multiple domains by simultaneously improving performance across all domains instead of domain pairs. We also plan to extend the convergence proposition to more general settings.

ACKNOWLEDGMENTS

The authors would like to thank Vladimir Bobrikov for providing the dataset for evaluation purposes.

Autoencoder	Book \longleftrightarrow Movie			
	RMSE	MAE	RMSE	MAE
AE	0.2213	0.1708	0.2213	0.1714
VAE	0.2240	0.1704	0.2196	0.1707
AAE	0.2236	0.1729	0.2195	0.1715
WAE	0.2236	0.1739	0.2202	0.1739
HVAE	0.2220	0.1717	0.2186	0.1704

Table 7: Comparison of Autoencoder Settings: Differences are not statistically significant

REFERENCES

- [1] Gediminas Adomavicius and Alexander Tuzhilin. 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge & Data Engineering* 6 (2005), 734–749.
- [2] Tim R Davidson, Luca Falorsi, Nicola De Cao, Thomas Kipf, and Jakub M Tomczak. 2018. Hyperspherical variational auto-encoders. *arXiv preprint arXiv:1804.00891* (2018).
- [3] Ignacio Fernández-Tobías, Iván Cantador, Marius Kaminskis, and Francesco Ricci. [n.d.]. Cross-domain recommender systems: A survey of the state of the art.
- [4] Di He, Yingce Xia, Tao Qin, Liwei Wang, Nenghai Yu, Tie-Yan Liu, and Wei-Ying Ma. 2016. Dual learning for machine translation. In *Advances in Neural Information Processing Systems*. 820–828.
- [5] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 173–182.
- [6] Guangneng Hu, Yu Zhang, and Qiang Yang. 2018. CoNet: Collaborative Cross Networks for Cross-Domain Recommendation. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. ACM, 667–676.
- [7] Liang Hu, Jian Cao, Guandong Xu, Longbing Cao, Zhiping Gu, and Can Zhu. 2013. Personalized recommendation via cross-domain triadic factorization. In *Proceedings of the 22nd international conference on World Wide Web*. ACM, 595–606.
- [8] Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).
- [9] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 8 (2009), 30–37.
- [10] Daniel D Lee and H Sebastian Seung. 2001. Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*. 556–562.
- [11] Bin Li, Qiang Yang, and Xiangyang Xue. 2009. Can movies and books collaborate? cross-domain collaborative filtering for sparsity reduction. In *Twenty-First International Joint Conference on Artificial Intelligence*.
- [12] Pan Li and Alexander Tuzhilin. 2019. Latent Modeling of Unexpectedness for Recommendations. (2019).
- [13] Pan Li and Alexander Tuzhilin. 2019. Latent multi-criteria ratings for recommendations. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 428–431.
- [14] Pan Li and Alexander Tuzhilin. 2019. Latent Unexpected and Useful Recommendation. *arXiv preprint arXiv:1905.01546* (2019).
- [15] Pan Li and Alexander Tuzhilin. 2019. Towards Controllable and Personalized Review Generation. *arXiv preprint arXiv:1910.03506* (2019).
- [16] Xiaopeng Li and James She. 2017. Collaborative variational autoencoder for recommender systems. In *Proceedings of the 23rd ACM SIGKDD*. ACM, 305–314.
- [17] Jianxun Lian, Fuzheng Zhang, Xing Xie, and Guangzhong Sun. 2017. CCCFNet: a content-boosted collaborative filtering neural network for cross domain recommender systems. In *Proceedings of the 26th international conference on World Wide Web companion*. International World Wide Web Conferences Steering Committee, 817–818.
- [18] Dawen Liang, Rahul G Krishnan, Matthew D Hoffman, and Tony Jebara. 2018. Variational autoencoders for collaborative filtering. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*. International World Wide Web Conferences Steering Committee.
- [19] Chih-Jen Lin. 2007. On the convergence of multiplicative update algorithms for nonnegative matrix factorization. *IEEE Transactions on Neural Networks* 18, 6 (2007), 1589–1596.
- [20] Jialu Liu, Chi Wang, Jing Gao, and Jiawei Han. 2013. Multi-view clustering via joint nonnegative matrix factorization. In *Proceedings of the 2013 SIAM International Conference on Data Mining*. SIAM.
- [21] Mingsheng Long, Jianmin Wang, Guiguang Ding, Wei Cheng, Xiang Zhang, and Wei Wang. 2012. Dual transfer learning. In *Proceedings of the 2012 SIAM International Conference on Data Mining*. SIAM.
- [22] Mingsheng Long, Jianmin Wang, Guiguang Ding, Jianguang Sun, and Philip S Yu. 2013. Transfer feature learning with joint distribution adaptation. In *Proceedings of the IEEE ICCV*.
- [23] Babak Loni, Yue Shi, Martha Larson, and Alan Hanjalic. 2014. Cross-domain collaborative filtering with factorization machines. In *European conference on information retrieval*. Springer, 656–661.
- [24] Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, Ian Goodfellow, and Brendan Frey. 2015. Adversarial autoencoders. *arXiv preprint arXiv:1511.05644* (2015).
- [25] Sinno Jialin Pan and Qiang Yang. 2010. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering* (2010).
- [26] Francesco Ricci, Lior Rokach, and Bracha Shapira. 2011. Introduction to recommender systems handbook. In *Recommender systems handbook*. Springer, 1–35.
- [27] Shaghayegh Sahebi and Peter Brusilovsky. 2013. Cross-domain collaborative recommendation in a cold-start context: The impact of user profile size on the quality of recommendation. In *International Conference on User Modeling, Adaptation, and Personalization*.
- [28] Shaghayegh Sahebi and Peter Brusilovsky. 2015. It takes two to tango: An exploration of domain pairs for cross-domain collaborative filtering. In *Proceedings of the 9th ACM RecSys*. ACM, 131–138.
- [29] Shaghayegh Sahebi, Peter Brusilovsky, and Vladimir Bobrov. 2017. Cross-domain recommendation for large-scale data. In *CEUR Workshop Proceedings*, Vol. 1887. 9–15.
- [30] Shaghayegh Sahebi and Trevor Walker. 2014. Content-Based Cross-Domain Recommendations Using Segmented Models.
- [31] Badrul Munir Sarwar, George Karypis, Joseph A Konstan, John Riedl, et al. 2001. Item-based collaborative filtering recommendation algorithms. *WWW* 1 (2001), 285–295.
- [32] Andrew I Schein, Alexandrin Popescul, Lyle H Ungar, and David M Pennock. 2002. Methods and metrics for cold-start recommendations. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 253–260.
- [33] Suvash Sedhain, Aditya Krishna Menon, Scott Sanner, and Lexing Xie. 2015. Autorec: Autoencoders meet collaborative filtering. In *Proceedings of the 24th International Conference on World Wide Web*. ACM.
- [34] Ajit P Singh and Geoffrey J Gordon. 2008. Relational learning via collective matrix factorization. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 650–658.
- [35] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*. 3104–3112.
- [36] Ilya Tolstikhin, Olivier Bousquet, Sylvain Gelly, and Bernhard Schölkopf. 2017. Wasserstein auto-encoders. *arXiv preprint arXiv:1711.01558* (2017).
- [37] Hua Wang, Heng Huang, Feiping Nie, and Chris Ding. 2011. Cross-language web page classification via dual knowledge transfer using nonnegative matrix tri-factorization. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. ACM, 933–942.
- [38] Hao Wang, Naiyan Wang, and Dit-Yan Yeung. 2015. Collaborative deep learning for recommender systems. In *Proceedings of the 21th ACM SIGKDD*. ACM, 1235–1244.
- [39] Yijun Wang, Yingce Xia, Li Zhao, Jiang Bian, Tao Qin, Guiguan Liu, and Tie-Yan Liu. 2018. Dual transfer learning for neural machine translation with marginal distribution regularization. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- [40] Hao Wu, Zhengxin Zhang, Kun Yue, Binbin Zhang, Jun He, and Liangchen Sun. 2018. Dual-regularized matrix factorization with deep neural networks for recommender systems. *Knowledge-Based Systems* 145 (2018), 46–58.
- [41] Yao Wu, Christopher DuBois, Alice X Zheng, and Martin Ester. 2016. Collaborative denoising auto-encoders for top-n recommender systems. In *Proceedings of the Ninth ACM WSDM*.
- [42] Yingce Xia, Tao Qin, Wei Chen, Jiang Bian, Nenghai Yu, and Tie-Yan Liu. 2017. Dual supervised learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*.
- [43] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. 2019. Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys (CSUR)* 52, 1 (2019), 5.
- [44] Erheng Zhong, Wei Fan, Jing Peng, Kun Zhang, Jiangtao Ren, Deepak Turaga, and Olivier Verscheure. 2009. Cross domain distribution adaptation via kernel mapping. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM.
- [45] Fuzhen Zhuang, Ping Luo, Zhiyong Shen, Qing He, Yuhong Xiong, Zhongzhi Shi, and Hui Xiong. 2010. Collaborative dual-plsa: mining distinction and commonality across multiple domains for text classification. In *Proceedings of the 19th ACM international conference on Information and knowledge management*. ACM, 359–368.