Domain Disentanglement with Interpolative Data Augmentation for Dual-Target Cross-Domain Recommendation

Jiajie Zhu
jiajie.zhu1@students.mq.edu.au
Macquarie University
Australia

Yan Wang*
yan.wang@mq.edu.au
Macquarie University
Australia

Feng Zhu
zhufeng.zhu@antgroup.com
Ant Group
China

Zhu Sun
sunzhuntu@gmail.com
Institute of High Performance Computing and Centre for
Frontier AI Research, A*STAR
Singapore

ABSTRACT

The conventional single-target Cross-Domain Recommendation (CDR) aims to improve the recommendation performance on a sparser target domain by transferring the knowledge from a source domain that contains relatively richer information. By contrast, in recent years, dual-target CDR has been proposed to improve the recommendation performance on both domains simultaneously. However, to this end, there are two challenges in dual-target CDR: (1) how to generate both relevant and diverse augmented user representations, and (2) how to effectively decouple domain-independent information from domain-specific information, in addition to domain-shared information, to capture comprehensive user preferences. To address the above two challenges, we propose a **D**isentanglement-based framework with **I**nterpolative **D**ata **A**ugmentation for dual-target **C**ross-**D**omain **R**ecommendation, called DIDA-CDR. In DIDA-CDR, we first propose an interpolative data augmentation approach to generating both relevant and diverse augmented user representations to augment sparser domain and explore potential user preferences. We then propose a disentanglement module to effectively decouple domain-specific and domain-independent information to capture comprehensive user preferences. Both steps significantly contribute to capturing more comprehensive user preferences, thereby improving the recommendation performance on each domain. Extensive experiments conducted on five real-world datasets show the significant superiority of DIDA-CDR over the state-of-the-art methods.

CCS CONCEPTS

• **Information systems** → **Recommender systems**; • **Computing methodologies** → **Neural networks**.

*Corresponding author.

KEYWORDS

Cross-Domain Recommendation, Data Augmentation, Disentangled Representation Learning

1 INTRODUCTION

To alleviate the data sparsity problem, Cross-Domain Recommendation (CDR) [32] aims to employ abundant information from a relatively richer domain to improve recommendation performance on a sparser domain, forming the so-called *single-target CDR* [57]. By contrast, in recent years, *dual-target CDR* has been proposed to improve the recommendation performance on both domains simultaneously by sharing the common knowledge across domains [50, 60].

The existing dual-target CDR methods can be divided into two groups, i.e., (1) conventional methods, (2) disentanglement-based methods. *Conventional methods* (see Fig. 1(a)) mainly utilize various transfer layers [18, 26, 29, 55] to integrate the representations learned by two base encoders in their respective domains. In contrast, *disentanglement-based methods* (see Fig. 1(b)) tend to use the variational autoencoder (VAE) [3] or other disentangling techniques [4, 14, 53] to decouple the domain-shared and domain-specific information, and only transfer the domain-shared information to each domain, which enhances the recommendation accuracy on both domains simultaneously.

However, the existing dual-target CDR models have limitations in terms of effectively capturing comprehensive user preferences for the following reasons. Firstly, even though some of them use data augmentation to mitigate data imbalance between richer and sparser domains, few of them can balance the relevance (being able to represent user preferences in both domains) and diversity (having enough variations in user representations) of the user-item interaction augmentation. As a result, such augmented user representations can hardly provide strong support for subsequently capturing accurate and comprehensive user preferences. Secondly,

Figure 1: (a)-(b): Existing dual-target CDR models. (c): A motivating example of user preferences in movie and music domains. (d): Compared with existing dual-target CDR models, our proposed DIDA-CDR not only decouples domain-shared user preferences for cross-domain knowledge transfer, but also takes domain-independent user preferences into consideration.

none of the existing methods decouple all three essential components needed to capture comprehensive user preferences. To be specific, the existing methods only decouple two essential components, i.e., domain-shared and domain-specific information, and ignore the existence of domain-independent information. Since domain-independent information has different meanings from the other two types of information, it cannot be ignored when capturing comprehensive user preferences. It is worth mentioning here that some existing works also decouple so called 'domain-independent information', but its meaning is different from that in our work (cf. the detailed description in Section 2.1).

Below we introduce domain-shared, domain-specific and domain-independent information respectively with examples, and further differentiate them.

(i) **Domain-shared Information:** As shown in Fig. 1(c), there exists some *domain-shared* information in both movie domain and music domain, such as 'Category'. For example, people who like watching *suspense movies* (i.e., a category in movie domain) tend to like listening to *suspense music* (i.e., a category in music domain), and vice versa. Since the domain-shared information can provide the valuable information for cross-domain recommendations, it needs to be first decoupled and then transferred to both domains.

(ii) **Domain-specific Information:** In contrast, there also exists some *domain-specific* information in each domain. For example, the user preference for pictures in a movie (i.e., 'Frame' in movie domain) is not applicable in music domain because 'Frame' is unique to movie domain. Thus, such domain-specific information should be decoupled too, which helps improve the recommendation performance on its own domain, but it should not be

transferred to another domain in order to avoid the negative transfer [61].

(iii) **Domain-independent Information:** In addition, some *domain-independent* information also exists in each domain, but should not be transferred to other domains. For instance, 'Rhythm' exists in each of movie, music and book domains. However, in movie domain, it means the use of sound effects, the speed of camera cuts, and the changes in the pace of movie scenes, etc [38]. In music domain, it means , etc [15]. In book domain, it means the fluidity of the writing and the ups and downs of the storyline, etc [12]. In other words, although 'Rhythm' is seemingly common in all three domains, it has different meanings in different domains. Hence, such information is domain-independent and should be extracted from its own domain for capturing comprehensive user preferences, but should not be transferred to other domains.

(iv) **Difference:** Different to the domain-independent information, the domain-shared information (e.g., 'Category') extracted from two domains expresses the same meaning in each domain. In addition, the domain-specific information, e.g., 'Frame' in the movie domain, exists only in its own domain. By contrast, although the domain-independent information, e.g., 'Rhythm', exists in each domain, it has different meanings in different domains.

(v) **Summary:** Therefore, it is vital to recognize the existence of domain-independent information and to clearly differentiate it from domain-shared and domain-specific information. More importantly, decoupling domain-independent information is crucial for capturing more comprehensive user preferences, otherwise it will cause suboptimal recommendation results. However, existing dual-target CDR methods neglect the above insights. Hence, novel dual-target CDR solutions are needed to incorporate the

above insights for capturing more comprehensive user preferences.

Following the above discussions, to target superior dual-target CDR, there are two major challenges as follows.

CH1: *How to generate relevant and diverse augmented user representations to augment sparser domain and explore potential user preferences?* The existing dual-target CDR methods [14, 48] mainly obtain augmentation views by perturbing the original data. Although these methods can increase the diversity of augmentation by increasing the types of perturbations, they cannot generate augmented user representations that represent user preferences in both domains because they perform data augmentation on the data from each domain separately without considering the correlation between data from both domains, and thus can hardly capture the potential common user preferences in both domains.

CH2: *How to effectively decouple domain-independent information from domain-specific information, in addition to domain-shared information, to capture comprehensive user preferences on each domain, thereby improving the recommendation performance on both domains?* The existing dual-target CDR methods either ignore decoupling the domain-specific and domain-shared information [18] (*Group 1*), or directly transfer the domain-shared information to only fuse with the domain-specific information in each domain, overlooking the domain-independent information [3] (*Group 2*). The methods in *Group 1* (also see Fig. 1(a)) disregard the discrimination between domain-specific and domain-shared information, which may lead to the negative transfer. The methods in *Group 2* (also see Fig. 1(b)) do not differentiate the domain-independent information from domain-specific information and decouple these two types of information, which results in suboptimal recommendation results.

Our Approach and Contributions: To address the above challenges, we propose a novel disentanglement-based framework with interpolative data augmentation for dual-target CDR. To the best of our knowledge, this is the first work in the literature that explicitly takes domain-independent information into consideration in addition to domain-shared and domain-specific information, and decouples it to capture more comprehensive user preferences for cross-domain recommendation. The characteristics and contributions of our proposed model are summarized as follows:

- We first propose a **D**isentanglement-based framework with **I**nterpolative **D**ata **A**ugmentation for dual-target **C**ross-**D**omain **R**ecommendation, called DIDA-CDR, which can augment the sparser domain, disentangle three essential components of user preferences and transfer the domain-shared user preferences of common users across domains, thus enhancing the recommendation accuracy on both domains simultaneously;

- To address **CH1**, we propose an interpolative data augmentation approach to generating both relevant and diverse augmented user representations, which augments the sparser domain and explores the potential common user preferences and therefore improves the recommendation performance on both domains;

- To address **CH2**, we propose a disentanglement module to effectively decouple the domain-independent and domain-specific user preferences. The disentanglement module also

extracts the domain-shared user preferences from augmented user representations, which can be transferred to both domains to provide the valuable information. We then apply the attention mechanism to combine the above three essential components of user preferences to capture more comprehensive user preferences in each domain, which can improve the recommendation performance on each of both domains;

- Extensive experiments conducted on five real-world datasets show that our proposed DIDA-CDR outperforms the best-performing state-of-the-art baselines with an average improvement of 8.54% and 11.10% with respect to HR@10 and NDCG@10, respectively.

2 RELATED WORK

In this section, we first review the relevant literature on two major categories of CDR in Section 2.1. Next, since we utilize disentangled representation learning and interpolative data augmentation for our model, we also review the relevant literature on them in Section 2.2 and Section 2.3, respectively [57].

2.1 Single-Target and Dual-Target CDR

Single-Target CDR. The existing single-target CDR methods can be divided into two categories, i.e., (1) content-based transfer methods and (2) feature-based transfer methods [60]. Content-based transfer methods [21] mainly use the various content information, such as user/item attributes, tags, reviews, etc., to link domains and share their information across domains. By contrast, feature-based transfer methods [9, 19] aim to first get embeddings or rating patterns [49] using various learning techniques, and then transfer them across domains.

Dual-Target CDR. In contrast to single-target CDR, dual-target CDR aims to achieve better recommendation performance on both domains simultaneously, which can be extended to multiple domains, leading to multi-target CDR [8, 14, 59]. The existing dual-target CDR approaches can be divided into two categories, i.e., (1) conventional approaches, (2) disentanglement-based approaches. For *conventional dual-target CDR*, CoNet [18] utilizes cross-connection networks to achieve knowledge transfer between two domains. DDTCDR [26] employs a latent orthogonal mapping function to transfer user embeddings across domains. PPGN [55] constructs a cross-domain interaction graph to learn and transfer representations. BiTGCF [29] first leverages two base graph encoders to learn user/item embeddings, and then performs the feature propagation and transfer to fuse user embeddings. However, most of them lead to the negative transfer, because they neglect to decouple the domain-shared and domain-specific information.

In addition, we further classify the existing *disentanglement-based dual-target CDR approaches* into two classes, i.e., (1) VAE-based approaches and (2) other approaches, according to disentangling techniques. VAE-based approaches [43] (*Class 1*) employ the explicit reconstruction loss included in the Evidence Lower Bound (ELBO) and extra regularizers as the disentanglement loss to learn desirable disentangled representations. Other approaches (*Class 2*) often utilize Graph Convolutional Networks (GCNs) [41], adversarial learning [7, 56], self-supervised learning [14, 25, 52]

Table 1: Important notations.

Symbol	Definition	Symbol	Definition
k	the dimension of embedding matrix	P	the predicted domain probability
m	the number of users	O	the ground truth domain label
n	the number of items	$\mathbf{E}_u$	the graph embedding matrix of users
R	the rating matrix	$\mathbf{E}_v$	the graph embedding matrix of items
$\mathcal{U}$	the set of users	$\mathbf{Z}_{sha}$	the domain-shared user preferences
$\mathcal{V}$	the set of items	$\mathbf{Z}_{spe}$	the domain-specific user preferences
y	the user-item interaction	$\mathbf{Z}_{ind}$	the domain-independent user preferences
$*^{aug}$	the notation for data augmentation	$\mathbf{E}_u^*$	the comprehensive user preferences
$*^A, *^B$	the notations for domains A and B, e.g., n^A represents the number of items on domain A	$\mathbf{E}_u^{aug}$	the augmented user representations of common users
$\hat{*}$	the predicted notations, e.g., $\hat{y}_{ij}$ represents the predicted interaction of user u_i on item v_j	G	the heterogeneous graph, where Q is the set of user-item relationships

and fixed or flexible combination strategies [48, 57, 59] to disentangle the latent knowledge, such as domain-shared and domain-specific information. The domain-shared information is also termed as domain-invariant information in [31, 39], domain-common information in [7] and domain-independent information in [28, 37] (note that the domain-independent information in other works does not have the same meaning as in our work). However, none of them can effectively capture comprehensive user preferences, because they entangle the domain-independent information with domain-specific information.

2.2 Disentangled Representation Learning

Disentangled representation learning is originally introduced in the field of computer vision [6, 13] and is mainly used to learn visual features such as shape, color and location features of objects [54]. In addition to computer vision, in recent years, disentangled representation learning has also been used in recommender systems (RSs) [33, 34, 46]. The main idea of disentangled representation learning is to focus on decomposing the latent factors behind the observed instances in the low-dimension vector space [1, 42]. For recommendation, MacridVAE [33] performs the macro disentanglement and micro disentanglement based on the user behavior data. DGCF [46] leverages a graph disentangling module to decouple user embeddings learned from user-item interaction data into fine-grained user intents. The work proposed in [34] reconstructs the embedding of future sequence by self-supervised learning to decouple the intentions of users.

2.3 Interpolative Data Augmentation

Mixup [51] has been recently proposed as a representative interpolative data augmentation method that enhances the prediction accuracy of networks for image classification. In addition to image classification, interpolative data augmentation has been used in recent RSs [5, 20]. The general idea of interpolative data augmentation is to linearly interpolate inputs and model targets of random samples. For recommendation, existing methods with interpolative data augmentation [2] tend to generate natural augmented user representations to model sequential user behaviors [17].

3 THE PROPOSED MODEL

In this section, first, we formulate the target problem of our proposed model. Next, we present a novel **D**isentanglement-based framework with **I**nterpolative **D**ata **A**ugmentation for dual-target **C**ross-**D**omain **R**ecommendation, called DIDA-CDR. Finally, we elaborate on the basic components of DIDA-CDR.

3.1 Problem Formulation

For the sake of better readability, we list the important notations of this paper in Table 1. This paper considers the dual-target CDR on two domains D^A and D^B with a shared user set, denoted by $\mathcal{U}$ (of size $m = |\mathcal{U}|$). The sets of items in D^A and D^B are defined as $\mathcal{V}^A$ (of size $n^A = |\mathcal{V}^A|$) and $\mathcal{V}^B$ (of size $n^B = |\mathcal{V}^B|$), respectively. Let $\mathbf{R}^A \in \{0,1\}^{m \times n^A}$ and $\mathbf{R}^B \in \{0,1\}^{m \times n^B}$ denote the binary user-item interaction matrices in D^A and D^B, respectively. By aggregating the interaction data in each domain, we first construct two heterogeneous graphs $G^A = (\mathcal{U}, \mathcal{V}^A, Q^A)$ and $G^B = (\mathcal{U}, \mathcal{V}^B, Q^B)$ to learn user embeddings $\mathbf{E}_u^A$, $\mathbf{E}_u^B$ and item embeddings $\mathbf{E}_v^A$, $\mathbf{E}_v^B$ in domains D^A and D^B, respectively, where Q^A and Q^B are the edge sets that represent the observed user-item interactions. By linearly interpolating the user embeddings $\mathbf{E}_u^A$ and $\mathbf{E}_u^B$, we then generate augmented user representations $\mathbf{E}_u^{aug}$ to augment the sparser domain D^B. Given two coarse user embeddings $\mathbf{E}_u^A$, $\mathbf{E}_u^B$ and the augmented user representations $\mathbf{E}_u^{aug}$, our goal is to disentangle domain-shared, domain-specific and domain-independent user preferences, i.e., $\mathbf{Z}_{sha}$, $\mathbf{Z}_{spe}$, and $\mathbf{Z}_{ind}$, and then transfer domain-shared user preferences $\mathbf{Z}_{sha}$ to each domain to capture comprehensive user preferences $\mathbf{E}_u^*$, thereby improving the recommendation performance on both domains.

3.2 Overview of DIDA-CDR

To enhance the recommendation accuracy on both domains, we propose a novel <u>di</u>sentanglement-based framework with <u>i</u>nterpolative <u>d</u>ata <u>a</u>ugmentation for dual-target <u>c</u>ross-<u>d</u>omain <u>r</u>ecommendation, called DIDA-CDR. This framework contains five major components, i.e., (1) *Graph Convolution and Propagation Module*, (2) *Interpolative Data Augmentation Module*, (3) *User Preference Disentanglement Module*, (4) *Information Fusion Module*, and (5) *Prediction Module*.

Figure 2: The details of each module of our proposed DIDA-CDR.

The details of each module of our proposed DIDA-CDR are illustrated in Fig. 2. They are briefly introduced below and described in detail in the following subsections.

(1) Graph Convolution and Propagation. First, we construct two heterogeneous graphs to extract the high-order user-item interaction relationships using the interaction data in domain *A* and domain *B*, respectively. Based on the above graphs, we apply the graph convolution and propagation layer in the GCN [23] to generate user and item embeddings.

(2) Interpolative Data Augmentation. Next, we propose an interpolative data augmentation approach to augmenting user embeddings at the representation level. The interpolative data augmentation approach linearly interpolates user embeddings in domain *A* and domain *B* to generate both relevant and diverse augmented user representations.

(3) User Preference Disentanglement. Thereafter, we propose a disentanglement module guided by a domain classifier to decouple more accurate domain-specific and domain-independent user preferences from user embeddings generated by GCNs. This module also disentangles the domain-shared user preferences from the augmented user representations, which are then transferred to both domains to provide the valuable information.

(4) Information Fusion. Next, we use three approaches, i.e., concatenation, element-wise sum, and attention mechanism to incorporate essential components of user preferences, i.e., domain-shared, domain-specific and domain-independent information, which are decoupled by the disentanglement module, to capture comprehensive user preferences.

(5) Prediction. Finally, we apply the multi-layer perception (MLP) to model the user-item interaction relationships, endowing the nonlinearity to our proposed model. Based on the MLP, the predicted user-item interaction matrix can be obtained. The prediction loss between it and the observed user-item interaction matrix, together with two domain classification losses, constitute the final loss for training.

Overall, our model can be easily extended to a multi-target CDR model. Specifically, we can disentangle domain-shared user preferences and transfer them to all domains, decouple the domain-specific and domain-independent user preferences from each domain, and then capture the comprehensive user preferences to improve the recommendation performance on multiple domains simultaneously.

3.3 Graph Convolution and Propagation

GCNs excel at capturing the relationships between nodes and learning the representation of graph data, and are well suited for modeling user-item relationships in RSs, because the user-item interaction data can be easily transformed into the graph structure. To distill the high-order user-item interaction relationships, we construct two heterogeneous graphs G^A and G^B for domains *A* and *B*, respectively, where nodes refer to entities (i.e., users and items) and edges refer to interactions. In this paper, we apply the graph convolution and propagation layer in the GCN to encode the user and item embeddings according to the user-item interaction matrix $\mathbf{R}^A$ (or $\mathbf{R}^B$). The node embeddings $\mathbf{E}_0^A$ (or $\mathbf{E}_0^B$) are randomly initialized. Given a

graph G^A, the propagation rule is represented as:

$$\mathbf{E}_l^A = f(\tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{R}}^A \tilde{\mathbf{D}}^{-\frac{1}{2}} \mathbf{E}_{l-1}^A \mathbf{W}_l + \mathbf{b}_l), \tag{1}$$

where $\tilde{\mathbf{R}}^A = \mathbf{R}^A + \mathbf{I}$ is the user-item interaction matrix of graph G^A after adding a self-loop identity matrix $\mathbf{I}$. $\tilde{\mathbf{D}}$ is a degree matrix for normalization. $\mathbf{W}_l$ and $\mathbf{b}_l$ are the trainable weight matrix and bias vector in the lth layer respectively, and $\mathbf{E}_l^A$ is the hidden embedding matrix of graph G^A in the lth layer [36]. $f(\cdot)$ denotes the ReLU activation function.

After l times propagation, we can obtain the global hidden representations $\mathbf{E}^A$ by concatenating multiple embedding matrices from $\mathbf{E}_0^A$ to $\mathbf{E}_l^A$, which can be rearranged into the user embeddings $\mathbf{E}_u^A$ and item embeddings $\mathbf{E}_v^A$ in domain A [55]. Similarly, we can obtain the user embeddings $\mathbf{E}_u^B$ and item embeddings $\mathbf{E}_v^B$ in domain B.

3.4 Interpolative Data Augmentation

Although the user embeddings in both domains are obtained by GCNs, the user embeddings learned from the sparser domain are not as accurate as those learned from the richer domain due to the data imbalance between the two domains. To augment the sparser domain and explore the potential user preferences, inspired by the mixup technique [51], we design an interpolative data augmentation approach, which generates both relevant and diverse augmented user representations. However, the conventional mixup technique cannot be directly utilized for our task, because the interaction data cannot be directly mixed at the pixel level like images. Therefore, we propose to linearly interpolate the embeddings of common users in both domains. The augmentation of common users ensures that augmented user representations maintain the relevance of user preferences in both domains, i.e., relevant augmented user representations, while corresponding to adding more interaction data for users in the sparser domain. In addition, introducing randomness in linear interpolation instead of using a fixed mixing coefficient can generate diverse augmented user representations, which can provide richer information for subsequent disentanglement. The formula for interpolative data augmentation can be expressed as follows:

$$\mathbf{E}_u^{aug} = \lambda \mathbf{E}_u^A + (1 - \lambda) \mathbf{E}_u^B, \tag{2}$$

where $\mathbf{E}_u^{aug}$ denotes the augmented user representations[1]. Since we aim to generate more diverse augmented user representations through data augmentation, instead of using attention-based methods in this module, we propose the interpolative data augmentation approach. $\lambda \in [0, 1]$ is the mixing coefficient sampled from $Beta(\alpha, \alpha)$, $\alpha \in (0, \infty)$. The advantages of adopting a mixing coefficient sampled from $Beta(\alpha, \alpha)$ are as follows. First, since the user embeddings of the common user in each of the two domains should be equivalent, the mixing coefficient should be sampled within the interval $[0, 1]$ and be symmetric around 0.5. Beta distribution $Beta(\alpha, \alpha)$ satisfies this characteristic. Second, it has been proven to effectively improve the generalization ability of the model [47]. Meanwhile, λ also introduces randomness into our model, thus weakening the negative transfer that may result from performing

a linear interpolation operation with fixed weights (cf. the experimental results in Section 4.3 and the discussion of parameter α in Section 4.6.2).

3.5 User Preference Disentanglement

To capture domain comprehensive preferences of users, we utilize disentangled representation learning to extract the essential components of user preferences from previously obtained user embeddings. Inspired by the method introduced in [11], we design a disentanglement module to decouple the domain-specific and domain-independent user preferences. Specifically, our disentanglement module adopts a similar architecture to the encoder of VAE, but it is quite different from VAE. In particular, VAE only encodes one latent feature, while our disentanglement module aims to better learn to decouple the domain-specific and domain-independent user preferences. Although both domain-specific and domain-independent user preferences cannot be transferred to other domains, they have different meanings, thus their importance in capturing comprehensive user preferences is different. If they are not distinguished, they are equally important in capturing comprehensive user preferences, which is inappropriate. By contrast, if they are decoupled, the subsequent information fusion module can use attention mechanism to learn their weights respectively, thereby capturing more accurate and comprehensive user preferences. This module can also disentangle the domain-shared user preferences from the augmented user representations. In this case, our model can not only extract the domain-shared preferences of common users in both domains, but also explore the domain-specific personalized user preferences and domain-independent user preferences, which enhances the comprehensiveness of capturing user preferences, and thus improves the recommendation performance on both domains. To this end, we first feed user embeddings in both domains (see methods introduced in Section 3.3) and augmented user representations (see methods introduced in Section 3.4) into this module and perform the following processing respectively.

Taking domain A as an example, the user embeddings $\mathbf{E}_u^A$ are first entered into the disentanglement module, which consists of several fully connected (FC) layers. More specifically, the first FC layer (see navy blue FC layer in Fig. 2), followed by the ReLU activation function, is utilized to extract general representations. In addition, the subsequent FC layers (also see lake blue FC layers in Fig. 2) outputs two sets of latent vectors, each representing different mean and deviation information of the input user embeddings. Next, following the method introduced in [11], a reparametrization trick is adopted to generate domain-independent $\mathbf{Z}_{ind}^A$ and domain-specific user preferences $\mathbf{Z}_{spe}^A$. Similarly, we can obtain the domain-independent $\mathbf{Z}_{ind}^B$ and domain-specific user preferences $\mathbf{Z}_{spe}^B$ in domain B, as well as the domain-shared $\mathbf{Z}_{sha}^{aug}$ and domain-specific user preferences $\mathbf{Z}_{spe}^{aug}$ decoupled from augmented user representations.

To ensure that the above essential components of user preferences can be accurately disentangled, we introduce a domain classifier $H_{cls}(\cdot)$, which includes a single FC layer to predict the domain probability of user preferences, to guide the disentanglement process. In order to supervise the optimization process of disentanglement module, we further set two domain classification

---

[1]The two terms, i.e., embedding and representation, are exchangeable in this paper.

tasks to train our DIDA-CDR. First of all, the disentanglement module is guided to decouple the domain-specific information with stronger domain identification ability, i.e., more accurate domain-specific user preferences, by minimizing the domain classification loss $\mathcal{L}_{cls_1}$. In other words, if the decoupled user preferences can be easily recognized by a domain classifier as belonging to a particular domain, then such user preferences are considered as the domain-specific information. By contrast, the domain-independent and domain-shared information are leveraged to confuse the domain classifier, i.e., to make the domain classifier unable to identify the domain to which they belong, to ensure that they can be distinguished from the domain-specific information. In other words, if the decoupled user preferences are no better than random guesses in identifying the domain to which they belong when fed into a domain classifier, then such user preferences are mutually exclusive with the domain-specific information and should be classified as the domain-independent information. When the input to the disentanglement module is the augmented user representations containing the user preferences from both domains, the above user preferences refer to the domain-shared information. Specifically, the domain classification loss $\mathcal{L}_{cls_1}$ can be defined as follows:

$$
\begin{aligned}
\mathcal{L}_{cls_1} = \frac{1}{3} \sum \big[&\ell_1(H_{cls}(\mathbf{Z}_{spe}^A), O_{spe}^A) \\
&+ \ell_1(H_{cls}(\mathbf{Z}_{spe}^B), O_{spe}^B) \\
&+ \lambda \cdot \ell_1(H_{cls}(\mathbf{Z}_{spe}^{aug}), O_{spe}^{augA}) \\
&+ (1 - \lambda) \cdot \ell_1(H_{cls}(\mathbf{Z}_{spe}^{aug}), O_{spe}^{augB}) \big],
\end{aligned}
\tag{3}
$$

where $\ell_1(P, O)$ denotes the cross-entropy loss function. P is the predicted domain probability of input user preferences and O is the ground truth (GT) domain label. We define the corresponding GT for domain-specific user preferences $\mathbf{Z}_{spe}^A$, $\mathbf{Z}_{spe}^B$ and $\mathbf{Z}_{spe}^{aug}$ as O_{spe}^A, O_{spe}^B and O_{spe}^{aug}, respectively. All the elements in O_{spe}^A and O_{spe}^{augA} are set to 1, while those in O_{spe}^B and O_{spe}^{augB} are set to 0. Here, λ represents the confidence score that an augmented user representation belongs to its initial GT, since augmented user representations are generated by incorporating embeddings of common users in both domains with a mixing coefficient λ [11].

Similarly, we define the corresponding GT for domain-independent user preferences $\mathbf{Z}_{ind}^A$, $\mathbf{Z}_{ind}^B$ and domain-shared user preferences $\mathbf{Z}_{sha}^{aug}$ as O_{ind}^A, O_{ind}^B and O_{sha}^{aug}, respectively. We set all the items in O_{ind}^A, O_{ind}^B and O_{sha}^{aug} as [0.5, 0.5] [10]. This ensures that the learned domain-shared and domain-independent user preferences cannot be used to identify the domain to which they belong, and thus be distinguished from the domain-specific user preferences. Specifically, the loss function is represented as follows:

$$
\begin{aligned}
\mathcal{L}_{cls_2} = \frac{1}{3} \sum \big[&\ell_2(H_{cls}(\mathbf{Z}_{ind}^A), O_{ind}^A) \\
&+ \ell_2(H_{cls}(\mathbf{Z}_{ind}^B), O_{ind}^B) \\
&+ \ell_2(H_{cls}(\mathbf{Z}_{sha}^{aug}), O_{sha}^{aug}) \big],
\end{aligned}
\tag{4}
$$

where $\ell_2(P, O)$ is the Kullback-Leibler divergence loss function.

Table 2: Comparison of information fusion approaches [40].

	Formula
	Formula
Concatenation	$\mathbf{E}_u^* = [\mathbf{Z}_{spe}, \mathbf{Z}_{ind}, \mathbf{Z}_{sha}]$
Element-wise sum	$\mathbf{E}_u^* = \mathbf{Z}_{spe} + \mathbf{Z}_{ind} + \mathbf{Z}_{sha}$
Attention	$\mathbf{C}_u = Softmax(\mathbf{W}_s \cdot \sigma(\mathbf{W}_{spe} \cdot \mathbf{Z}_{spe} + \mathbf{W}_{ind} \cdot \mathbf{Z}_{ind} + \mathbf{W}_{sha} \cdot \mathbf{Z}_{sha}))$ $\mathbf{E}_u^* = [\mathbf{Z}_{spe}, \mathbf{Z}_{ind}, \mathbf{Z}_{sha}] \cdot \mathbf{C}_u$

3.6 Information Fusion

The domain-shared, domain-specific and domain-independent information are three essential components of user preferences, which need to be integrated in a reasonable and efficient way to capture comprehensive user preferences. To this end, in this paper, we leverage three approaches, i.e., concatenation, element-wise sum, and attention mechanism, to aggregate individual representations into comprehensive user preferences [40]. The specific operations of these information fusion approaches are expressed in Table 2. (cf. the experimental comparison of them in Section 4.5).

3.7 Model Prediction and Training

After the information fusion, we obtain the comprehensive user preferences $\mathbf{E}_u^*$ and we also have the corresponding item embeddings $\mathbf{E}_v$ generated by GCNs. To give our model the non-linearity, we adopt a neural network, i.e., MLP, to represent the user-item interactions. Taking the domain A as an example, the input user embeddings and item embeddings in domain A for the MLP are defined as $\mathbf{S}_{in}^A = \mathbf{E}_u^{A*}$ and $\mathbf{T}_{in}^A = \mathbf{E}_v^A$, respectively. Moreover, the output embeddings of user u_i and item v_j of MLP is expressed as:

$$
\mathbf{S}_i^A = \mathbf{S}_{out_i}^A = \delta(\ldots \delta(\delta(\mathbf{S}_{in_i}^A \cdot W_{\mathbf{S}_1}^A) \cdot W_{\mathbf{S}_2}^A)),
\tag{5}
$$

$$
\mathbf{T}_i^A = \mathbf{T}_{out_i}^A = \delta(\ldots \delta(\delta(\mathbf{T}_{in_i}^A \cdot W_{\mathbf{T}_1}^A) \cdot W_{\mathbf{T}_2}^A)),
\tag{6}
$$

where $\delta(\cdot)$ is the LeakyReLU activation function. $\mathbf{W}_{\mathbf{S}_1}^A$, $\mathbf{W}_{\mathbf{S}_2}^A \ldots$ and $\mathbf{W}_{\mathbf{T}_1}^A$, $\mathbf{W}_{\mathbf{T}_2}^A \ldots$ denote the trainable weight matrices of MLP in various layers, respectively.

Next, the predicted user-item interaction $\hat{y}_{ij}^A$ between user u_i and item v_j in domain A can be formulated as follows:

$$
\hat{y}_{ij}^A = cosine(\mathbf{S}_i^A, \mathbf{T}_j^A) = \frac{\mathbf{S}_i^A \cdot \mathbf{T}_j^A}{\|\mathbf{S}_i^A\| \|\mathbf{T}_j^A\|}.
\tag{7}
$$

Furthermore, the prediction loss in domain A is defined as follows:

$$
\mathcal{L}_{prd}^A = \sum_{y \in \mathcal{Y}^{A+} \cup \mathcal{Y}^{A-}} \ell_1(\hat{y}, y) + \gamma(\|\mathbf{S}^A\|_F^2 + \|\mathbf{T}^A\|_F^2),
\tag{8}
$$

where $\ell_1(\hat{y}, y)$ is the cross-entropy loss function. y denotes an observed user-item interaction, and $\hat{y}$ is the corresponding predicted user-item interaction. $\mathcal{Y}^{A+}$ denotes the set of observed interactions and $\mathcal{Y}^{A-}$ is a certain number of negative instances randomly sampled from the set of unseen interactions in domain A to avoid over-fitting. $\|\mathbf{S}^A\|_F^2 + \|\mathbf{T}^A\|_F^2$ is a regularizer controlled by γ.

Finally, we utilize a multi-task learning mechanism consisting of a prediction task and two domain classification tasks to optimize our model in domain A. Specifically, the final loss function is formulated as follows:

$$
\mathcal{L}^A = \mathcal{L}_{prd}^A + \mu_1 \cdot \mathcal{L}_{cls_1} + \mu_2 \cdot \mathcal{L}_{cls_2}.
\tag{9}
$$

where μ_1 and μ_2 denote the weights of domain classification losses $\mathcal{L}_{cls_1}$ and $\mathcal{L}_{cls_2}$, respectively (cf. the discussion of parameters μ_1 and μ_2 in Section 4.6.3). Similarly, we perform the same optimization process for domain B.

4 EXPERIMENTS AND ANALYSIS

In order to demonstrate the superiority of our proposed DIDA-CDR and explore the effectiveness of its various modules, we conduct extensive experiments on five real-world datasets to answer the following five research questions:

- **RQ1.** How does our model perform when compared to representative and state-of-the-art baseline models?
- **RQ2.** How do various modules (i.e., interpolative data augmentation and user preference disentanglement) affect the results of our model?
- **RQ3.** How do various components of user preferences (i.e., domain-shared, domain-specific and domain-independent information) contribute to the performance improvement of our model?
- **RQ4.** How do various information fusion approaches influence the performance of our models?
- **RQ5.** How does the performance of our model change with various values of hyper-parameters?

4.1 Experimental Settings

4.1.1 Experimental Datasets and Tasks. In order to verify the recommendation performance of our proposed DIDA-CDR, we conduct extensive experiments on five real-world datasets, i.e., Douban subsets (Douban-Movie, Douban-Book and Douban-Music) released in GA-DTCDR [59] and Amazon subsets (Amazon-Elec and Amazon-Cloth) released in DisenCDR [3]. For these five datasets, we first convert the explicit ratings into implicit feedback, i.e., we binarize the ratings into 0 and 1 to indicate whether the user has interacted with the item or not. Following the methods introduced in [29, 30], we then filter these datasets to remove users and items with less than 5 interactions. Since the two Amazon subsets are filtered out of the cold-start item entry in the test set, following DisenCDR [3], we perform the same preprocessing operation on the three Douban subsets as well for a fair comparison. Finally, we divide the above five subsets into three pairs of datasets, extract the common users in each pair of datasets, and design three dual-target CDR tasks in a scenario where users completely overlap, which can be listed as follows:

- **Task 1:** Douban-Movie (richer) ↔ Douban-Book (sparser)
- **Task 2:** Douban-Movie (richer) ↔ Douban-Music (sparser)
- **Task 3:** Amazon-Elec (richer) ↔ Amazon-Cloth (sparser)

The details of these three dual-target CDR tasks and the corresponding datasets are shown in Table 3.

4.1.2 Parameter Settings. For the graph convolution and propagation module in Fig. 2, the layer structure of GCN is '$k \rightarrow k$' and for the disentanglement module, the layer structure is '$2k \rightarrow k$'. In the prediction module, the layer structure of user branch MLP is '$k \rightarrow 2k \rightarrow k$', and the layer structure of item branch MLP is '$2k \rightarrow 2k \rightarrow k$'. k is the embedding dimension. We vary k in the range of $\{64, 128\}$, but in order to balance the trade-off of recommendation

Table 3: Statistics of three dual-target CDR tasks.

Tasks	Datasets	#Users	#Items	#Interactions	Density
Task 1	Douban-Movie	2106	9555	907219	4.508%
	Douban-Book	2106	6777	95974	0.672%
Task 2	Douban-Movie	1666	9555	781288	4.908%
	Douban-Music	1666	5567	69681	0.751%
Task 3	Amazon-Elec	15761	51447	224689	0.027%
	Amazon-Cloth	15761	48781	133609	0.017%

accuracy and model training time, we finally set k to 64. The parameters of all these layers are initialized as the Gaussian distribution $X \sim \mathcal{N}(0, 0.01)$. For each observed user-item interaction, following GA-DTCDR [59], we randomly sample 7 unseen interactions as negative instances. For a fair comparison, we leverage the grid search to tune the choice of parameters of all models. For the baseline models, we tune them based on the best parameter settings listed in their original papers. Specifically, we choose the learning rate from $\{0.01, 0.005, 0.001, 0.0005, 0.0001\}$, and search the regularization coefficient in the range of $\{0.001, 0.0001, 0.00001\}$. In addition, we apply the Adam [22] to optimize all the models, and the batch size is 1024. We train our model and other baseline models with 100 epochs in order to guarantee the convergence. Moreover, we investigate the number of GCN layers l in $\{1, 2, 3, 4\}$, α of $Beta(\alpha, \alpha)$ in $\{0.1, 0.5, 1, 2, 5\}$ and the weights of domain classification losses μ_1, μ_2 in $\{0.1, 0.3, 0.5, 0.7, 1, 3, 5, 10\}$, and analyze their impact on the recommendation performance of our model in Section 4.6. In the experiments, we set $l = 2$, $\alpha = 1$, and $\mu_1 = \mu_2 = 1$ by default and we resample the mixing coefficient λ once for training each batch of data.

4.1.3 Evaluation Metrics. Since the leave-one-out method is ubiquitous in baseline models, such as GA-DTCDR [59] and BiT-GCF [29], we also employ it to evaluate the recommendation performance of our proposed DIDA-CDR and baseline models. In other words, we utilize the last interaction record of each test user to form the test set, while all the other interaction records are used as the training set. Following the methods introduced in [3, 24], for each test user-item interaction, we randomly sample 999 items that the test user has not interacted with as negative items, and then predict 1000 candidate scores for ranking. The leave-one-out method contains two main metrics, i.e., Hit Ratio (HR) and Normalized Discounted Cumulative Gain (NDCG) [44], which are widely-used ranking evaluation metrics [30, 61]. In the experiments, we employ them to evaluate the performance of the top-10 ranking results. For a fair evaluation, we perform all experiments 5 times and present the average results.

4.1.4 Comparison Methods. We select a total of nine representative and state-of-the-art baseline models to compare with our proposed DIDA-CDR. These nine baseline models can be divided into four categories, i.e., (1) Single-Domain Recommendation (SDR), (2) Single-Target Cross-Domain Recommendation (CDR), (3) Conventional Dual-Target CDR, and (4) Disentanglement-Based Dual-Target or Multi-Target CDR. For a clear comparison, in Table 4, we elaborate on the embedding strategies and transfer strategies of the above nine baseline models and our proposed DIDA-CDR.

Table 4: The comparison of the baselines and our approach [59].

	Model		Embedding Strategy	Transfer Strategy
Baselines	Single-Domain Recommendation (SDR)	**NGCF** [45]	Non-linear MLP	-
		LightGCN [16]	Non-linear MLP	-
	Single-Target Cross-Domain Recommendation (CDR)	**BPR_EMCDR** [35]	Linear Matrix Factorization (MF)	MLP
		BPR_DCDCSR [58]	Linear MF	Combination & MLP
	Conventional Dual-Target CDR	**PPGN** [55]	Graph Embedding	Combination & MLP
		BiTGCF [29]	Graph Embedding	Transfer Learning
	Disentanglement-Based Dual-Target or Multi-Target CDR	**GA-DTCDR** [59]	Graph Embedding	Combination (Element-wise Attention)
		DR-MTCDR [14]	Graph Embedding & Self-supervised Learning	Domain Adaptation
		DisenCDR [3]	Graph Embedding & VAE	Transfer Learning
Our Approach	Disentanglement-Based Dual-Target CDR	**DIDA-CDR**	Graph Embedding & Disentanglement	Transfer Learning & Combination (Attention)

Table 5: Performance comparison (%) of different approaches for three dual-target CDR tasks according to HR@10 and NDCG@10 [59]. While the results of best-performing baselines are underlined, the best results are marked in bold (* indicates $p < 0.05$, paired t-test of our proposed DIDA-CDR vs. the best-performing baselines) [62].

Datasets	SDR Baselines				Single-Target CDR Baselines				Conventional Dual-Target CDR Baselines			
	NGCF		LightGCN		BPR_EMCDR _MLP		BPR_DCDCSR		PPGN		BiTGCF	
	HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG
Douban-Movie	10.26	5.37	10.53	5.49	-	-	-	-	12.03	6.42	12.11	6.46
Douban-Book	7.31	4.08	7.35	4.15	6.25	3.93	6.74	4.02	10.52	4.78	10.58	4.93
Douban-Movie	9.16	4.23	9.24	4.25	-	-	-	-	10.09	4.35	10.14	4.41
Douban-Music	6.11	3.87	6.36	3.99	5.08	3.45	5.97	3.79	7.24	4.03	7.32	4.10
Amazon-Elec	20.22	11.97	20.03	10.94	-	-	-	-	22.06	12.44	21.79	12.31
Amazon-Cloth	10.95	6.01	11.38	6.10	9.87	5.33	10.90	5.86	13.04	6.91	13.16	6.88

Datasets	Disentanglement-Based Dual-Target or Multi-Target CDR Baselines						Disentanglement-Based Dual-Target CDR (our)								Improvement	
	GA-DTCDR		DR-MTCDR		DisenCDR		DIDA-CDR _Fixed		DIDA-CDR _Base		DIDA-CDR _ELBO		DIDA-CDR		(DIDA-CDR vs. best baselines)	
	HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG
Douban-Movie	12.25	6.51	14.74	7.89	15.09	8.02	16.12	8.97	12.98	6.51	15.74	8.51	**16.66***	**9.16***	10.40%	14.21%
Douban-Book	10.71	5.06	12.66	7.45	12.40	7.27	13.08	7.82	10.41	5.52	12.80	7.69	**13.79***	**8.18***	8.93%	9.80%
Douban-Movie	10.35	4.57	11.27	5.74	12.13	5.95	12.76	6.23	10.45	4.99	12.55	6.18	**13.01***	**6.60***	7.25%	10.92%
Douban-Music	7.42	4.19	8.49	4.73	8.92	5.02	9.54	5.49	8.36	4.24	9.36	5.37	**9.97***	**5.79***	11.77%	15.34%
Amazon-Elec	23.87	13.20	22.34	12.98	23.77	13.61	24.68	13.83	22.03	12.97	24.14	13.72	**25.05***	**14.36***	4.94%	5.51%
Amazon-Cloth	13.94	7.09	14.13	7.59	15.46	8.42	16.01	8.94	12.97	6.93	15.76	8.63	**16.69***	9.33	7.96%	10.81%

4.2 Performance Comparison (for RQ1)

Table 5 presents the performance comparison[2] of various approaches for three dual-target CDR tasks according to HR@10 and NDCG@10. Note that since single-target CDR baselines aim to improve the recommendation performance on the sparser domain, we train them on both domains and then only present their experimental results on the sparser domain. From Table 5, we have the following observations: (1) Our proposed DIDA-CDR improves other baseline models on sparser domain by a large margin. Specifically, on sparser domain, it outperforms the best-performing baseline model with an average improvement of 9.55% in terms of HR@10 and 11.98% in terms of NDCG@10. This is because we adopt the interpolative data augmentation, which effectively generate both

relevant and diverse augmented user representations to augment the sparser domain, and therefore significant improvements in recommendation performance can be obtained on the sparser domain; (2) Disentanglement-based dual-target CDR models improve conventional dual-target CDR models by an average of 14.98% in terms of HR@10 and 20.94% in terms of NDCG@10, which shows that decoupling and then transferring the domain-shared information to both domains is an efficient way for dual-target CDR; (3) Compared with other disentanglement-based dual-target CDR baselines, our model can achieve better recommendation performance over them. Specifically, our proposed DIDA-CDR achieves an average increase of 8.54% in terms of HR@10 and 11.10% in terms of NDCG@10, compared to the best-performing disentanglement-based dual-target CDR baseline. This is because our model particularly takes domain-independent user preferences into consideration, and the proposed disentanglement module can more effectively disentangle all three

---

[2]Due to space limitation, we only show the results when $k = 64$ in Table 5. For the results under other values of k that are omitted, similarly, our model also has a significant improvement over other baseline models.

essential components of user preferences, thus capturing more comprehensive user preferences. For a more detailed analysis of each module and each component of user preferences of the proposed DIDA-CDR, please refer to Section 4.3 and Section 4.4, respectively.

4.3 Ablation Study (for RQ2)

To show the contribution of each proposed component to the improvement of overall performance, we modify our proposed model to form three variants and conduct an ablation study for three dual-target CDR tasks.

*4.3.1 **Impact of Interpolative Data Augmentation**.* We construct a variant of DIDA-CDR, namely **DIDA-CDR_Fixed**, by replacing the interpolative data augmentation module with a fixed mixing strategy. In fact, we conducted experiments to select the best-performing mixing coefficient, i.e., 0.5, from {0.1, 0.3, 0.5, 0.7, 0.9} to implement the above variant. From Table 5, we can observe that with the interpolative data augmentation, our proposed DIDA-CDR outperforms **DIDA-CDR_Fixed** with an average improvement of 3.7%. This demonstrates that the interpolative data augmentation can not only augment the sparser domain by introducing randomness to increase the diversity of augmentation, but also generate representative augmented user representations by effectively mixing user embeddings for subsequent disentanglement. Meanwhile, the introduction of randomness also weakens the possible negative transfer caused by the linear interpolation operation with fixed weights, which can also be seen from the experimental results.

*4.3.2 **Impact of User Preference Disentanglement**.* Furthermore, another variant, namely **DIDA-CDR_Base**, directly feeds the generated user embeddings to the information fusion module and does not include the disentanglement module, thus the variant only includes the prediction loss $\mathcal{L}_{prd}$, that is $\mu_1 = \mu_2 = 0$. From Table 5, we can observe that without the disentanglement module, the recommendation performance of **DIDA-CDR_Base** would degrade to be comparable to that of the conventional dual-target CDR baselines, and weaker than that of disentanglement-based ones. This shows that the disentanglement module can indeed help the model perform more effective cross-domain knowledge transfer without negative transfer by decoupling domain-shared, domain-specific, domain-independent information and transfering only domain-shared information, thus improving the performance of cross-domain recommendations.

*4.3.3 **Impact of Domain Classifier**.* In addition, following DisenCDR [3], we modify the disentanglement module in our model, i.e., replace our domain classification losses with the standard ELBO, to form another variant, namely **DIDA-CDR_ELBO**. From Table 5, we can observe that our proposed DIDA-CDR outperforms **DIDA-CDR_ELBO** with an average improvement of 5.98%. This demonstrates that the proposed disentanglement module can indeed collaborate well with the domain classifier to decouple more accurate essential components of user preferences, especially the domain-independent information, to capture comprehensive user preferences, thus enabling the model to achieve better recommendation performance through superior disentanglement.

Figure 3: (a)-(b): Performance comparison between our model and its four variants. (c)-(d): Performance comparison of adopted information fusion approaches.

4.4 Impact of Various Components of User Preferences (for RQ3)

To demonstrate that all three components of user preferences, i.e., domain-share, domain-specific and domain-independent information, are essential and effective for recommendation, and do not require transfer of domain-independent user preferences, we compare DIDA-CDR with its four variants, including DIDA-CDR (w/o sha.), DIDA-CDR (w/o spe.), DIDA-CDR (w/o ind.) and DIDA-CDR (transfer ind.). Fig. 3(a)-(b) shows the performance comparison between our model and the above four variants[3]. The differences between various variants and the impact of each component of user preferences are elaborated in the following subsections.

*4.4.1 **Impact of Domain-Shared User Preferences**.* **DIDA-CDR (w/o sha.)** extracts the domain-shared user preferences from both domains, but does not transfer them to any domain. From Fig. 3(a)-(b), we can see that our proposed DIDA-CDR outperforms DIDA-CDR (w/o sha.) with an average improvement of 17.21%. This is because the domain-shared user preferences are valuable information, which plays an important role in cross-domain recommendation and can improve the recommendation performance on both domains simultaneously.

*4.4.2 **Impact of Domain-Specific User Preferences**.* **DIDA-CDR (w/o spe.)** does not consider the domain-specific user preferences when making recommendations. From Fig. 3(a)-(b), we can observe that our proposed DIDA-CDR outperforms DIDA-CDR (w/o spe.) with an average improvement of 36.12%. This is because

---

[3]Due to space limitation, we only show the results on Task 1 in Fig. 3, i.e., only the results on the pair of datasets consisting of Douban-Movie and Douban-Book are presented, and similar trends can be observed for results on the other omitted tasks. Similarly, Fig. 4 and Fig. 5 only shows the results on Task 1 for the same reason above.

Figure 4: (a)-(b): Impact of the number of GCN layers. (c)-(d): Impact of α.

Figure 5: (a)-(b): Impact of μ_1. (c)-(d): Impact of μ_2.

the domain-specific user preferences are inherent personalized preferences of users in each domain. If it is not considered when making recommendations, the recommendation performance of our model will be significantly reduced.

*4.4.3 **Impact of Domain-Independent User Preferences**.* **DIDA-CDR (w/o ind.)** includes the domain-shared and domain-specific user preferences, but does not include the domain-independent user preferences when capturing comprehensive user preferences; **DIDA-CDR (transfer ind.)** disentangles the domain-independent user preferences and then transfer them to another domain. It can be seen from Fig. 3(a)-(b) that our proposed DIDA-CDR outperforms DIDA-CDR (w/o ind.) and DIDA-CDR (transfer ind.) with an average improvement of 12.40% and 26.04%, respectively. In view of this, we make the following qualitative analysis. The domain-independent user preferences seemingly exist in each domain, but actually have different meanings. If they are not included when capturing user preferences, the captured user preferences are incomplete, resulting in the degraded recommendation performance of the model. Moreover, if the domain-independent information is transferred to other domains, it will provide the useless information and cause the performance degradation.

4.5 Impact of Various Information Fusion Approaches (for RQ4)

After we obtain three essential user preference components, we compare various information fusion approaches, i.e., concatenation, element-wise sum, and attention, to fuse them into comprehensive user preferences. The performance comparison of three used information fusion approaches is presented in Fig. 3(c)-(d). We find that when our model utilizes the attention mechanism for information fusion, it improves the variants using concatenation and summation by an average of 28.37% and 20.69%, respectively. This is because the attention mechanism can not only capture the relationship between

various components, but also selectively highlight the key information and weaken the redundant information by learning weights. In this paper, the attention mechanism measures the importance of domain-shared, domain-specific and domain-independent information to comprehensive user preferences through weights, and weakens the redundant information between the domain-shared and domain-independent user preferences, thus enabling our model to achieve better recommendation results.

4.6 Parameter Sensitivity (for RQ5)

*4.6.1 **Impact of l**.* Stacking too many layers when training a deep GCN is prone to over-smoothing [27]. In order to explore this effect, we investigate the number of GCN layers l in the range of $\{1, 2, 3, 4\}$ in the experiments and report the experimental results in Fig. 4(a)-(b). It can be observed that 2-layer GCN is significantly better than 1-layer GCN, which indicates that stacking a moderate number of layers is beneficial for mining higher-order user-item relationships. However, the recommendation performance of our proposed DIDA-CDR drops on some datasets when $l = 3$ and degrades even more when $l = 4$. The reason may be that when the number of layers is greater than 3, the problem of over-smoothing occurs, resulting in the fact that increasing the number of layers at this time will in turn reduce the recommendation performance of our model.

*4.6.2 **Impact of α**.* $Beta(\alpha, \alpha)$ is Uniform distribution when $\alpha = 1$, Bimodal distribution when $\alpha < 1$ and Bell-shaped distribution when $\alpha > 1$ [5]. In order to explore from which distribution sampling λ can help the recommendation performance of model the most, we search α in the range of $\{0.1, 0.5, 1, 2, 5\}$. The performance comparison is illustrated in Fig. 4(c)-(d). We can see that the best performance of our proposed DIDA-CDR is achieved when $\alpha = 1$, which presents that λ sampled from the Uniform distribution can effectively mix the user embeddings in the richer domain and the sparser domain, thereby effectively alleviating the data imbalance

and improving the recommendation performance of model. In contrast, when $\alpha < 1$ or $\alpha > 1$, sampling λ from $Beta(\alpha, \alpha)$ will result in the performance degradation.

4.6.3 **Impact of μ_1 and μ_2.** To explore the effect of weights of domain classification losses on the overall recommendation performance of our DIDA-CDR, we vary μ_1 and μ_2 in $\{0.1, 0.3, 0.5, 0.7, 1, 3, 5, 10\}$. The results are reported in Fig. 5. It can be seen from Fig. 5(a)-(b) that the contribution of domain classification loss $\mathcal{L}_{cls_1}$ to decoupling more accurate domain-specific user preferences is small when $\mu_1 \rightarrow 0$. When μ_1 is too large, the domain classification loss $\mathcal{L}_{cls_1}$ receives more attention during the learning process. As a result, the contribution of prediction loss $\mathcal{L}_{prd}$ to the model is weakened, which reduces the recommendation performance of the model. Similarly, from Fig. 5(c)-(d), a similar trend can be observed for the weights μ_2 of domain classification loss L_{cls_2}. Empirically, we choose $\mu_1 = \mu_2 = 1$.

5 CONCLUSION AND FUTURE WORK

In this paper, we have proposed a novel Disentanglement-based framework with Interpolative Data Augmentation for dual-target Cross-Domain Recommendation, called DIDA-CDR. DIDA-CDR consists of an interpolative data augmentation approach to generating both relevant and diverse augmented user representations to augment the sparser domain and explore the potential user preferences, and a user preference disentanglement module to decouple essential components of user preferences to capture comprehensive user preferences, all of which help improve the recommendation performance on both domains simultaneously. Also, we have conducted extensive experiments on five real-world datasets to show the significant superiority of DIDA-CDR over the state-of-the-art methods. In the future, we plan to extend our model to multi-target CDR scenarios where users partially overlap.

ACKNOWLEDGMENTS

REFERENCES

[1] Yoshua Bengio, Aaron Courville, and Pascal Vincent. 2013. Representation Learning: A Review and New Perspectives. *TPAMI* (2013), 1798–1828.
[2] Shuqing Bian, Wayne Xin Zhao, Jinpeng Wang, and Ji-Rong Wen. 2022. A Relevant and Diverse Retrieval-enhanced Data Augmentation Framework for Sequential Recommendation. In *CIKM*. 2923–2932.
[3] Jiangxia Cao, Xixun Lin, Xin Cong, Jing Ya, Tingwen Liu, and Bin Wang. 2022. DisenCDR: Learning Disentangled Representations for Cross-Domain Recommendation. In *SIGIR*. 267–277.
[4] Jiangxia Cao, Jiawei Sheng, Xin Cong, Tingwen Liu, and Bin Wang. 2022. Cross-Domain Recommendation to Cold-Start Users via Variational Information Bottleneck. In *ICDE*. 2209–2223.
[5] Huiyuan Chen, Chin-Chia Michael Yeh, Fei Wang, and Hao Yang. 2022. Graph Neural Transport Networks with Non-local Attentions for Recommender Systems. In *WWW*. 1955–1964.
[6] Ricky T. Q. Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. 2018. Isolating Sources of Disentanglement in Variational Autoencoders. In *NeurIPS*.
[7] Yoonhyuk Choi, Jiho Choi, Taewook Ko, Hyungho Byun, and Chong-Kwon Kim. 2022. Review-Based Domain Disentanglement without Duplicate Users or Contexts for Cross-Domain Recommendation. In *CIKM*. 293–303.
[8] Qiang Cui, Tao Wei, Yafeng Zhang, and Qing Zhang. 2020. HeroGRAPH: A Heterogeneous Graph Framework for Multi-Target Cross-Domain Recommendation. In *RecSys*.
[9] Wenjing Fu, Zhaohui Peng, Senzhang Wang, Yang Xu, and Jin Li. 2019. Deeply Fusing Reviews and Contents for Cold Start Users in Cross-Domain Recommendation Systems. In *AAAI*. 94–101.
[10] Yuqian Fu, Yanwei Fu, Jingjing Chen, and Yu-Gang Jiang. 2022. Generalized Meta-FDMixup: Cross-Domain Few-Shot Learning Guided by Labeled Target Data. *TIP* (2022), 7078–7090.
[11] Yuqian Fu, Yanwei Fu, and Yu-Gang Jiang. 2021. Meta-FDMixup: Cross-Domain Few-Shot Learning Guided by Labeled Target Data. In *MM*. 5326–5334.
[12] Anna Gibbs. 2015. Writing as method: attunement, resonance, and rhythm. *Affective methodologies: Developing cultural research strategies for the study of affect* (2015), 222–236.
[13] Abel Gonzalez-Garcia, Joost van de Weijer, and Yoshua Bengio. 2018. Image-to-image Translation for Cross-domain Disentanglement. In *NeurIPS*.
[14] Xiaobo Guo, Shaoshuai Li, Naicheng Guo, Jiangxia Cao, Xiaolei Liu, Qiongxu Ma, Runsheng Gan, and Yunan Zhao. 2023. Disentangled Representations Learning for Multi-target Cross-domain Recommendation. *TOIS* (2023), 1–27.
[15] Marc D Hauser and Josh McDermott. 2003. The evolution of the music faculty: A comparative perspective. *Nature neuroscience* (2003), 663–668.
[16] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. Lightgcn: Simplifying and Powering Graph Convolution Network for Recommendation. In *SIGIR*. 639–648.
[17] Yupeng Hou, Shanlei Mu, Wayne Xin Zhao, Yaliang Li, Bolin Ding, and Ji-Rong Wen. 2022. Towards Universal Sequence Representation Learning for Recommender Systems. In *KDD*. 585–593.
[18] Guangneng Hu, Yu Zhang, and Qiang Yang. 2018. Conet: Collaborative Cross Networks for Cross-Domain Recommendation. In *CIKM*. 667–676.
[19] Guangneng Hu, Yu Zhang, and Qiang Yang. 2019. Transfer Meets Hybrid: A Synthetic Approach for Cross-Domain Collaborative Filtering with Text. In *WWW*. 2822–2829.
[20] Tinglin Huang, Yuxiao Dong, Ming Ding, Zhen Yang, Wenzheng Feng, Xinyu Wang, and Jie Tang. 2021. MixGCF: An Improved Training Method for Graph Neural Network-based Recommender Systems. In *KDD*. 665–674.
[21] Heishiro Kanagawa, Hayato Kobayashi, Nobuyuki Shimizu, Yukihiro Tagami, and Taiji Suzuki. 2019. Cross-domain Recommendation via Deep Domain Adaptation. In *ECIR*. 20–29.
[22] Diederik P Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*.
[23] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *ICLR*.
[24] Walid Krichene and Steffen Rendle. 2022. On Sampled Metrics for Item Recommendation. *CACM* (2022), 75–83.
[25] Chenglin Li, Yuanzhen Xie, Chenyun Yu, Bo Hu, Zang Li, Guoqiang Shu, Xiaohu Qie, and Di Niu. 2023. One for All, All for One: Learning and Transferring User Embeddings for Cross-Domain Recommendation. In *WSDM*. 366–374.
[26] Pan Li and Alexander Tuzhilin. 2020. Ddtcdr: Deep Dual Transfer Cross Domain Recommendation. In *WSDM*. 331–339.
[27] Qimai Li, Zhichao Han, and Xiao-Ming Wu. 2018. Deeper Insights into Graph Convolutional Networks for Semi-supervised Learning. In *AAAI*. 3538–3545.
[28] Jian Liu, Pengpeng Zhao, Fuzhen Zhuang, Yanchi Liu, Victor S Sheng, Jiajie Xu, Xiaofang Zhou, and Hui Xiong. 2020. Exploiting Aesthetic Preference in Deep Cross Networks for Cross-Domain Recommendation. In *WWW*. 2768–2774.
[29] Meng Liu, Jianjun Li, Guohui Li, and Peng Pan. 2020. Cross Domain Recommendation via Bi-directional Transfer Graph Collaborative Filtering Networks. In *CIKM*. 885–894.
[30] Weiming Liu, Xiaolin Zheng, Jiajie Su, Mengling Hu, Yanchao Tan, and Chaochao Chen. 2022. Exploiting Variational Domain-Invariant User Embedding for Partially Overlapped Cross Domain Recommendation. In *SIGIR*. 312–321.
[31] Zhiwei Liu, Lei Zheng, Jiawei Zhang, Jiayu Han, and S Yu Philip. 2019. JSCN: Joint Spectral Convolutional Network for Cross Domain Recommendation. In *Big Data*. 850–859.
[32] Babak Loni, Yue Shi, Martha Larson, and Alan Hanjalic. 2014. Cross-domain Collaborative Filtering with Factorization Machines. In *ECIR*. 656–661.
[33] Jianxin Ma, Chang Zhou, Peng Cui, Hongxia Yang, and Wenwu Zhu. 2019. Learning Disentangled Representations for Recommendation. In *NeurIPS*. 5711–5722.
[34] Jianxin Ma, Chang Zhou, Hongxia Yang, Peng Cui, Xin Wang, and Wenwu Zhu. 2020. Disentangled Self-Supervision in Sequential Recommenders. In *KDD*. 483–491.
[35] Tong Man, Huawei Shen, Xiaolong Jin, and Xueqi Cheng. 2017. Cross-Domain Recommendation: An Embedding and Mapping Approach. In *IJCAI*. 2464–2470.
[36] Zaiqiao Meng, Siwei Liu, Craig Macdonald, and Iadh Ounis. 2021. Graph Neural Pre-training for Enhancing Recommendations using Side Information. *arXiv preprint arXiv:2107.03936* (2021).
[37] Ashish Kumar Sahu and Pragya Dwivedi. 2020. Knowledge Transfer by Domain-independent User Latent Factor for Cross-domain Recommender Systems. *FGCS* (2020), 320–333.
[38] Kimiaki Shirahama, Kazuhisa Iwamoto, and Kuniaki Uehara. 2004. Video data mining: rhythms in a movie. In *ICME*. 1463–1466.
[39] Hongzu Su, Yifei Zhang, Xuejiao Yang, Hua Hua, Shuangyang Wang, and Jingjing Li. 2022. Cross-domain Recommendation via Adversarial Adaptation. In *CIKM*. 1808–1817.

[40] Jianing Sun, Yingxue Zhang, Chen Ma, Mark Coates, Huifeng Guo, Ruiming Tang, and Xiuqiang He. 2019. Multi-Graph Convolution Collaborative Filtering. In *ICDM*. 1306–1311.

[41] Ke Wang, Yanmin Zhu, Haobing Liu, Tianzi Zang, Chunyang Wang, and Kuan Liu. 2022. Inter-and Intra-Domain Relation-Aware Heterogeneous Graph Convolutional Networks for Cross-Domain Recommendation. In *DASFAA*. 53–68.

[42] Xin Wang, Hong Chen, Si'ao Tang, Zihao Wu, and Wenwu Zhu. 2022. Disentangled Representation Learning. *arXiv preprint arXiv:2211.11695* (2022).

[43] Xin Wang, Hong Chen, Yuwei Zhou, Jianxin Ma, and Wenwu Zhu. 2022. Disentangled Representation Learning for Recommendation. *TPAMI* (2022), 408–424.

[44] Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. 2019. KGAT: Knowledge Graph Attention Network for Recommendation. In *KDD*. 950–958.

[45] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural Graph Collaborative Filtering. In *SIGIR*. 165–174.

[46] Xiang Wang, Hongye Jin, An Zhang, Xiangnan He, Tong Xu, and Tat-Seng Chua. 2020. Disentangled Graph Collaborative Filtering. In *SIGIR*. 1001–1010.

[47] Yiwei Wang, Wei Wang, Yuxuan Liang, Yujun Cai, and Bryan Hooi. 2021. Mixup for Node and Graph Classification. In *WWW*. 3663–3674.

[48] Shuo Xiao, Dongqing Zhu, Chaogang Tang, and Zhenzhen Huang. 2023. CATCL: Joint Cross-Attention Transfer and Contrastive Learning for Cross-Domain Recommendation. In *DASFAA*. 446–461.

[49] Feng Yuan, Lina Yao, and Boualem Benatallah. 2019. DARec: Deep Domain Adaptation for Cross-Domain Recommendation via Transferring Rating Patterns. In *IJCAI*. 4227–4233.

[50] Tianzi Zang, Yanmin Zhu, Haobing Liu, Ruohan Zhang, and Jiadi Yu. 2022. A Survey on Cross-Domain Recommendation: Taxonomies, Methods, and Future Directions. *TOIS* (2022), 1–39.

[51] Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. 2018. *mixup*: Beyond Empirical Risk Minimization. In *ICLR*.

[52] Ruohan Zhang, Tianzi Zang, Yanmin Zhu, Chunyang Wang, Ke Wang, and Jiadi Yu. 2023. Disentangled Contrastive Learning for Cross-Domain Recommendation.

[53] Xinyue Zhang, Jingjing Li, Hongzu Su, Lei Zhu, and Heng Tao Shen. 2022. Multi-Level Attention-Based Domain Disentanglement for Bidirectional Cross-Domain Recommendation. *TOIS* (2022).

[54] Yin Zhang, Ziwei Zhu, Yun He, and James Caverlee. 2020. Content-Collaborative Disentanglement Representation Learning for Enhanced Recommendation. In *RecSys*. 43–52.

[55] Cheng Zhao, Chenliang Li, and Cong Fu. 2019. Cross-Domain Recommendation via Preference Propagation Graphnet. In *CIKM*. 2165–2168.

[56] Xiaoyun Zhao, Ning Yang, and Philip S Yu. 2022. Multi-Sparse-Domain Collaborative Recommendation via Enhanced Comprehensive Aspect Preference Learning. In *WSDM*. 1452–1460.

[57] Feng Zhu, Chaochao Chen, Yan Wang, Guanfeng Liu, and Xiaolin Zheng. 2019. DTCDR: A Framework for Dual-Target Cross-Domain Recommendation. In *CIKM*. 1533–1542.

[58] Feng Zhu, Yan Wang, Chaochao Chen, Guanfeng Liu, Mehmet Orgun, and Jia Wu. 2018. A Deep Framework for Cross-Domain and Cross-System Recommendations. In *IJCAI*. 3711–3717.

[59] Feng Zhu, Yan Wang, Chaochao Chen, Guanfeng Liu, and Xiaolin Zheng. 2020. A Graphical and Attentional Framework for Dual-Target Cross-Domain Recommendation. In *IJCAI*. 3001–3008.

[60] Feng Zhu, Yan Wang, Chaochao Chen, Jun Zhou, Longfei Li, and Guanfeng Liu. 2021. Cross-Domain Recommendation: Challenges, Progress, and Prospects. In *IJCAI*. 4721–4728.

[61] Feng Zhu, Yan Wang, Jun Zhou, Chaochao Chen, Longfei Li, and Guanfeng Liu. 2021. A Unified Framework for Cross-Domain and Cross-System Recommendations. *TKDE* (2021), 1171–1184.

[62] Yongchun Zhu, Ruobing Xie, Fuzhen Zhuang, Kaikai Ge, Ying Sun, Xu Zhang, Leyu Lin, and Juan Cao. 2021. Learning to Warm Up Cold Item Embeddings for Cold-start Recommendation with Meta Scaling and Shifting Networks. In *SIGIR*. 1167–1176.