Diverse Preference Augmentation with Multiple Domains for Cold-start Recommendations

Yan Zhang$^{†‡♭}$, Changyu Li†, Ivor W. Tsang$^{♯‡}$, Hui Xu†*, Lixin Duan†*, Hongzhi Yin§, Wen Li†, Jie Shao†

†University of Electronic Science and Technology of China, Chengdu 611731, China
‡Australian Artificial Intelligence Institute, University of Technology Sydney, Sydeny 2007, Australia
$^♭$Intelligent Terminal Key Laboratory of Sichuan Province, Yibin 644000, China
$^♯$Center for Frontier AI Research, Research Agency for Science, Technology and Research (A$^⋆$STAR), Singapore
§The University of Queensland, Brisbane 4067, Australia

*yixianqianzy@gmail.com, changyulve@std.uestc.edu.cn, Ivor.Tsang@gmail.com, hui_xu@std.uestc.edu.cn,
lxduan@uestc.edu.cn, db.hongzhi@gmail.com, {liwen, shaojie}@uestc.edu.cn*

Abstract—Cold-start issues have been more and more challenging for providing accurate recommendations with the fast increase of users and items. Most existing approaches attempt to solve the intractable problems via content-aware recommendations based on auxiliary information and/or cross-domain recommendations with transfer learning. Their performances are often constrained by the extremely sparse user-item interactions, unavailable side information, or very limited domain-shared users. Recently, meta-learners with meta-augmentation by adding noises to labels have been proven to be effective to avoid overfitting and shown good performance on new tasks. Motivated by the idea of meta-augmentation, in this paper, by treating a user's preference over items as a task, we propose a so-called Diverse Preference Augmentation framework with multiple source domains based on meta-learning (referred to as MetaDPA) to i) generate diverse ratings in a new domain of interest (known as target domain) to handle overfitting on the case of sparse interactions, and to ii) learn a preference model in the target domain via a meta-learning scheme to alleviate cold-start issues. Specifically, we first conduct multi-source domain adaptation by dual conditional variational autoencoders and impose a Multi-domain InfoMax (MDI) constraint on the latent representations to learn domain-shared and domain-specific preference properties. To avoid overfitting, we add a Mutually-Exclusive (ME) constraint on the output of decoders to generate diverse ratings given content data. Finally, these generated diverse ratings and the original ratings are introduced into the meta-training procedure to learn a preference meta-learner, which produces good generalization ability on cold-start recommendation tasks. Experiments on real-world datasets show our proposed MetaDPA clearly outperforms the current state-of-the-art baselines.

Index Terms—Recommender system, preference augmentation, meta leaning, cold-start

I. INTRODUCTION

Recommender systems have shown great success in both academia and industries, and so become indispensable in our life by helping us filter millions of possible choices. Recommender systems provide a small set of items from the underlying pool of items based on users' historical interactions and their side information. One of the well known recommendation frameworks is Collaborative Filtering (CF) [1], [2], where the only available data is user-item historical interactive information. A key challenge in CF-based methods is to provide accurate recommendations from a large number of items with extremely sparse interactions [3]. Such recommender systems suffer from poor performance due to sparse interactive data or ratings and cannot even handle user cold-start and item cold-start issues brought by new users and new items [4]. So as a very critical problem in recommender systems, how to make accurate recommendation under sparse and cold-start scenarios attracts rising attention from a wide range of stakeholders in recent years.

Existing approaches for solving cold-start and sparse issues are proposed from the following three directions: (1) content-aware recommender systems [5]–[9] that integrate auxiliary side information and interactive data to enhance representations of users and items, and then feed them into the preference model to improve the performance of recommender systems, where user's and item's content information (user's profile, item's description, user's review, etc.) is taken as auxiliary side information for strengthening representations of users and items. (2) Cross-domain recommender systems transfer preference knowledge from source domains to its similar target domain, and then improve the performance of recommendations in the target domain. These methods can be categorized into single source [10]–[17] and multiple source [18]–[24] cross-domain recommendations based on the number of source domains applied for transferring preference knowledge. (3) Meta-learning based recommender systems [25], [25]–[27] learn the prior preference distribution of users over items by taking users' preferences as meta-learning tasks. They can improve the performance of recommendations under sparse and cold-start settings by fine-tuning the preference model with only a few ratings.

Content-aware recommender systems have been widely studied for solving the cold-start and sparse issues with the help of auxiliary side information. One of the most commonly used side information is content [2], [5]–[9], [28]–[30]. By learning with content data, the features of users or items with just a few ratings or even no ratings can be then effectively represented, making those content-aware recommender systems be able to improve the performance under sparse and cold-start

---

*Corresponding authors: Lixin Duan, Hui Xu.

scenarios. However, there exist inconsistencies between item content and user preferences. That is to say, users under the same profile (e.g., age, occupation, gender, place of residence, etc.) often have different preferences over the same item; and items having similar content (e.g., description, category, users' reviews, etc.) are often rated with different scores by the same user. As a result, the performance improvement of content-aware recommender systems are limited by the gap between content and preferences.

Cross-domain recommendations based on transfer learning are another type of solutions for solving cold-start and sparse challenges. Existing methods transfer knowledge from source domains with rich preference information to a target domain with very sparse historical data [12]–[14], [31]. Moreover, those methods transfer preference information by domain adaptation with domain-shared users in order to strengthen the representations of users and items. Thus, the cross-domain model can learn effective representations of users and items than only using preference information of a single domain. However, the limit number of shared users affects the capability of preference transferring, which restricts the performance improvement in the target domain. For example, on Amazon datasets, Books and Electronics subsets only share 5% users, which limits the transferable preference patterns from the source domain to the target domain, and thus the performance of cross-domain recommendations is also constrained.

To acquire more preference patterns, some scholars study multi-domain recommendations of transferring preference patterns from multiple source domains to a target domain [18]–[24]. These methods extract correlations between source domains and the target domain and tie factors from different source domains together. Such correlation enrich rating patterns of the target domain with multiple related source domains. Thus, these methods can achieve better performance than single source domain. The transferable preference patterns are also limited because the correlation is dependent on shared users. The performance is still limited by the ratio of shared users among all users. In addition, some of these multi-source cross-domain methods can only provide recommendation for shared users and do not work for providing recommendations for unshared users. Besides, the augmentation method is another way to acquire more preference patterns, such as AugCF [32]. It is designed based on Conditional Generative Adversarial Nets by considering the class (like or dislike) as a feature to generate new interaction data, which is evaluated to be a sufficiently real augmentation to the original dataset in their work.

Meta-learning has been validated as a promising approach for mitigating cold-start issues of recommendations [4], [25], [27], [33], [34], which treats user's preferences as meta-learning tasks and learns a preference prior distribution of all users over items. Meta-learning based methods can fast adapt to new users' or new items' recommendations with the learned meta-learner of the preference prediction model. However, the meta-learner easily overfits to the sparse preference (rating) data, which makes it difficult to provide accurate recommendations under new users or new items settings.

To avoid overfitting on meta-training tasks, meta-augmentation [35] adds noise to labels y without changing inputs x. It is capable of handling two forms of overfitting: (1) memorization overfitting, in which the model is able to overfit to the training set without relying on the meta-learner, and (2) meta-learner overfitting, in which the learner overfits to the training set and does not generalize to the test set. Yin et al. [36] identify the memorization overfitting can happen when the set of tasks are non-mutually-exclusive. The meta-augmentation is proven to avoid memorization overfitting effectively by transforming the task setting from non-mutually-exclusive to mutually-exclusive and proven to avoid the learner overfitting effectively [36]. Tasks are said to be mutually-exclusive [35] if a single model cannot solve them all at once. For example, if the task $\mathcal{T}_1$ is 'output 0 if the input image is a dog', and task $\mathcal{T}_2$ is 'output 1 if the image is a dog', then we call tasks $\{\mathcal{T}_1, \mathcal{T}_2\}$ are mutually-exclusive.

Generally, we define *'Mutual Exclusivity'* as: Training samples $(x, y_1), (x, y_2), \cdots, (x, y_k)$ are called mutually-exclusive samples, if all continuous labels $y_1, y_2, \cdots, y_k$ are different from each other with the same input x. For example, training samples $(x, 0.1), (x, 0.2), (x, 0.3)$ with the same input x are mutually-exclusive samples because all labels are different from each other. However, in practice, it is difficult to obtain training samples that meet such a strict assumption (i.e., all labels are required to be different with the same input). By relaxing the assumption, we define *'Diversity'* as: Training samples $(x, y_1), (x, y_2), \cdots, (x, y_k)$ are called diverse samples, if not all continuous labels $y_1, y_2, \cdots, y_k$ are different from each other with the same input x. For example, training samples $(x, 0.1), (x, 0.1), (x, 0.3)$ are diverse samples because not all labels are different (the first and the second samples are with the same label 0.1). It's worth noting that labels in the above two definitions are continuous values. Particularly, continuous labels are within the interval $[0, 1]$, because we train our model on real interactive data('0' or '1') and augmented interactive data within the interval $[0, 1]$. It is also worth mention that all labels are required to be different from each other in mutually-exclusive samples; while in diverse samples, some labels from $\{y_1, y_2, \cdots, y_k\}$ may be the same, so mutually-exclusive samples are diverse samples, but not vice versa.

To avoid overfitting, our initial goal is to construct mutually-exclusive samples (tasks) in the target domain for training our model. Specifically, we develop a multi-source domain adaptation module that transfers the preference patterns from multiple source domains (k source domains) to a target domain by k dual conditional variational autoencoders (Dual-CVAEs) shown in Fig. 1. By enforcing Mutually-Exclusive (ME) constraints on decoders of those k Dual-CVAEs, we hope to generate mutually-exclusive ratings by k encoder-decoder frameworks (highlighted with red line in Fig. 1) given the input $\mathbf{x}_t$. However, it is computationally intractable to obtain an optimal solution due to the strict assumption (i.e., all generated ratings from different source domains are required to be different from each other), because it needs $\mathcal{O}(k^2)$ operations

to assert the assumption in each iteration of training those k Dual-CVAEs. Accordingly, we relax the ME constraint by adding it to the objective with a weight (hyper-parameter β_2 in Eq. (8)). Hence, these generated ratings may not be all different from each other, but they increase the rating diversity [35]. Consequently, we generate diverse ratings $\hat{\mathbf{r}}_{t1}, \hat{\mathbf{r}}_{t2}, \cdots, \hat{\mathbf{r}}_{tk}$ by the learned k encoder-decoder frameworks with the same content $\mathbf{x}_t$ in the target domain, which is different from the work [35] where it augments diverse samples by adding noises to the ground-truth label y.

We call our proposed method as Diverse Preference Augmentation based on meta-learning (MetaDPA). To learn domain-shared and domain-specific information by these k Dual-CVAEs, we add Multi-domain InfoMax (MDI) constraints imposed on the latent representations of source and target domains [37], which maximizes the mutual information between representations of source and target domains. To augment diverse ratings, we impose ME constraints on the decoders of k Dual-CVAEs. Then we adopt the learned k encoder-decoders (red line in Fig. 1) to generate diverse ratings by the content data $\mathbf{x}_u$ of target domain. Finally we combine the augmented diverse ratings and true ratings to learn a prior preference distribution via a meta-learning framework in the target domain, which is expected to quickly adapt to new users' or new items' recommendations via only a few fine-tuning steps. The main contributions are summarized as follows:

- We propose a multi-source cross-domain recommender system, coined as Diverse Preference Augmentation based meta-learning (MetaDPA), for solving cold-start and sparse issues in recommendations. MetaDPA consists of three blocks: multi-source domain adaptation, diverse preference augmentation, and preference meta-learning.
- For multi-source domain adaptation, MetaDPA augments diverse ratings with content data via the ME constraint imposed on the multi-source domain adaptation.
- We develop the MDI constraint to learn domain-shared and domain-specific preference information, where the domain-shared information makes it possible to transfer preference patterns from the source domain to the target domain, and the domain-specific preference information contributes to generate diverse ratings in the preference augmentation step.
- In experiments, we demonstrate the effectiveness of MetaDPA to relieve the sparse issue ('Warm-start') and three types of cold-start issues ('C-U', 'C-I', 'C-UI') by comparing with existing competitive baselines. Besides, we conduct ablation studies to evaluate the effectiveness of ME and MDI constraints.

The rest of this paper is organized as follows. Section II introduces three types of closely related work, content-aware recommender systems in Section II-A, cross-domain recommendations in Section II-B and meta-learning based recommender systems in Section II-C. In Section III, we introduce the problem formulation and notations in Section III-A, and briefly state the recommender systems equipped with the meta-learning framework in Section III-B. Next, we give a detailed introduction of the proposed three-block MetaDPA in Section IV. Specifically, we firstly introduce the multi-source cross-domain adaptation in Section IV-A including MDI and ME constraints added on the domain adaption; we then introduce the diverse preference augmentation in Section IV-B after multi-source domain adaptation; finally we introduce the preference meta-learning framework in Section IV-C. Then we demonstrate the experiments of the proposed MetaDPA together with the competing baselines in Section V, which includes introductions of the experimental settings in Section V-A, the overall experimental results compared with baselines in Section V-B, the ablation studies to evaluate the effectiveness of MDI and ME constraints in Section V-E, and the impact of hyper-parameters β_1, β_2 the above two constraints in Section V-F. Finally, we conclude this paper in Section VI.
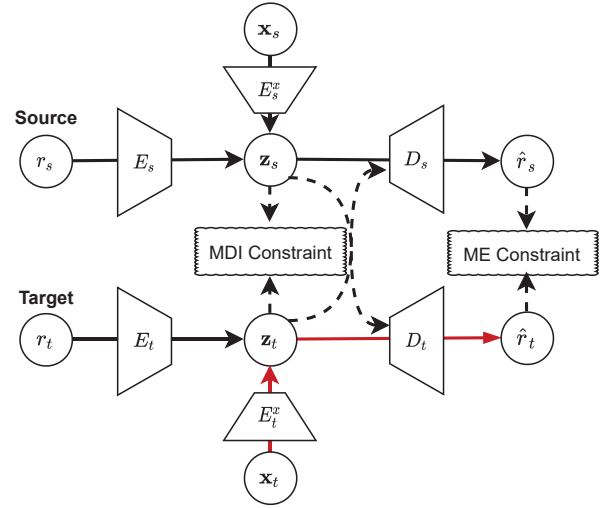

Fig. 1. The framework of cross-domain adaptation with Dual-CVAE. The MDI constraint is imposed on latent representations $\mathbf{z}_s$ and $\mathbf{z}_t$. The ME constraint is enforced on the outputs of two decoders D_s and D_t. Given ratings $\mathbf{r}_s$, $\mathbf{r}_t$ and content data $\mathbf{x}_s$, $\mathbf{x}_t$, the goal of Dual-CVAE is to learn domain-shared and domain-specific preference information, and then generate diverse ratings by decoders of k Dual-CVAEs.

II. RELATED WORK

In this section, we focus on closely related work to our method: content-aware recommender systems, cross-domain recommender systems, and mete-learning based recommender systems. The proposed MetaDPA is a kind of cross-domain recommendation that transfers preference patterns from multiple source domains to a target domain based on a meta-learning optimization framework with content information.

A. Content-aware Recommender Systems

Content-aware recommender system, as a representative content-aware method, is widely studied by scholars recently. Such as the state-of-the-art collaborative deep learning (CDL) [6] was proposed as a probabilistic model by jointly learning a probabilistic stacked denoising auto-encoder (SDAE) [38] and CF. CDL exploits the interaction and content data to alleviate cold start and data sparsity problems. However, unlike CTR, CDL takes advantage of deep learning framework

to learn effective real latent representations. Thus, it can also be applied in the cold-start and sparse settings. CDL is also a tightly coupled method for recommender systems by developing a hierarchical Bayesian model.

Another classic content-aware method is deep cooperative neural networks (CoNN) [39], which consists of two parallel neural networks: one learns user behaviors exploiting users' reviews, and the other one learns item properties from the items' reviews. A shared layer is introduced on the top to couple these two networks together. Then, dual attention mutual learning (DAML) [40] integrates ratings and reviews into a joint neural network with a local and mutual attention mechanism to strengthen the interpretability. In addition, higher-order nonlinear interaction of features are extracted by the neural factorization machines to predict ratings.

B. Cross-domain Recommender Systems

Cross-domain recommender systems, as a type of methods for solving cold-start and sparse challenges, transfer preference information from source domains to its related target domain and then improve the performance of recommendations in the target domain. These methods can be categorized into single source and multiple source cross-domain recommendations based on the number of source domains applied for transferring.

Cross-domain recommendations with single source domain, such as cross-domain triadic factorization (CDTF) [41], deep domain adaptation model (DARec) [42], and equivalent transformation learner (ETL) [43] were proposed to transfer user-item preference relations from a single source domain to a target domain without relying on any auxiliary information. By combining content information, a transfer meeting content-aware method (TMH) [44] is formulated with unstructured text in an end-to-end manner. Then, Cross-domain recommendation framework via aspect transfer network (CATN) [45] is developed via an aspect transfer network for cold-start users. Another related study is text-enhanced domain adaptation recommendation (TDAR) [46] that extracts the textual features in word semantic space for each user and item and feeds them into a domain classifier with the embeddings of users and items for better domain adaptation.

For multi-source cross-domain recommendations, one pioneer work is Collective Matrix Factorization (CMF) [18] that extends linear models to arbitrary relational domains. Then, multi-domain collaborative filtering (MCF) [47] is proposed by considering multiple collaborative filtering tasks in different domains simultaneously and exploiting the relationships between domains. MCF also introduces the link function for different domains to correct their biases. Recently, MINDTL [14] exploits transfer learning and compresses the knowledge from the source domain into a cluster-level rating matrix to model user's rating pattern. The rating patterns in the low level matrix are transferred to the target domain by enriching rating patterns by relaxing the full rating restriction on source domains.

C. Meta-learning Based Recommender Systems

Recently, many scholars focus on formulating recommendation frameworks with meta-learning [4], [26], [48] for solving cold-start and sparse issues. Current meta-learning-based recommender systems adopt optimization-based meta-learning methods, such as model-agnostic meta-learning (MAML) [49], to improve the performance of new tasks. One representative work is MeLU [4] that utilizes MAML to rapidly adapts to new tasks with a few ratings. Another straightforward application of MAML on the recommender system is MetaCS [50], which formulates each user's rating pattern as a task in MAML and designs a recommendation framework based on this. Then, Memory-augmented meta-optimization (MAMO) [26] designs two memory matrices to store task-specific memories for parameter initialization and feature-specific memories for fast predicting users' preferences, respectively. By integrating heterogeneous information, MetaHIN [27] exploits meta-learning to alleviate the cold-start problem at both the data and model levels, which leverages multifaceted semantic contexts and a co-adaptation meta-learner to learn finer grained semantic priors for new tasks in both semantic and task-wise manners. By dynamic subgraph sampling to construct representative training tasks, MetaCF [51] dynamically samples subgraph centered at a user in the training phase to account for the effect of limited interactions under cold-start settings, and extends the historical interactions by incorporating potential interactions to avoid the overfitting problem. It can achieve good performance under cold-start settings. However, the drawback is obvious due to the introduction of potential interaction leading to low efficiency of training the model.

As a promising approach, meta-learning frameworks can fast adapt to new users' or new items' recommendations. However, existing meta-learning based methods suffer poor performance caused by meta-overfitting on sparse interactive data. So in this paper we aim to solve the meta-overfitting problems in meta-learning based recommendations by the proposed diverse preference augmentation technique as introduced in Section I.

III. PRELIMINARIES

In this section, we first describe the basic notations used in this paper and the problem formulation in Section III-A. Then we briefly introduce a general framework of meta-learning based recommender systems in Section III-B.

A. Problem Formulation

In this paper, we aim to provide recommendations in the target domain by transferring preference knowledge from source domains. We suppose there are n users and m items in the target domain, and their index sets are denoted as $U = \{1, \cdots, n\}$ and $I = \{1, \cdots, m\}$, respectively. The available data includes user-item interactive matrix $\mathbf{R} = \{r_{ui} \geq 0 : u \in U, i \in I\}$, the user content data $\mathbf{C}^U$, and the item content data $\mathbf{C}^I$. If user u has an interaction with item i, then $r_{ui} = 1$; otherwise, $r_{ui} = 0$. For each user u, the content data $\mathbf{c}_u \in \mathbf{C}^U$ is composed of bag-of-vectors generated from her/his rated items' review data.

Similarly, the content data of each item $\mathbf{c}_i \in \mathbf{C}^I$ is extracted from all obtained reviews.

In this work, we divide users' set U as U_e and U_n, where we define U_e as '*existing users*', and each $u \in U_e$ represents the user who rates no less than 5 items, i.e., $|\{r_{ui} : u \in U_e\}| \geq 5$. The remaining users $u \in U_n$ are defined as '*new users*' (or cold-start users). Similarly, we divide items' set I as I_e and I_n, where we define I_e as '*existing item*s', and each $i \in I_e$ denotes the item of receiving no less than 5 ratings, i.e., $|\{r_{ui} : i \in I_e\}| \geq 5$, and the remaining items $i \in I_n$ are defined as '*new items*' (or cold-start items). Then, we define the following four recommendation problems including the sparse issue ('Warm-start') and three kinds of cold-start issues solved in this paper:

(1) **Warm-start**: how to improve the accuracy of recommending *existing items* to *existing users* with sparse interactions available? Given sparse ratings R_w of users U_e to items I_e, i.e., $R_w = \{r_{ui} > 0, u \in U_e, i \in I_e\}$, then we train our model $f(\theta) : U_e \times I_e \rightarrow R_w$ on R_w. The goal of 'Warm-start' is to predict unknown ratings $\hat{r}_{ui} \notin R_w$ of *existing users* U_e to *existing items* I_e with the trained model $f(\theta)$ and recommend k *existing items* with top-k ratings to each user in U_e.

(2) **C-U**: how to improve the accuracy of recommending *existing items* to *new users*? After training our model $f(\theta)$ with R_w, we finetune $f(\theta)$ with only a few ratings R_{cu} of *new users* $u \in U_n$ to *existing items* $i \in I_e$. The goal of 'C-U' is to predict unknown ratings $\hat{r}_{ui} \notin R_{cu}$ of *new users* U_n to *existing items* I_e by the finetuned model $f(\theta)$ and recommend k *existing items* with top-k ratings to *new users*.

(3) **C-I**: how to improve the accuracy of recommending *new items* to *existing users*? Similar to **C-U**, we finetune $f(\theta)$ with only a few ratings R_{ci} of *existing users* U_e to *new items* I_n. The goal of 'C-I' is to predict unknown ratings $\hat{r}_{ui} \notin R_{ci}$ of *existing users* U_e to *new items* I_n by the finetuned model $f(\theta)$ and recommend k *new items* with top-k ratings to *existing users*.

(4) **C-UI**: how to improve the accuracy of recommending *new items* to *new users*? We finetune $f(\theta)$ with a few ratings R_{cui} of *new users* U_n to *new items* I_n. The goal of 'C-UI' is to predict unknown ratings $\hat{r}_{ui} \notin R_{cui}$ of *new users* U_n to *new items* I_n with the finetuned model and recommend k *new items* with top-k ratings to *new users*.

B. Meta-learning for Recommendations

The objective of meta-learning is to learn good initial weights for a model that can fast adapt to previously unseen tasks with a few samples [25]. In meta-learning based recommendations [4], [52], a user's preference prediction over all items is treated as a meta-learning task $\mathcal{T}_u$, which is denoted as the dataset used for the task. To be specific, $\mathcal{T}_u = (\mathbf{c}_u, \mathbf{r}_u)$, where $\mathbf{c}_u$ is the input of all items for a specific user u and $\mathbf{r}_u$ is the ratings of a user to all items. The input identifies a user, and it could be her/his identity, profile, historical preference information, or the review comments written by the user. In this paper, we user the review comments as the input. In particular, according to the setting of meta-learning, we divide into a support set $\mathcal{S}_u$ and a query set $\mathcal{Q}_u$. Tasks from all users' preferences are randomly splited into two disjoint partitions: one for meta-training tasks denoted as $\mathbb{T}_{tr}$, and another one for meta-testing tasks denoted as $\mathbb{T}_{te}$. The meta-learning based recommendations aim to learn a good preference prior distribution on meta-training tasks, and then can fast adapt to new tasks in meta-testing tasks.

Suppose a supervised learning problem considers training on a dataset for a single task $\mathcal{T}$. In contrast, meta-learning considers to learn a set of tasks $\mathcal{T}_u \in \mathbb{T}_{tr}$, which are sampled from the task distribution $p(\mathcal{T})$. The objective of meta-training for recommendations is formulated as follows:

$$\min_{\theta} \sum_{\mathcal{T}_u \sim \mathbb{T}_{tr}} \mathcal{L}_{\mathcal{T}_u}(\theta - \alpha \nabla_\theta \mathcal{L}_{\mathcal{T}_u}(\theta, \mathcal{S}_u), \mathcal{Q}_u), \qquad (1)$$

where ∇_θ denotes the gradient w.r.t parameters θ of a preference prediction model, and α is the meta-learning rate, and $\theta - \alpha \nabla_\theta \mathcal{L}_{\mathcal{T}_u}(\theta, \mathcal{S}_u$ is the task-specific parameters adapted to the task $\mathcal{T}_u$ after one gradient step from the global θ. The algorithm locally updates parameters θ based on the gradient with $\mathcal{S}_u$, and then globally updates θ based on $\mathcal{Q}_u$, so that the globally updated parameters adapt to various tasks [4]. During meta-testing, the meta-learner adapts the learned θ to $\mathcal{T}_u \in \mathbb{T}_{te}$ with its support set $\mathcal{S}_u$, and then the adapted parameters θ is used to predict ratings and evaluate the recommendation performance in its query set $\mathcal{Q}_u$ [52].

IV. METHODOLOGY

In this section, we propose a novel method Diverse Preference Augmentation with multiple domains (MetaDPA) for cold-start recommendations as illustrated in Fig. 2. Our method consists of three blocks: *Multi-source Domain Adaptation*, *Diverse Preference Augmentation*, and *Preference Meta-learning*.

Firstly, we conduct multi-source domain adaptation between the source domains and the target domain with Dual CVAEs, and align the representations of the source and target domains by the principle of InfoMax [37] in Section IV-A. Then, we employ the learned encoder-decoders to generate diverse ratings, which is named as diverse preference augmentation in Section IV-B. Next, these augmented diverse ratings together with the original true ratings are fed into the preference meta-learning procedure in Section IV-C. Finally, we test the recommendation performances under three cold-start settings separately.

A. Multi-source Domain Adaptation

In this section, our objective is to conduct domain adaptation from multiple source domains to the target domains. As users'/items' content information is commonly used for alleviating cold-start issues, and the content data including items' images, descriptions and users' profile, user-item review texts. Most of the content data are not shared by different domains. In TDAR [46], it takes the review texts as domain-invariant features to align the latent space for domain adaptation.
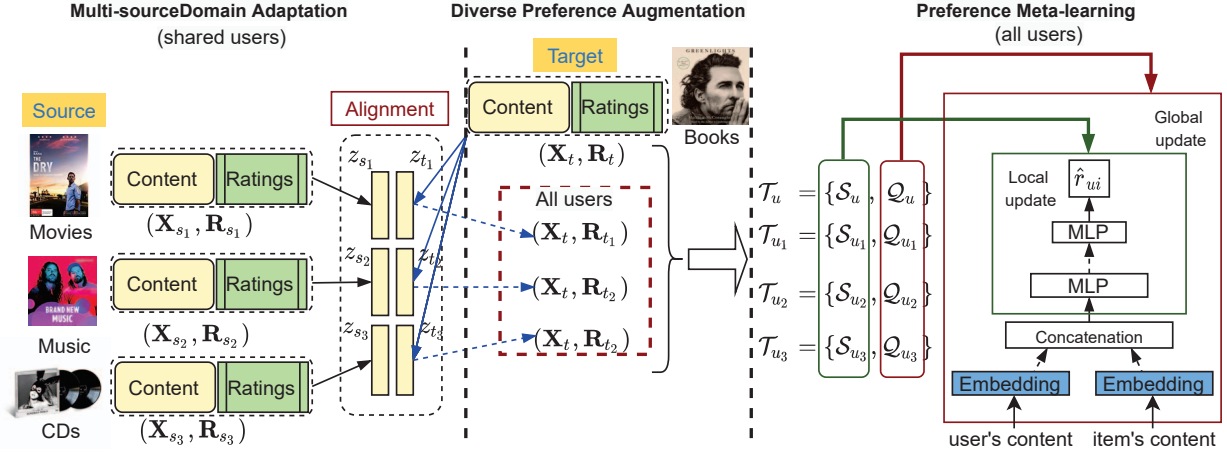
Fig. 2. The framework of diverse preference augmentation with multiple source domains based on meta-learning (MetaDPA) consists of three blocks: Multi-source Domain Adaptation, Diverse Preference Augmentation, and Preference Meta-learning. We first train the framework of multi-source domain adaptation; and then we generate diverse ratings for the target domain given the content data; and finally we learn the preference model formulated as a multi-layer architecture based on the meta-learning optimization framework MAML.

To facilitate the domain adaptation, we also adopt the domain-invariant reviews as the content data. Specifically, the user content $\mathbf{c}_u$ is the collection of reviews from items rated by the user u. We first encode the user's content into a dense low-dimensional embeddings $\mathbf{x}_u$. We denote $\mathbf{x}_u^{(s)}$ and $\mathbf{x}_u^{(t)}$ as the user content vectors in the source domain and in the target domain, respectively. For simplicity, we use $\mathbf{x}_s$ and $\mathbf{x}_t$ to represent them. Similarly, we denote $\mathbf{r}_s$ and $\mathbf{r}_t$ as ratings rated by the shared user u to items in the source domain and the target domain, respectively.

We use a Dual-CVAE network to learn users' latent representations and reconstruct ratings $\mathbf{r}_s$ and $\mathbf{r}_t$. In the Dual-CVAE, we add conditions $\mathbf{x}_s$ ($\mathbf{x}_t$) on the latent representations in a similar way with HCVAE [53]. Firstly, the input $\mathbf{x}_s$ and $\mathbf{x}_t$ are encoded into a distribution $q_{\phi_s}(\mathbf{z}_s|\mathbf{r}_s,\mathbf{x}_s)$ and $q_{\phi_t}(\mathbf{z}_t|\mathbf{r}_t,\mathbf{x}_t)$, respectively. The encoders of the Dual-CVAE maps ratings $\mathbf{r}_s$ and $\mathbf{r}_t$ to latent representations $\mathbf{z}_s$ and $\mathbf{z}_t$, respectively. The optimization objective of the Dual-CVAE is the evidence lower bound (ELOB) [54], which consists of the sum of the reconstruction error, namely, maximizing the likelihood estimation of the decoders $\log p_{\theta_s}(\mathbf{r}_s|\mathbf{z}_s,\mathbf{x}_s)$ and $\log p_{\theta_t}(\mathbf{r}_t|\mathbf{z}_t,\mathbf{x}_t)$, and the negative KL-divergence between the variational posterior and the prior. So the loss function in the source domain and the target domain can be written as follows [53]:

$$
\begin{aligned}
\mathcal{L}_{ELBO} &= \mathcal{L}(\mathbf{r}_s,\mathbf{x}_s;\theta_s,\phi_s) + \mathcal{L}(r_t,\mathbf{x}_t;\theta_t,\phi_t) \\
&= \mathbb{E}_{q_{\phi_s}(\mathbf{z}_s|\mathbf{r}_s,\mathbf{x}_s)}[\log p_{\theta_s}(\mathbf{r}_s|\mathbf{z}_s,\mathbf{x}_s)] \\
&+ \mathbb{E}_{q_{\phi_t}(\mathbf{z}_t|\mathbf{r}_t,\mathbf{x}_t)}[\log p_{\theta_t}(\mathbf{r}_t|\mathbf{z}_t,\mathbf{x}_t)] \\
&- D_{KL}[q_{\phi_s}(\mathbf{z}_s|\mathbf{r}_s,\mathbf{x}_s)||p(\mathbf{z}_s)] \\
&- D_{KL}[q_{\phi_t}(\mathbf{z}_t|\mathbf{r}_t,\mathbf{x}_t)||p(\mathbf{z}_t)].
\end{aligned} \quad (2)
$$

We employ the softmax function as the activation function in the output layer that maps the reconstructed output into the range of [0,1], which is consistent with the range of reconstructed ratings. The reconstruction loss between the predictions and

true ratings could be MSE if the user-item interactions are explicit feedback and binary cross entropy if the user-item interactions are implicit feedbacks [55]. In this paper, we use the implicit feedback as ratings, so we adopt the binary cross-entropy as the reconstruction loss.

The KL divergence loss of the Dual-CVAE can be estimated using the Stochastic Gradient Variational Bayes (SGVB) estimator [56]. Besides, we hope to learn latent representations from the content embeddings $\mathbf{x}_s$ and $\mathbf{x}_t$, and so the learned distributions of latent representations store preference information from both ratings and content, which makes it capable of reconstructing ratings only using content. So we have the following objective:

$$
\begin{aligned}
\mathcal{L}_{KL} =& -\frac{1}{2}\sum_{l=1}^{L}[\sigma_{s_l}^2 + (\mu_{s_l} - \mathbf{z}_{s_l}^x)^2 - \log\sigma_{s_l}^2 - 1] \\
&+ \frac{1}{2}\sum_{l=1}^{L}[\sigma_{t_l}^2 + (\mu_{t_l} - \mathbf{z}_{t_l}^x)^2 - \log\sigma_{t_l}^2 - 1].
\end{aligned} \quad (3)
$$

where L is the dimension of the latent sampled representations. For the source domain, $\mathbf{u}_s$ and σ_s are the mean and variance of the approximate posterior, and $\mathbf{z}_s^x$ is the output of a dense embedding encoder E_s^x, parameterized by ϕ_{x_s}. The notations $\mathbf{u}_t$, σ_t, $\mathbf{z}_t^x$, E_t^x, and ϕ_{x_t} in the target domain have similar meanings.

To reconstruct ratings only using content, we align the latent representations of the Dual-CVAE to the output dense embedding vector of encoder E_s^x by the following mean square error (MSE) loss:

$$
\mathcal{L}_{MSE} = ||\mathbf{z}_s - q_{\phi_{x_s}}(\mathbf{z}_s^x|\mathbf{x}_s)||^2 + ||\mathbf{z}_t - q_{\phi_{x_t}}(\mathbf{x}_t^x|\mathbf{x}_t)||^2. \quad (4)
$$

For domain adaptation, we construct the cross-domain reconstruction objectives by generating ratings $\mathbf{r}_s$ of the source domain with the latent representation of the target domain $\mathbf{z}_t$, and reconstructing ratings $\mathbf{r}_t$ of the target domain with the latent factors of the source domain $\mathbf{z}_s$. To align the latent

representations $\mathbf{z}_s$ and $\mathbf{z}_t$, we have the following cross-domain reconstruction loss,

$$\mathcal{L}_{Rec} = \frac{1}{2m} \sum_{i=1}^{m} [r_{si} \log(\hat{r}_{ti}) + (1 - r_{si}) \log(1 - \hat{r}_{ti})]$$
$$+ \frac{1}{2m} \sum_{i=1}^{m} [r_{ti} \log(\hat{r}_{si}) + (1 - r_{ti}) \log(1 - \hat{r}_{si})], \quad (5)$$

where $\hat{r}_{ti}$ is the reconstructed rating of user u to item i by feeding the latent representation of target domain $\mathbf{z}_t$ into the decoder D_s in the target domain as shown in the Fig. 1, and $\hat{r}_{si}$ is the generated rating of user u to item i by injecting the latent representation of source domain $\mathbf{z}_s$ into the decoder of target domain D_t.

To maximize the mutual information from the source and target domains and preserve the domain-specific properties, we impose the MDI constraint with the InfoMax principle [57] on sampled latent representations $\mathbf{z}_s$ and $\mathbf{z}_t$. Since the MDI constraint does not discard superfluous information of each domain for predicting ratings. So it allows the reconstructed user's preference to have the capacity of preserving the domain-specific properties as well as the domain-shared properties. The MDI constraint is written as follows:

$$\mathcal{L}_{MDI}(\phi_s, \phi_t) = -I_{\phi_s, \phi_t}(\mathbf{z}_s, \mathbf{z}_t), \quad (6)$$

where the function $I(\cdot)$ denotes the mutual information between inputs [37]. The MDI is employed here because we consider each user's latent preferences $\mathbf{z}_s$ and $\mathbf{z}_t$ encoded from different domains should have both domain-shared and domain-specific properties. So the MDI advocates the decoders of each domains to generate domain-specific ratings, which is consistent with the goal of generating diverse ratings by the content in the target domain.

B. Diverse Preference Augmentation

As introduced in Section I, the goal of diverse preference augmentation is to generate diverse ratings by the learned encoder-decoders (highlighted with red line shown in Fig. 1) of target domain given the input $\mathbf{x}_t$. If there are k source domains, we have k Dual-CVAEs for the multi-source domain adaptation, then we generate k ratings for each user in the target domain by his/her content $\mathbf{x}_t$. Due to the integrated effect of MDI constraint imposed on the latent representations, the reconstruction loss (2) and the cross-domain reconstruction loss (5), the Dual-CVAE can learn domain-shared preference information well enough. Thus, these generated k ratings from k Dual-CVAEs are similar because all source domains adapts to the target domain. To avoid overfitting, we impose the ME constraints on the outputs of k decoders D_t to generate ratings as different as possible from each other. With the ME constraint, we force the generated rating from the decoder D_t as close to the reconstructed ratings as possible from the decoder D_s. Thus, the generated ratings preserve domain-specific preference patterns of multiple source domains, so we can generate diverse ratings by k encoder-decoder frameworks of Dual-CVAEs. The ME constraint is realized by maximizing the mutual information

between two generated ratings $\mathbf{r}_s$ and $\mathbf{r}_t$ of a shared user u as follows:

$$\mathcal{L}_{ME} = -I(\mathbf{r}_s, \mathbf{r}_t), \quad (7)$$

where $I(\cdot)$ denotes the mutual information between two inputs, that is implemented by InfoNCE [58].

By enforcing ME and MDI constraints on the Dual-CVAE, the objective of cross-domain adaptation can be derived by summing up objectives (2), (4), (5), and two constraints (6), (7) together. Single-source cross-domain adaptation is one special case of multi-source cross-domain adaptation. We can obtain the multi-source cross-domain objective by integrating the following cross-domain objective together,

$$\mathcal{L}_{Dual\text{-}CVAE} = \mathcal{L}_{ELBO} + \mathcal{L}_{MSE} + \mathcal{L}_{Rec} + \beta_1 \mathcal{L}_{MDI} + \beta_2 \mathcal{L}_{ME}. \quad (8)$$

We train the Dual-CVAE shown in Fig. 1 by minimizing the objective (8) of cross-domain adaptation. The multi-source cross-domain adaptation can be implemented by training multiple Dual-CVAEs in parallel. Suppose we have k source domains, then we learn k Dual-CVAEs independently. After that, we obtain the learned k encoders E_t^x and k decoders D_t, and then employ them to generate k diverse ratings $\mathbf{r}_{t1}, \mathbf{r}_{t2}, \cdots, \mathbf{r}_{tk}$ with the content $\mathbf{x}_t$. It's worth note that these generated ratings are in the continuous scale of $[0, 1]$ because we focus on implicit feedback in this paper.

By taking users' preferences / ratings prediction as meta-learning tasks, the meta-learner trained on these generated ratings and original ratings is expected to avoid overfitting as introduced in Section I. In Section II, we know that the meta-learning optimization scheme MAML [49] can fast adapt to new tasks. In recommendations, the meta-learner can fast adapt to cold-start recommendations with MAML.

C. Preference Meta-learning

This block focuses on training a preference prediction model based on meta-learning in the target domain by generated diverse ratings $\mathbf{r}_{t1}, \mathbf{r}_{t2}, \cdots, \mathbf{r}_{tk}$. Firstly, we encode the original content $\mathbf{c}_u$ and $\mathbf{c}_i$ to their dense embedding $\mathbf{x}_u$ and $\mathbf{x}_i$. We denote $\mathbf{c}_t$ as the combination of $\mathbf{c}_u$ and $\mathbf{c}_i$ in the following paper. Then, we construct the meta-learning task $\mathcal{T}_u$ and the augmented tasks $\mathcal{T}_u^1, \cdots, \mathcal{T}_u^k$ as follows:

$$\mathcal{T}_u = (\mathbf{c}_t, \mathbf{r}_t) \quad (9)$$
$$\mathcal{T}_{u_1} = (\mathbf{c}_t, \hat{\mathbf{r}}_{t1}), \cdots, \mathcal{T}_{u_k} = (\mathbf{c}_t, \hat{\mathbf{r}}_{tk}), \quad (10)$$

where $\mathbf{r}_t$ is the original ratings in the target domain. Next, we learn the preference prediction model via the meta-learning framework MAML.

As shown in Fig. 2, we firstly employ a fully connected embedding layer to encode content vectors $\mathbf{c}_u$ and $\mathbf{c}_i$ into dense embeddings $\mathbf{x}_u$ and $\mathbf{x}_i$, and then we adopt a multi-layer neural architecture [29] to predict rating scores by the concatenation of $\mathbf{x}_u$ and $\mathbf{x}_i$. As implicit feedback we considered in this paper, so we use the binary cross-entropy loss on the top of multi-layer neural network to predict ratings.

The embedding layer and the multi-layer architecture constitute the preference prediction model. We train the model

TABLE I
STATISTICS OF DATASETS FOR MULTIPLE SOURCE DOMAINS.

Source (S)	#shared users (T)		#items	#ratings	sparsity
	Books	CDs			
Electronics	28,505	6,260	63,001	1,687,993	99.98%
Movies	37,387	18,031	50,052	1,697,438	99.97%
Music	1,952	5,331	3,568	64,705	99.67%

TABLE II
STATISTICS OF DATASETS FOR TARGET DOMAINS.

Datasets	#users	#items	#ratings	sparsity
Books	603,374	348,957	8,575,000	99.99%
CDs	25,400	24,904	43,903	99.99%

with diverse tasks to avoid overfitting via the meta-learning framework MAML. The objective can be written as:

$$\hat{\mathbf{r}}_{ui} = f(\theta_l, \theta_e, \mathbf{c}_u, \mathbf{c}_i), \tag{11}$$

where θ_e is the parameters of the fully connected embedding layer that encodes the content vector $\mathbf{c}_u$ and $\mathbf{c}_i$ into dense embeddings $\mathbf{x}_u$ and $\mathbf{x}_i$, and θ_l is the parameters of the multi-layer neural network.

As shown in Fig. 2, MAML includes the inner loop for local update of the model and the outer loop for global update of the model. In this work, we train the MAML on training task set $\mathbb{T}_{tr}$, which includes the original tasks $\mathcal{T}_u$ and augmented tasks $\mathcal{T}_{u1}, \cdots, \mathcal{T}_{uk}$. We divide samples of each task $\mathcal{T} \in \mathbb{T}_{tr}$ randomly into a support set $\mathcal{S}_u$ and a query set $\mathbf{Q}_u$ as follows:

$$\mathcal{T} = \{\mathcal{S}_u, \mathcal{Q}_u\}. \tag{12}$$

In the meta-testing phase, meta-testing tasks $\mathcal{T} \in \mathbb{T}_{te}$ includes only original tasks constituted from original ratings of the target domain. Samples of each $\mathcal{T}$ are also split into a support set $\mathcal{S}_u$ for fine-tuning the preference model and a query set $\mathbf{Q}_u$ for testing the recommendation performance.

D. Time Complexity Analysis

Our model consists of three blocks (Fig.2). (1) the Dual-CVAE (Fig. 1) is trained in parallel for multiple source domains. It encodes users' ratings and users' content into latent vectors $\mathbf{z}_s, \mathbf{z}_t$ with 2-layer networks and then decodes the latent vectors into users' ratings with 2-layer networks. The dimensions of users' content, hidden layers and the latent vectors are constant with the data size $\mathcal{O}\{\max\{n, m\}\}$, where n and m are numbers of users and items. If we denote B, l and m as the batch size, the numbers of items in the source and target domains, then the dimensions of inputs $\mathbf{r}_s$ and $\mathbf{r}_t$ are l and m, respectively. So the time complexity is $\mathcal{O}(B(l+m))$. (2) The second block forwards the encoder and decoder (red line in Fig. 1) one-pass to generate ratings, so the time complexity is $\mathcal{O}(B)$. (3) The third block learns a 2-layer network, and the dimension of users' (items') content is constant with the data size, so the time complexity is $\mathcal{O}(B)$. Overall, the time complexity is $\mathcal{O}(B(l+m))$, so it scales linearly with the data size.

V. EXPERIMENTS

In this work, we claim that (1) diverse preference augmentation can handle overfitting in the case of sparse interactions

to improve the performance of four recommendation problems defined in Section III-A; (2) learning a preference model via a meta-learning scheme can alleviate cold-start issues. The goal of experiments is to evaluate the effectiveness of the proposed MetaDPA to avoid overfitting and solve the recommendation problems including 'Warm-start', 'C-U', 'C-I', and 'C-UI'. In addition, the experiments study the impact of two constraints added in the model. To be specific, we analyze the effectiveness from the following aspects:

- **RQ1:** Does our method outperforms the state-of-the-art cross-domain baselines?
- **RQ2:** Does our method handle overfitting on sparse interactions?
- **RQ3:** How is the scalability of our method?
- **RQ4:** How do the two constraints MDI and ME affect the performances of recommendations under different settings?
- **RQ5:** How do the hyper-parameters, such as β_1 of MDI, and β_2 of ME, affect the effectiveness of DPA?

A. Experimental Settings

1) Datasets: We evaluate the performance of our method against the state-of-the-art baselines on public Amazon datasets[1]. Among the largest categories, we choose Electronics, Movies and Music as three source domains, and Books and CDs as two target domains. As this paper focuses provide recommendations based on implicit feedbacks, we transform explicit ratings greater than 0 as the positive feedbacks '1' and others as negative feedbacks '0'. Statistics for the source domains and the target domains are shown in Table I and Table II, respectively.

Experiments of our method contains three phases: (1) multi-source domain adaptation from three source to two target domains, (2) diverse preference augmentation, and (2) preference meta-learning in the two target domains. In the first phase, we train three independent Dual-CVAEs of multi-source domain adaptation. In this phase, we discard users or items with fewer than 20 positive ratings for both source and target domains. We randomly split ratings into 80% as training set and the remaining 20% as evaluation set for the domain adaptation. Then, we generate diverse ratings from three decoders of Dual-CVAEs for users in the target domains by their content data. In the third phase, we train preference model with the generated ratings and the original ratings in the target domains, and test the performances of four problems defined in Section III-A.

2) Evaluation Protocols: In this paper, we only test performances on target domains. Recommendation problems consist of 'C-U', 'C-I', 'C-UI', and 'Warm-start' introduced in Section III-A. We first train our model on training tasks $\mathcal{T}_{tr}$ constructed of ratings in $\mathbf{R}_w$. For solving 'Warm-start', we test the performance on the query set of $\mathcal{T}_{tr}$. For cold-start problems 'C-U', 'C-I', and 'C-UI', we first finetune the trained model on training tasks $\mathcal{T}_{tr}$ by the support set of testing tasks $\mathcal{T}_{te}$, and then we test the performance on the query set of $\mathcal{T}_{te}$.
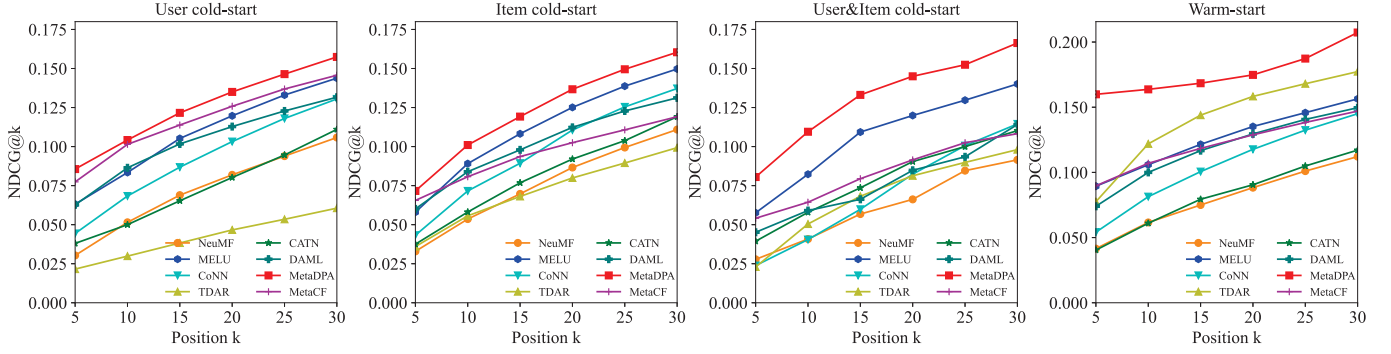
[1]http://jmcauley.ucsd.edu/data/amazon/

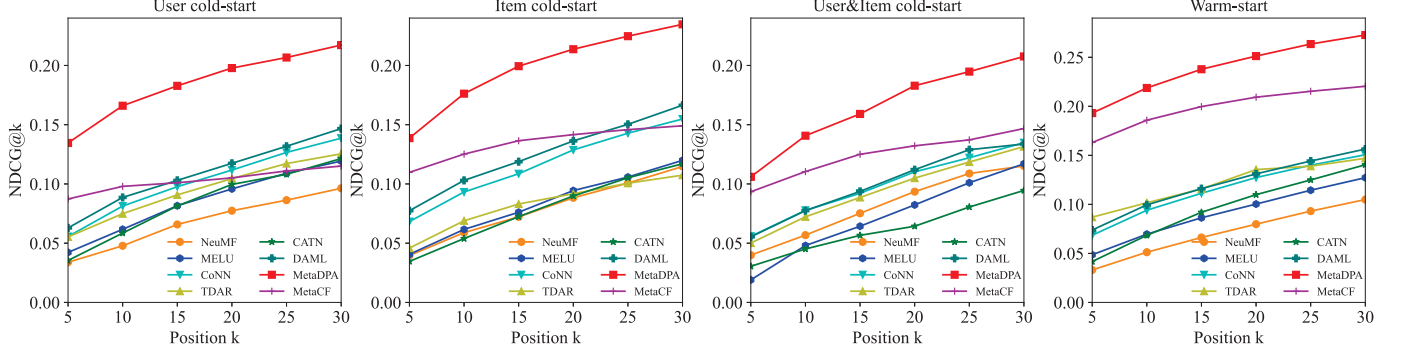Fig. 3. Performance comparison on Books.

Fig. 4. Performance comparisons on CDs.

Taking 'C-U' as an example, $\mathcal{T}_{te}$ is constructed from ratings $\{r_{ui} : u \in U_n, i \in I_e\}$. Similarly, we can construct testing tasks $\mathcal{T}_{te}$ for other two cold-start settings.

Following the common strategy of implicit feedbacks [29], we adopt the similar *leave-one-out* evaluation protocols for evaluating the performance of recommendations. Specifically, we sample 99 negative unobserved items for each positive item to evaluate MetaDPA and other baselines. Then we report the performance of top-k recommendations under four common metrics: *hit ratio* (HR) [59], *mean reciprocal rank* (MRR) [60], *normalized discounted cumulative gain* (NDCG) [4], and *area under ROC curve* (AUC) [61].

3) Baselines: To investigate whether MetaDPA can improve the recommendation performance under cold-start settings or not, we compare it with several state-of-the-art baselines, including the competitive NeuMF, content-aware recommender systems (CoNN, DAML), meta-learning based methods (Melu, MetaCF), cross-domain recommendation frameworks (TDAR, CATN).

- **NeuMF:** Neural collaborative filtering [29] is the most favorite technique in recommender systems.
- **MeLU:** Meta-learned user preference estimator [62] demonstrates satisfactory performance when applied to a wide range of users for providing personalized recommendations due to the generalization ability.
- **MetaCF:** Fast adaptation for recommendations with meta-learning [51] is formulated with a dynamic subgraph sampling that accounts for the dynamic arrival of new users by dynamically generating representative adaptation tasks for existing users.

- **CoNN:** Deep cooperative neural networks [39] learns item properties and user behaviors from reviews consisting of two parallel neural networks with the shared last layer.
- **DAML:** Dual attention mutual learning [40] adapts local and mutual attention network to extract the rating and review features. With such features, neural factorization machine can effectively make predictions.
- **TDAR:** Text-enhanced domain adaptation recommendation (TDAR) [46] extracts the textual features in word semantic space for each user and item and feds them into a collaborative filtering model for predicting ratings.
- **CATN:** Cross-domain recommendation framework via aspect transfer network [45] learns cross-domain aspect-level preference matching by bridging multiple user's inherent traits via reviews in different domains.

4) Hyper-parameter Settings: To obtain the optimal hyper-parameters β_1 and β_2 in Eq.(8), we apply the grid search in the range of $\{1e^{-2}, 1e^{-1}, 1, 1e^1, 1e^2\} \bigotimes \{1e^{-2}, 1e^{-1}, 1, 1e^1, 1e^2\}$ on Books and CDs, respectively. After searching, we set $\beta_1 = 0.1$, $\beta_2 = 1$ on CDs, and $\beta_1 = 0.1$, $\beta_2 = 1$ on Books, because these settings can achieve best performance.

B. Overall Performance Compared with Baselines (**RQ1 & RQ2**)

We compare our model MetaDPA with other competitive baselines to show the performance improvements under NDCG@k in Fig. 3 and Fig. 4 on Books and CDs, respectively. The results under other metrics are shown in Table V-A3. We conclude that the performance of the proposed MetaDPA is significantly superior to all competing baselines. The over-

TABLE III
THE OVERALL PERFORMANCE COMPARISON OF METADPA (THE PROPOSED MODEL) AND BASELINES. BEST PERFORMANCE MARKED AS BOLD FONT, THE SECOND BEST MARKED AS ^{o}.

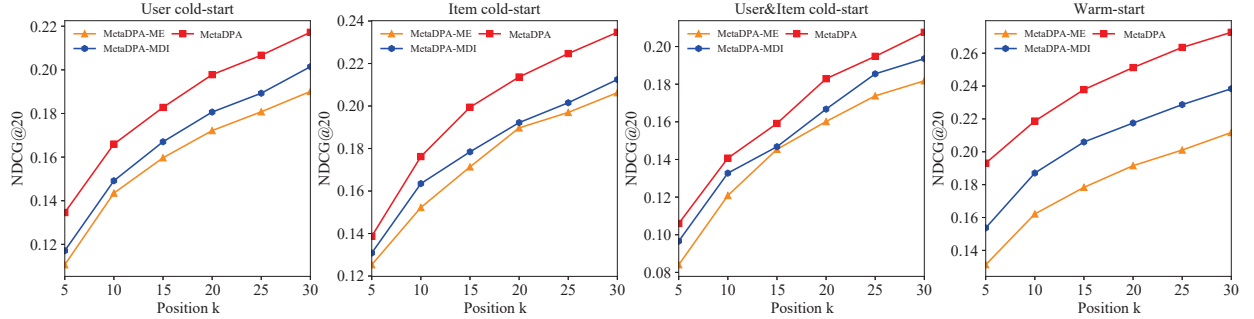| Dataset		Books				CDs			
Metric		HR@10	MRR@10	NDCG@10	AUC	HR@10	MRR@10	NDCG@10	AUC
C-U	NeuMF	0.1159	0.0327	0.0515	0.5009	0.1069	0.0300	0.0478	0.5054
	MeLU	0.1524	0.0630	0.0834	**0.5819**	0.1278	0.042	0.0617	0.5635
	CoNN	0.1515	0.0437	0.0683	0.5810	0.1696	0.0549	0.0813	0.5853
	TDAR	0.0641	0.0195	0.0298	0.3852	0.1417	0.0549	0.0748	0.5451
	CATN	0.1071	0.0329	0.0500	0.5376	0.1346	0.0364	0.0585	0.5910^o
	DAML	**0.1756**	0.0596	0.086	0.5454	0.1789^o	0.0618	0.0886^o	0.5876
	MetaCF	0.1607	0.0746^o	0.1014^o	0.5383	0.1535	0.0807^o	0.0979	0.5024
	MetaDPA	0.1702^o	**0.0848**	**0.1042**	0.5803^o	**0.2648**	**0.1371**	**0.1660**	**0.6364**
C-I	NeuMF	0.1248	0.0329	0.0536	0.5043	0.117	0.0416	0.0589	0.5054
	MeLU	0.1846^o	0.0610	0.0891^o	0.5977	0.1326	0.0404	0.0616	0.577
	CoNN	0.1660	0.0438	0.0716	0.6014^o	0.1871	0.0651	0.09319	0.6092
	TDAR	0.1275	0.0343	0.0556	0.5169	0.1423	0.0472	0.0688	0.5324
	CATN	0.1280	0.0373	0.0581	0.5628	0.1195	0.0347	0.0539	0.5670
	DAML	0.1766	0.0561	0.0839	0.5466	0.2008^o	0.0738	0.1030	0.6143^o
	MetaCF	0.1393	0.0631^o	0.0807	0.5298	0.1930	0.1041^o	0.1252^o	0.5108
	MetaDPA	**0.2178**	**0.0775**	**0.1101**	**0.6392**	**0.3119**	**0.1359**	**0.1762**	**0.6808**
C-UI	NeuMF	0.0843	0.0277	0.0408	0.5387	0.1240	0.0371	0.0568	0.5144
	MeLU	0.1742^o	0.0621^o	0.0872^o	0.5699^o	0.1240	0.0265	0.048	0.5733
	CoNN	0.1011	0.0223	0.0404	0.5493	0.1473	0.0570	0.0776	0.6045
	TDAR	0.1180	0.0308	0.0504	0.5076	0.1705	0.0809	0.1016	0.5560
	CATN	0.1209	0.0391	0.0579	0.5315	0.0916	0.0314	0.0451	0.5256
	DAML	0.1292	0.0379	0.0590	0.5340	0.1550	0.0543	0.0774	0.6053^o
	MetaCF	0.1236	0.0462	0.0644	0.5351	0.1860^o	0.0876^o	0.1104^o	0.5221
	MetaDPA	**0.1910**	**0.0694**	**0.0971**	**0.5921**	**0.2558**	**0.1069**	**0.1407**	**0.6515**
Warm-start	NeuMF	0.1292	0.0415	0.0616	0.5107	0.1106	0.0336	0.0512	0.5014
	MeLU	0.1512	**0.0929**	0.1059	0.5562	0.1470	0.0465	0.0696	0.5826
	CoNN	0.1773	0.0528	0.0813	0.6023	0.1922	0.0645	0.0940	0.6029
	TDAR	**0.2798**	0.0749	0.1219^o	**0.6331**	0.1501	0.0489	0.0722	0.5843
	CATN	0.1363	0.0386	0.0610	0.5539	0.1505	0.0440	0.0684	0.6183^o
	DAML	0.2011	0.0696	0.1000	0.5710	0.2007	0.0691	0.0995	0.6062
	MetaCF	0.1775	0.0857^o	0.1070	0.5127	0.2084^o	0.1511^o	0.1859^o	0.5472
	MetaDPA	0.2099^o	0.0829	**0.1396**	0.6211^o	**0.2866**	**0.1994**	**0.2187**	**0.6481**

Fig. 5. Effectiveness of Mutually-Exclusive (ME) and Multi-domain InfoMax (MDI) constraints.

whelming advantages over baselines attribute to the following three points:

(1) Compared with cross-domain recommendations CATN and TDAR: By transferring preference properties from multiple source domains to the target domain via augmented diverse ratings, the proposed model can capture more preference properties, which is helpful to avoid overfitting and improve the recommendation accuracy, so it performs much better than cross-domain recommender systems;

(2) Compared with content-aware recommendations CoNN and DAML: The diverse preference augmentation block aims to reduce the gap between content and preference that exists in content-aware recommender systems, so most of the time, it performs better than content-aware

recommender systems;

(3) Compared with meta-learning based recommender systems Melu and MetaCF: With the diverse ratings fed into the preference meta-learning, MetaDPA avoids overfitting to the insufficient interactive training set, so it performs much better than meta-learning based recommender systems for solving recommendation problems.

Specifically, both CoNN and DAML are content-aware recommendations with deep models. The difference is that CoNN uses a parallel neural network to learn user behaviors and item properties while DAML learns rating and review features by local and mutual attention networks. In our experiments, DAML shows a little better than CoNN. Both methods show middle performance in all scenarios. Actually, this indicates that content information can effectively deal with cold-start
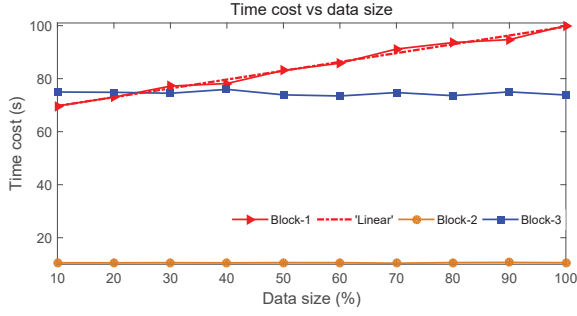
Fig. 6. Training time cost varies with Data size.

problems and work well in warm-start scenarios.

Besides, TDAR and CATN perform worse than content-based methods CoNN and DAML in most scenarios. However, the performance of TDAR is unstable. In some cases, it performs very well, e.g., the warm-start scenario on Books and CDs. The possible reason is that TDAR is designed for warm-start recommendations instead of cold-start settings, and the training datasets are very sparse for cold-start users/items, so it performs inferior to other baselines under cold-start settings and achieves better performance under warm-start settings.

In addition, MeLU and MetaCF are effective meta-learning recommendations that are designed for solving cold-start issues with meta-learning frameworks. Both of them demonstrate powerful performances, which outperforms almost all other baselines in cold-start settings, and both perform well in the warm-start scenarios on Books as well. However, MeLU performs badly in all scenarios on CDs. A possible reason is that it easily sticks into the serious meta-overfitting on sparse user-item interactions. In contrast, MetaCF performs much better than other baselines on CDs, which indicates that incorporating potential interactions by neighborhood users / items can enrich users' preference information.

To explore the reasons why the performances of other baselines' on CDs are worse than the performances on Books, we calculate the proportion of users who have no less than 40 interactions, and we obtain 13% for CDs and 16% for Books, which may account for the different results of baselines. We are known that the recommendation performance is heavily dependent on the rating sparsity. When we learn models by a batch of users, the performances rely on the rating sparsity of a batch of users. If more users rated more items (such as more than 40 items), the recommendation problems would converges to a better solution, so the performances would be better.

Overall, under both cold-start and warm-start scenarios, MetaDPA outperforms all baselines significantly w.r.t NDCG@k metric. Regarding other metrics, MetaDPA outperforms almost all baselines except HR@10 and AUC for solving user cold-start problem ('C-U'), MRR@10 for solving 'Warm-start' issue on Books. However, MetaDPA obtains the second-best performance on these problems. This indicates that meta-learning-based recommendations significantly improve cold-start recommendations and warm-start recommendations by addressing the meta-overfitting problem.

C. Scalability (RQ3)

In this section, we run a series of experiments on the source domain Electronics and the target domain Books. To evaluate the scalability, we choose items in Books randomly with different percentages, 10%, 20%, ..., 100% to create 10 group new datasets. To speed up the computation, we train our model on GPU platform (NVIDIA GeForce RTX 3090). In the experiments, we set batch size $B = 32$ heuristically similar to the work [63], and we report the training time costs of 1 epoch in each block. As shown in Fig. 6, the training procedure scales linearly with the data size for the first block (Block-1 in Fig. 6), and the training time costs are constant with the data size for the second block (Block-2) and third block (Block-3). Which is consistent with the time analysis in Section IV-D. So the proposed framework can be extended to larger datasets.

D. Significance Test

We use the one-sided test, Wilcoxon signed-rank test [64], to test the significance of our method surpassing the second-best methods, which has the null hypothesis that the median of the differences of two results $x_i - y_i$ under an evaluation metric (x_i and y_i denote the results of our method and the second-best method) is negative against the alternative that it is positive. For different metrics, we obtain different p-values. By randomly splitting training set and testing set 30 times independently, we obtain two sets of 30 results for our method and the second-best method, respectively.

By comparing with the second-best method MeLU, we test the significance on Books and get p-values are $5.96e^{-8}, 1.23e^{-5}, 1.78e^{-7}, 5.96e^{-8}$ for HR@10, MRR@10, NDCG@10, AUC under cold-start user ('C-U'). Similarly, for cold-start item ('C-I'), p-values are $5.96e^{-8}, 2.98e^{-7}, 5.96e^{-8}, 5.96e^{-8}$. For 'C-UI' and 'Warm-start', all p-values are $5.96e^{-8}$ except p-value is $2.15e^{-4}$ under AUC for 'Warm-start' setting. Besides, we get all p-values are $1.19e^{-7}$ on CDs under all recommendation settings. Similarly, we obtain similar results when we compare the second-best methods MetaCF, DAML, TDAR, and CATN under specific settings. So we conclude that our method significantly outperforms other baselines with p-values<0.05 under all evaluation metrics.

E. Ablation Studies (RQ4)

In this subsection, we discuss the effectiveness of MDI and ME constraints. We test two variants of MetaDPA to validate the effectiveness of the two constraints.

- **MetaDPA-ME**: MetaDPA only with the ME constraint.
- **MetaDPA-MDI**: MetaDPA only with the MDI constraint.

MetaDPA-ME only considers making the preference distribution of the target domain close to the source and ignores how to learn domain-specific and domain-shared preference properties for domain adaptation. MetaDPA-ME generates more diverse but less meaningful ratings for the target domain. In contrast, MetaDPA-MDI considers learning domain-specific and domain-shared properties as other cross-domain methods and ignores
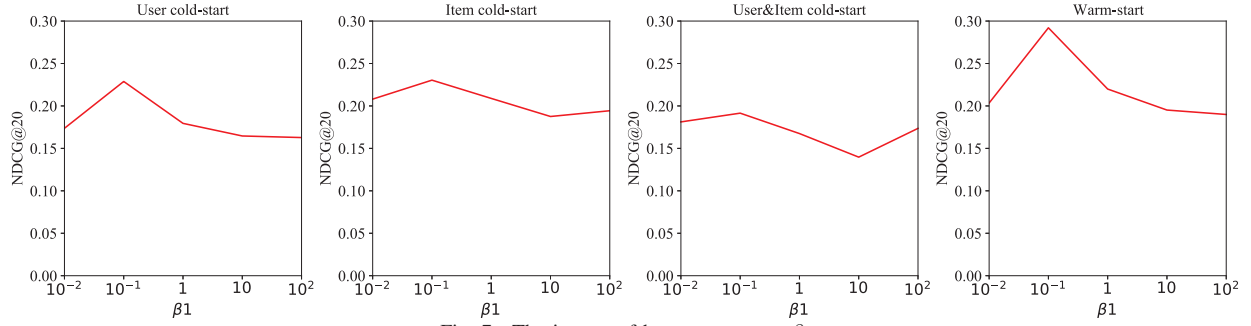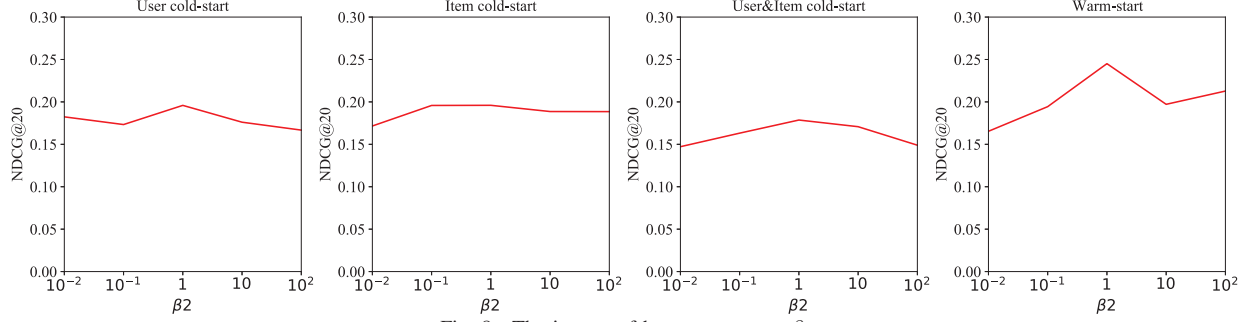
Fig. 7. The impact of hyper-parameter β_1.

Fig. 8. The impact of hyper-parameter β_2.

diverse requirements. The results on all four scenarios on CDs are shown in Fig. 5. Obviously, both MetaDPA-ME and MetaDPA-MDI suffer performance declines. MetaDPA-ME performs worst due to it generating meaningless ratings and introducing useless bias to the meta-training tasks. MetaDPA-MDI performs worse than MetaDPA because it generates ratings that are too close to the true ratings and lack diversity. However, both variants still perform better than MeLU and other baselines since the Dual-CVAE framework brings weak but good enough diversity and domain adaptation.

F. The Impact of Hyper-parameters (**RQ5**)

In this subsection, we test the sensitivity of our framework w.r.t the hyper-parameters β_1 and β_2 on CDs by grid search introduced in Section V-A. The results for β_1 and β_2 are respectively shown in Fig. 7 and Fig. 8. We conclude that β_1 is more sensitive than β_2. Actually, β_1 and β_2 weigh the importances of constraints MDI and ME, respectively. MDI affects both domain adaptation and diverse ratings generation; while ME only affects the latter, so the variation of β_1 affects the recommendation performance more seriously than β_2, i.e., β_1 is more sensitive than β_2. Besides, we find that warm-start scenarios are more sensitive than cold-start scenarios. As introduced in Section V-A, we use a few samples to finetune the model (trained in the warm-start setting) for cold-start testings, so it has a better generalization ability than that for warm-start testing. That's why warm-start scenarios are more sensitive than cold-start scenarios.

VI. CONCLUSION

In this paper, we propose a Diverse Preference Augmentation based on meta-learning (MetaDPA) method by multiple source domains for solving cold-start issues in recommendation tasks. The proposed MetaDPA consists of three blocks. The first block is multi-source domain adaptation, formulated with multiple Dual-CVAEs. To preserve both domain-shared and domain-specific preference properties in the latent space, we add the MDI constraint with the principle of Info-Max, which is effective to maintain domain-shared properties without discarding domain-specific information. Besides, we impose the ME constraint on the outputs of decoders for generating diverse ratings from different source domains. The second block is diverse preference augmentation, which is realized by feeding the content data in the target domain into the content encoder E_t^x and the decoder D_t. The last block, preference meta-learning, optimizes the preference model which is formulated with a multi-layer neural network based on the model-agnostic meta-learning scheme. Experimental results clearly show that MetaDPA significantly outperforms state-of-the-art baselines including content-aware, cross-domain, and meta-learning methods on public datasets. Besides, we conduct ablation studies to evaluate the effectiveness of two constraints MDI and ME. Finally, we also evaluate the impacts of hyper-parameters β_1 and β_2 on the recommendation performance.

REFERENCES

[1] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," in *The adaptive web*. Springer, 2007, pp. 291–324.

[2] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proceedings of the 26th international conference on world wide web*, 2017, pp. 173–182.

[3] L. Zou, L. Xia, Y. Gu, X. Zhao, W. Liu, J. X. Huang, and D. Yin, "Neural interactive collaborative filtering," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 749–758.

[4] H. Lee, J. Im, S. Jang, H. Cho, and S. Chung, "Melu: meta-learned user preference estimator for cold-start recommendation," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 1073–1082.

[5] C. Wang and D. M. Blei, "Collaborative topic modeling for recommending scientific articles," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2011, pp. 448–456.

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

[7] D. Agarwal and B.-C. Chen, "Regression-based latent factor models," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2009, pp. 19–28.

[8] S. Rendle, "Factorization machines with libfm," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 3, no. 3, pp. 1–22, 2012.

[9] D. Lian, Y. Ge, F. Zhang, N. J. Yuan, X. Xie, T. Zhou, and Y. Rui, "Content-aware collaborative filtering for location recommendation based on human mobility data," in *2015 IEEE international conference on data mining*. IEEE, 2015, pp. 261–270.

[10] H. Zhao, Q. Yao, J. T. Kwok, and D. L. Lee, "Collaborative filtering with social local models," in *2017 IEEE international conference on data mining (ICDM)*. IEEE, 2017, pp. 645–654.

[11] H. Chen, H. Yin, X. Sun, T. Chen, B. Gabrys, and K. Musial, "Multi-level graph convolutional networks for cross-platform anchor link prediction," in *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, 2020, pp. 1503–1511.

[12] J. Ding, G. Yu, Y. Li, D. Jin, and H. Gao, "Learning from hometown and current city: Cross-city poi recommendation via interest drift and transfer learning," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 3, no. 4, pp. 1–28, 2019.

[13] H. T. Nguyen, M. Wistuba, and L. Schmidt-Thieme, "Personalized tag recommendation for images using deep transfer learning," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2017, pp. 705–720.

[14] M. He, J. Zhang, P. Yang, and K. Yao, "Robust transfer learning for cross-domain collaborative filtering using multiple rating patterns approximation," in *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, 2018, pp. 225–233.

[15] L. Guo, L. Tang, T. Chen, L. Zhu, Q. V. H. Nguyen, and H. Yin, "Da-gcn: A domain-aware attentive graph convolution network for shared-account cross-domain sequential recommendation," *arXiv preprint arXiv:2105.03300*, 2021.

[16] I. Cantador, I. Fernández-Tobías, S. Berkovsky, and P. Cremonesi, "Cross-domain recommender systems," in *Recommender systems handbook*. Springer, 2015, pp. 919–959.

[17] J. Lian, F. Zhang, X. Xie, and G. Sun, "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*, 2017, pp. 817–818.

[18] A. P. Singh and G. J. Gordon, "Relational learning via collective matrix factorization," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2008, pp. 650–658.

[19] D. Cortes, "Cold-start recommendations in collective matrix factorization," *arXiv preprint arXiv:1809.00366*, 2018.

[20] D. Kim, S. Kim, H. Zhao, S. Li, R. A. Rossi, and E. Koh, "Domain switch-aware holistic recurrent neural network for modeling multi-domain user behavior," in *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, 2019, pp. 663–671.

[21] Y. Zhang, B. Cao, and D.-Y. Yeung, "Multi-domain collaborative filtering," *arXiv preprint arXiv:1203.3535*, 2012.

[22] B. Yi, S. Zhao, X. Shen, and L. Zhang, "Multi-domain neural network recommender," in *2018 IEEE International Conference on Electronics and Communication Engineering (ICECE)*. IEEE, 2018, pp. 41–45.

[23] H. Yan, C. Yang, D. Yu, Y. Li, D. Jin, and D. M. Chiu, "Multi-site user behavior modeling and its application in video recommendation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, no. 1, pp. 180–193, 2019.

[24] X. Zhang, J. Cheng, S. Qiu, Z. Zhu, and H. Lu, "When personalization meets conformity: Collective similarity based multi-domain recommendation," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2015, pp. 1019–1022.

[25] M. Vartak, A. Thiagarajan, C. Miranda, J. Bratman, and H. Larochelle, "A meta-learning perspective on cold-start recommendations for items," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, 2017, pp. 6904–6914.

[26] M. Dong, F. Yuan, L. Yao, X. Xu, and L. Zhu, "MAMO: memory-augmented meta-optimization for cold-start recommendation," in *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, 2020, pp. 688–697.

[27] Y. Lu, Y. Fang, and C. Shi, "Meta-learning on heterogeneous information networks for cold-start recommendation," in *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, 2020, pp. 1563–1573.

[28] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM Computing Surveys (CSUR)*, vol. 52, no. 1, pp. 1–38, 2019.

[29] X. He and T.-S. Chua, "Neural factorization machines for sparse predictive analytics," in *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*, 2017, pp. 355–364.

[30] H.-J. Xue, X. Dai, J. Zhang, S. Huang, and J. Chen, "Deep matrix factorization models for recommender systems." in *IJCAI*, vol. 17. Melbourne, Australia, 2017, pp. 3203–3209.

[31] L. Zhao, S. J. Pan, and Q. Yang, "A unified framework of active transfer learning for cross-system recommendation," *Artificial Intelligence*, vol. 245, pp. 38–55, 2017.

[32] Q. Wang, H. Yin, H. Wang, Q. V. H. Nguyen, Z. Huang, and L. Cui, "Enhancing collaborative filtering with generative augmentation," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 548–556.

[33] M. Dong, F. Yuan, L. Yao, X. Xu, and L. Zhu, "Mamo: Memory-augmented meta-optimization for cold-start recommendation," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 688–697.

[34] X. Lin, J. Wu, C. Zhou, S. Pan, Y. Cao, and B. Wang, "Task-adaptive neural process for user cold-start recommendation," *arXiv preprint arXiv:2103.06137*, 2021.

[35] J. Rajendran, A. Irpan, and E. Jang, "Meta-learning requires meta-augmentation," *arXiv preprint arXiv:2007.05549*, 2020.

[36] M. Yin, G. Tucker, M. Zhou, S. Levine, and C. Finn, "Meta-learning without memorization," *arXiv preprint arXiv:1912.03820*, 2019.

[37] M. Federici, A. Dutta, P. Forré, N. Kushman, and Z. Akata, "Learning robust representations via multi-view information bottleneck," 2020.

[38] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *Journal of Machine Learning Research*, vol. 11, no. Dec, pp. 3371–3408, 2010.

[39] L. Zheng, V. Noroozi, and P. S. Yu, "Joint deep modeling of users and items using reviews for recommendation," in *Proceedings of the tenth ACM international conference on web search and data mining*, 2017, pp. 425–434.

[40] D. Liu, J. Li, B. Du, J. Chang, and R. Gao, "Daml: Dual attention mutual learning between ratings and reviews for item recommendation," in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp. 344–352.

[41] L. Hu, J. Cao, G. Xu, L. Cao, Z. Gu, and C. Zhu, "Personalized recommendation via cross-domain triadic factorization," in *22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013*, 2013, pp. 595–606.

[42] F. Yuan, L. Yao, and B. Benatallah, "Darec: Deep domain adaptation for cross-domain recommendation via transferring rating patterns," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, 2019, pp. 4227–4233.

[43] X. Chen, Y. Zhang, I. W. Tsang, Y. Pan, and J. Su, "Towards equivalent transformation of user preferences in cross domain recommendation," *CoRR*, vol. abs/2009.06884, 2020. [Online]. Available: https://arxiv.org/abs/2009.06884

[44] G. Hu, Y. Zhang, and Q. Yang, "Transfer meets hybrid: A synthetic approach for cross-domain collaborative filtering with text," in *The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019*, 2019, pp. 2822–2829.

[45] C. Zhao, C. Li, R. Xiao, H. Deng, and A. Sun, "Catn: Cross-domain recommendation for cold-start users via aspect transfer network," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 229–238.

[46] W. Yu, X. Lin, J. Ge, W. Ou, and Z. Qin, "Semi-supervised collaborative filtering by text-enhanced domain adaptation," in *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, 2020, pp. 2136–2144.

[47] Y. Zhang, B. Cao, and D. Yeung, "Multi-domain collaborative filtering," in *UAI 2010, Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence, Catalina Island, CA, USA, July 8-11, 2010*, 2010, pp. 725–732.

[48] H. Bharadhwaj, "Meta-learning for user cold-start recommendation," in *International Joint Conference on Neural Networks, IJCNN 2019 Budapest, Hungary, July 14-19, 2019*, 2019, pp. 1–8.

[49] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. PMLR, 2017, pp. 1126–1135. [Online]. Available: http://proceedings.mlr.press/v70/finn17a.html

[50] H. Bharadhwaj, "Meta-learning for user cold-start recommendation," in *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–8.

[51] T. Wei, Z. Wu, R. Li, Z. Hu, F. Feng, X. He, Y. Sun, and W. Wang, "Fast adaptation for cold-start collaborative filtering with meta-learning," in *2020 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2020, pp. 661–670.

[52] Y. Lu, Y. Fang, and C. Shi, "Meta-learning on heterogeneous information networks for cold-start recommendation," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 1563–1573.

[53] Y. Wu, C. Macdonald, and I. Ounis, "A hybrid conditional variational autoencoder model for personalised top-n recommendation," in *Proceedings of the 2020 ACM SIGIR on International Conference on Theory of Information Retrieval*, 2020, pp. 89–96.

[54] C. Doersch, "Tutorial on variational autoencoders," *arXiv preprint arXiv:1606.05908*, 2016.

[55] W.-Y. Chen, J.-C. Chu, J. Luan, H. Bai, Y. Wang, and E. Y. Chang, "Collaborative filtering for orkut communities: discovery of user latent behavior," in *Proceedings of the 18th international conference on World wide web*. ACM, 2009, pp. 681–690.

[56] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[57] R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, P. Bachman, A. Trischler, and Y. Bengio, "Learning deep representations by mutual information estimation and maximization," 2019.

[58] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[59] H. Yin, Y. Sun, B. Cui, Z. Hu, and L. Chen, "Lcars: a location-content-aware recommender system," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2013, pp. 221–229.

[60] Y. Zhang, H. Yin, Z. Huang, X. Du, G. Yang, and D. Lian, "Discrete deep learning for fast content-aware recommendation," in *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. ACM, 2018, pp. 717–726.

[61] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "Bpr: Bayesian personalized ranking from implicit feedback," in *Proceedings of UAI'09*. AUAI Press, 2009, pp. 452–461.

[62] H. Lee, J. Im, S. Jang, H. Cho, and S. Chung, "Melu: Meta-learned user preference estimator for cold-start recommendation," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, 2019, pp. 1073–1082.

[63] J. You, Y. Wang, A. Pal, P. Eksombatchai, C. Rosenberg, and J. Leskovec, "Hierarchical temporal convolutional networks for dynamic recommender systems," in *The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019*, L. Liu, R. W. White, A. Mantrach, F. Silvestri, J. J. McAuley, R. Baeza-Yates, and L. Zia, Eds. ACM, 2019, pp. 2236–2246. [Online]. Available: https://doi.org/10.1145/3308558.3313747

[64] R. F. Woolson, "Wilcoxon signed-rank test," *Wiley encyclopedia of clinical trials*, pp. 1–3, 2007.