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 learning, 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 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 $T_1$ is ‘output 0 if the input image is a dog’, and task $T_2$ is ‘output 1 if the image is a dog’, then we call tasks $\{T_1, T_2\}$ are mutually-exclusive.

Generally, we define ‘Mutual Exclusivity’ as: Training samples $(x, y_1), (x, y_2), \ldots, (x, y_k)$ are called mutually-exclusive samples, if all continuous labels $y_1, y_2, \ldots, y_k$ are different from each other with the same input $x$. For example, training samples $(x, 0.1), (x, 0.2), (x, 0.3)$ 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 $(x, y_1), (x, y_2), \ldots, (x, y_k)$ are called diverse samples, if not all continuous labels $y_1, y_2, \ldots, y_k$ 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 $\{y_1, y_2, \ldots, y_k\}$ 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 (D-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 $x_t$. 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 $O(k^2)$ 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 $\beta_2$ 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 $\hat{r}_{t1}, \hat{r}_{t2}, \cdots, \hat{r}_{tk}$ by the learned $k$ encoder-decoder frameworks with the same content $x_t$ 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 $x_s$ 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-C and the impact of hyper-parameters $\beta_1, \beta_2$ the above two constraints in Section V-D. Finally, we conclude this paper in Section VI.
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 recommender systems 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 better domain adaptation.

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 III.

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 $R = \{r_{ui} \geq 0 : u \in U, i \in I\}$, the user content data $C^U$, and the item content data $C^I$. If user $u$ has an interaction with item $i$, then $r_{ui} = 1$; otherwise, $r_{ui} = 0$. For each user $u$, the content data $c_u \in C^U$ is composed of bag-of-vectors generated from her/his rated items’ review data. Similarly, the content data of each item $c_i \in C^I$ 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., \( \{|r_{ui} : u \in U_e^s\} \geq 5 \). 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 items’, and each \( i \in I_e \) denotes the item of receiving no less than 5 ratings, i.e., \( \{|r_{iu} : i \in I_e\} \geq 5 \), 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 \( r_{ui} \notin R_w \) 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 \( R_u \) of new users \( u \in U_n \) to existing items \( i \in I_e \). The goal of ‘C-U’ is to predict unknown ratings \( r_{ui} \notin R_u \) of new users \( U_n \) to existing items \( I_e \) by the finetuned model \( f(\theta) \) and recommend \( k \) new 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 \( r_{ui} \notin R_{ci} \) 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 \( R_{cu} \) of new users \( U_n \) to new items \( I_n \). The goal of ‘C-UI’ is to predict unknown ratings \( r_{ui} \notin R_{cu} \) 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 \( T_u \), which is denoted as the dataset used for the task. To be specific, \( T_u = (c_u, r_u) \), where \( c_u \) is the input of all items for a specific user \( u \) and \( r_u \) 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 use the review comments as the input. In particular, according to the setting of meta-learning, we divide into a support set \( S_u \) and a query set \( Q_u \). Tasks from all users’ preferences are randomly split into two disjoint partitions: one for meta-training tasks denoted as \( T_{tu} \), and another one for meta-testing tasks denoted as \( T_{te} \). 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 \( T \). In contrast, meta-learning considers to learn a set of tasks \( T_u \in T_{te} \), which are sampled from the task distribution \( p(T) \). The objective of meta-training for recommendations is formulated as follows:

\[
\min_{\theta} \sum_{T_u \sim T_{te}} \mathcal{L}_{T_u}(\theta - \alpha \nabla_{\theta} \mathcal{L}_{T_u}(\theta, S_u), Q_u),
\]

where \( \nabla_{\theta} \) denotes the gradient w.r.t parameters \( \theta \) of a preference prediction model, and \( \alpha \) is the meta-learning rate, and \( \theta - \alpha \nabla_{\theta} \mathcal{L}_{T_u}(\theta, S_u) \) is the task-specific parameters adapted to the task \( T_u \) after one gradient step from the global \( \theta \). The algorithm locally updates parameters \( \theta \) based on the gradient with \( S_u \), and then globally updates \( \theta \) based on \( Q_u \), so that the globally updated parameters adapt to various tasks \([4]\). During meta-testing, the meta-learner adapts the learned \( \theta \) to \( T_u \in T_{te} \) with its support set \( S_u \), and then the adapted parameters \( \theta \) is used to predict ratings and evaluate the recommendation performance in its query set \( Q_u \)[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 CV AEs, 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 \([45]\), it takes the review texts as domain-invariant features to align the latent space for domain adaptation. To facilitate the domain adaptation, we also adopt the domain-invariant reviews as the content data. Specifically, the user content \( c_u \) 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 \( x_u \). We denote \( x_u^{(s)} \) and \( x_u^{(t)} \)
Multi-source Domain Adaptation (shared users) | Diverse Preference Augmentation | Preference Meta-learning (all users)
---|---|---
Content | Ratings | Books
(Movies, CDs, Music) | \(r_s, x_s\) | \(r_t, x_t\)
\(p(x_t, r_t)\) | \(q(z_t|x_t)\) | \(q(z_t|x_t)\)
\(\{\mathcal{S}_{u}, \mathcal{T}_u\}\) | \(\{\mathcal{S}_{u}, \mathcal{T}_u\}\) | \(\{\mathcal{S}_{u}, \mathcal{T}_u\}\)

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.

We employ the softmax function as the activation function in the variational posterior and the prior. So the loss function in the 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 \(x_s\) and \(x_t\), 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:

\[
\mathcal{L}_{KL} = - \frac{1}{2} \sum_{l=1}^{L} [\sigma_{z_l}^2 + (\mu_{z_l} - \mu_{x_l}^z)^2 - \log \sigma_{z_l}^2 - 1] + \frac{1}{2} \sum_{l=1}^{L} [\sigma_{t_l}^2 + (\mu_{t_l} - \mu_{x_l}^t)^2 - \log \sigma_{t_l}^2 - 1].
\]

where \(L\) is the dimension of the latent sampled representations. For the source domain, \(u_s\) and \(\sigma_s\) are the mean and variance of the approximate posterior, and \(z_s^t\) is the output of a dense embedding encoder \(E_s^z\), parameterized by \(\phi_{z_s}\). The notations \(u_t, \sigma_t, z_t^t, E_t^z\), and \(\phi_{z_t}\) in the target domain have similar meanings.

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

\[
\mathcal{L}_{MSE} = ||z_s - q_{\phi_{z_s}}(z_s^t|x_s)||^2 + ||z_t - q_{\phi_{z_t}}(z_s^t|x_t)||^2.
\]

as the user content vectors in the source domain and in the target domain, respectively. For simplicity, we use \(x_s\) and \(x_t\) to represent them. Similarly, we denote \(r_s\) and \(r_t\) 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 \(r_s\) and \(r_t\). In the Dual-CVAE, we add conditions \(x_s, x_t\) on the latent representations in a similar way with HCVAE [53]. Firstly, the input \(x_s\) and \(x_t\) are encoded into a distribution \(q_{\phi_s}(z_s|x_s)\) and \(q_{\phi_t}(z_t|x_t)\), respectively. The encoders of the Dual-CVAE maps ratings \(r_s\) and \(r_t\) to latent representations \(z_s\) and \(z_t\), respectively. The optimization objective of the Dual-CVAE is the evidence lower bound (ELBO) [54], which consists of the sum of the reconstruction error, namely, maximizing the likelihood estimation of the decoders \(\log p_{\theta_s}(r_s|z_s, x_s)\) and \(\log p_{\theta_t}(r_t|z_t, x_t)\), 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]:

\[
\mathcal{L}_{ELBO} = \mathcal{L}(r_s, x_s; \theta_s, \phi_s) + \mathcal{L}(r_t, x_t; \theta_t, \phi_t)
= \mathbb{E}_{q_{\phi_s}(z_s|x_s)}[\log p_{\theta_s}(r_s|z_s, x_s)] + \mathbb{E}_{q_{\phi_t}(z_t|x_t)}[\log p_{\theta_t}(r_t|z_t, x_t)]
- D_{KL}[q_{\phi_s}(z_s|x_s, x_s)||p(z_s)]
- D_{KL}[q_{\phi_t}(z_t|x_t, x_t)||p(z_t)].
\]

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.
we force the generated rating from the decoder to the reconstructed ratings as possible from the decoder
\[ \hat{r}_{si} = \frac{1}{2m} \sum_{i=1}^{m} [r_{si} \log(\hat{r}_{si}) + (1 - r_{si}) \log(1 - \hat{r}_{si})] \]

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 \( \hat{r}_s \) and \( \hat{r}_t \) of a shared user \( u \) as follows:
\[ \mathcal{L}_{\text{ME}} = -I(\hat{r}_s, \hat{r}_t), \]

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 \( \{9, 7\} \) 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,
\[ \mathcal{L}_{\text{Dual-CVAE}} = \mathcal{L}_{\text{ELBO}} + \mathcal{L}_{\text{MSE}} + \mathcal{L}_{\text{Rec}} + \beta_1 \mathcal{L}_{\text{MDI}} + \beta_2 \mathcal{L}_{\text{ME}}. \]

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 \( E_k^c \) and \( k \) decoders \( D_t \), and then employ them to generate \( k \) diverse ratings \( r_{t1}, r_{t2}, \ldots, r_{tk} \) with the content \( x_t \). 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 4. In Section 4 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 \( r_{t1}, r_{t2}, \ldots, r_{tk} \). Firstly, we encode the original content \( c_u \) and \( c_t \) to their dense embedding \( x_u \) and \( x_t \). We denote \( c_u \) as the combination of \( c_u \) and \( c_t \) in the following paper. Then, we construct the meta-learning task \( T_u \) and the augmented tasks \( T_{u1}, \ldots, T_{uk} \) as follows:
\[ T_u = (c_t, r_t) \]
\[ T_{u1} = (c_t, \hat{r}_{t1}), \ldots, T_{uk} = (c_t, \hat{r}_{tk}) \]

where \( r_t \) 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 \( c_u \) and \( c_t \) into dense embeddings \( x_u \) and \( x_t \), and then we adopt a multi-layer neural architecture [29] to predict rating scores by the concatenation of \( x_u \) and \( x_t \). 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
with diverse tasks to avoid overfitting via the meta-learning framework MAML. The objective can be written as:

\[ \hat{r}_{ui} = f(\theta_1, \theta_c, c_u, c_i), \]

where \( \theta_c \) is the parameters of the fully connected embedding layer that encodes the content vector \( c_u \) and \( c_i \) into dense embeddings \( x_u \) and \( x_i \), and \( \theta_l \) 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 \( \mathcal{T}_{tr} \), which includes the original tasks \( \mathcal{T}_u \) and augmented tasks \( \mathcal{T}_{u1}, \ldots, \mathcal{T}_{uk} \). We divide samples of each task \( \mathcal{T} \in \mathcal{T}_{tr} \) randomly into a support set \( \mathcal{S}_u \) and a query set \( \mathcal{Q}_u \) as follows:

\[ \mathcal{T} = \{\mathcal{S}_u, \mathcal{Q}_u\}. \]

In the meta-testing phase, meta-testing tasks \( \mathcal{T} \in \mathcal{T}_{te} \) 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 \( \mathcal{Q}_u \) 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 \( z_u, z_i \) 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 \( O\{\max\{n, m\}\} \), 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 \( r_u \) and \( r_i \) are \( l \) and \( m \), respectively. So the time complexity is \( O(B(l + m)) \). (2) The second block forwards the encoder and decoder (red line in Fig. 1) one-pass to generate ratings, so the time complexity is \( O(B) \). (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 \( O(B) \). Overall, the time complexity is \( O(B(l + m)) \), 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 \( \beta_1 \) of MDI, and \( \beta_2 \) 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 dataset\(^1\). 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 \( \mathcal{R}_u \). 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 fine-tune 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\(^{http://jmcauley.ucsd.edu/data/amazon/}\)
Taking ‘C-U’ as an example, \( T_{te} \) is constructed from ratings \( \{ r_{ui} : u \in U_n, i \in I_e \} \). Similarly, we can construct testing tasks \( T_{te} \) 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 feeds 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 \( \beta_1 \) and \( \beta_2 \) in Eq.8, we apply the grid search in the range of \{1e\(^{-2}\), 1e\(^{-1}\), 1, 1e\(^1\), 1e\(^2\}\) \( \square \{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 5A.3. We conclude that the performance of the proposed MetaDPA is significantly superior to all competing baselines. The over-
wholesome 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
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
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 $E_C$ and the decoder $D_1$. 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 hyperparameters $\beta_1$ and $\beta_2$ on the recommendation performance.

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 $E_C$ and the decoder $D_1$. 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 hyperparameters $\beta_1$ and $\beta_2$ on the recommendation performance.

ACKNOWLEDGMENTS

This work is supported by the Major Project for New Generation of AI under Grant No. 2018AAA0100400, the National Natural Science Foundation of China (Grant No. 62002052, 61801060, 62176047, 62172075, 61832001), the Sichuan Science and Technology Program (Grant No. 2022YFG0189, 2021JDRC0079). Besides, this work is supported in part by Australian Research Council (Grant No. FT210100624, DP180100106, DP200101328, DP190101985), the Beijing Natural Science Foundation (Grant No. Z190023), the China Postdoctoral Science Foundation (Grant No. 2019TQ0051), the Open Fund of Intelligent Terminal Key Laboratory of Sichuan Province (Grant No. SCITLAB-1017).
