MetaCAR: Cross-Domain Meta-Augmentation for Content-Aware Recommendation

Hui Xu, Changyu Li, Yan Zhang, Lixin Duan, Ivor W. Tsang and Jie Shao

Abstract—Cold-start has become critical for recommendations, especially for sparse user-item interactions. Recent approaches based on meta-learning succeed in alleviating the issue, owing to the fact that these methods have strong generalization, so they can fast adapt to new tasks under cold-start settings. However, these meta-learning-based recommendation models learned with single and spase ratings are easily falling into the meta-overfitting, since the one and only rating r_{ui} to a specific item i cannot reflect a user's diverse interests under various circumstances(e.g., time, mood, age, etc), i.e. if r_{ui} equals to 1 in the historical dataset, but r_{ui} could be 0 in some circumstance. In meta-learning, tasks with these single ratings are called Non-Mutually-Exclusive(Non-ME) tasks, and tasks with diverse ratings are called Mutually-Exclusive(ME) tasks. Fortunately, a meta-augmentation technique is proposed to relief the meta-overfitting for meta-learning methods by transferring Non-ME tasks into ME tasks by adding noises to labels without changing inputs. Motivated by the meta-augmentation method, in this paper, we propose a cross-domain meta-augmentation technique for content-aware recommendation systems (MetaCAR) to construct ME tasks in the recommendation scenario. Our proposed method consists of two stages: meta-augmentation and meta-learning. In the meta-augmentation stage, we first conduct domain adaptation by a dual conditional variational autoencoder (CVAE) with a multi-view information bottleneck constraint, and then apply the learned CVAE to generate ratings for users in the target domain. In the meta-learning stage, we introduce both the true and generated ratings to construct ME tasks that enables the meta-learning recommendations to avoid meta-overfitting. Experiments evaluated in real-world datasets show the significant superiority of MetaCAR for coping with the cold-start user issue over competing baselines including cross-domain, content-aware, and meta-learning-based recommendations.

Index Terms—Recommendation systems, meta-augmentation, cold-start, content-aware.

1 Introduction

Personalized recommendations have been recognized as one of the most critical and effective approaches for alleviating information overload. They are key decision support systems in various applications such as e-commerce websites (Amazon, Netflix, Yelp, etc.), online educational systems and online news systems. Existing recommendation models, e.g., NeuMF [1], are mainly based on users' previous behavior interactions, such as purchase records, ratings, click actions, and watch records. However, the interaction matrix in real-world applications is generally very sparse. Most users and items have only a few or even no interactions. As a consequence, the recommendation model cannot effectively learn effective users' presentations from these limited interactions and this leads to poor performance. Existing recommendation models addressing cold-start issues can generally be categorized into three subclasses: content-aware recommendations [2], [3], [4], [5], cross-domain recommendations [6], [7], and meta-learning-based recommendations [8], [9], [10].

Content-aware recommendations [11], [12], [13], [14], [15] integrate user-item interaction data and user/item content data, including context, figures and knowledge graphs, to learn effective representations of users or items. These content-aware recommendation systems can handle sparsity and cold-start issues by effectively extracting representations from the content data. However, the performance improvement is constrained by sparse interactions, and the limitation of available content data, e.g., user profile data is often unavailable.

Cross-domain recommendations [16], [17], [18] explore across source and target domains by using interaction data or content data from the source domain to enhance the representations in the target domain to mitigate the data sparsity and cold-start issues. These methods learn prior knowledge, which is close to true data distributions, by utilizing domain-shared and domain-specific properties of source and target domains. The closer the prior knowledge is to the true data distribution, the more meaningful it is, and its performance in real-world applications is better. Cross-domain recommendation models are based on the shared property between two domains. Therefore, these recommendations can easily overfit the shared properties and lead to poor generalization ability.

Meta-learning is an emerging machine learning discipline having fast adaption capabilities to new concepts and can obtain optimal convergence with limited training samples. Due to its strong generalization ability, one

- *H. Xu, C. Li, L. Duan and J. Shao are with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China. H. Xu and J. Shao are also with Sichuan Artificial Intelligence Research Institute, Yibin 644000, China. Email: {huxu.kim,lxduan,shaojie}@uestc.edu.cn, changyulve@std.uestc.edu.cn.*
- *Y. Zhang and L. Duan are with the Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China, Shenzhen 518028, China. Y. Zhang is also with Intelligent Terminal Key Laboratory of Sichuan Province, Yibin 644000, China. L. Duan is also with the Sichuan Provincial People's Hospital, University of Electronic Science and Technology of China, Chengdu 610072, China. Email: yixianqianzy@gmail.com.*
- *I. W. Tsang is with Center for Frontier AI Research, Research Agency for Science, Technology and Research (A*STAR), Singapore, and also with the Australian Artificial Intelligence Institute, University of Technology Sydney, NSW 2007, Australia. E-mail: Ivor.Tsang@gmail.com.*
- *Corresponding authors: Lixin Duan, Yan Zhang.*

representative meta-learning framework MAML [19] has been introduced to recommendation systems to address the data sparsity and cold-start issues [20], [21], [22]. Usually, meta-learning-based recommendations [23] treat users' preferences over items as meta-learning tasks. In such a setting, they learn a good initialization for the model with meta-training tasks (constructed by ratings of active users), and then fast adapt to meta-testing tasks with only a few ratings from cold-start user. However, existing meta-learning-based recommendation models [20], [21], [22] suffer meta-overfitting caused by the sparse and Non-Mutually-Exclusive (Non-ME) tasks constructed with the one and only ratings of users, which leads to poor performance for cold-start recommendations. Specifically, in meta-learning-based recommender systems, a user u with different rantings $r_u^1, r_u^2, r_u^3 \cdots$ can construct mutually-exclusive (ME) tasks $\{\mathcal{T}_1 = (x_u, r_u^1), \mathcal{T}_2 = (x_u, r_u^2), \mathcal{T}_3 = (x_u, r_u^3), \cdots\}$, where x_u is the input, and a user u with the one and only ratings, e.g., r_u^1, can only construct Non-ME task $\{\mathcal{T}_1 = (x_u, r_u^1)\}$.

Fig. 1. A motivating example. Taking Movies dataset on Amazon as an example, suppose we have n users for training a recommender system. Then we can construct n training tasks based on their ratings. Tasks $A_1 \cdots A_n$ denote the observed interest of users $u_1 \cdots u_n$ (denoted as'Non-ME' tasks). Tasks $I_1 \cdots I_n$ denote the potential(invisible) interests, which is not available in the historical dataset. The above two groups of tasks constitute a new training set, also named 'ME' tasks. A test user who is probably interested in Comedy and Horror movies. The upper flow chart of illustrates the *meta-overfitting* problem suffered in the recommender systems trained on Non-ME tasks . The lower flow chart shows how to avoid meta-overfitting by *meta-augmentation* in a recommender system. The recommender system trained on 'ME' tasks is supposed to recommend what the test user really like.

In practical, the one and only rating of a user to an item cannot reflect the user's dynamic interests in this item. But a user's preference (rating) over an item varies indeed with different circumstances(e.g., time, mood, age, etc). In meta-learning-based recommender systems, tasks constructed with single ratings are called Non-ME tasks. Figure 1 explains the meta-overfitting problem caused by the sparse and Non-ME tasks intuitively on Movies dataset. In this figure, the white box displays a training user's preferable movies so that ratings on these movies are 1, and ratings on other movies are 0 in the historical dataset. Tasks with available ratings are Non-ME tasks. As listed in the figure, if all preferable movies in the training dataset are Adventure movies, then the trained recommendation model will recommend a test user with Adventure movie, even though the test user prefers Horror and Comedy movies. This phenomena is meta-overfitting in meta-learning-based recommender systems, which caused by sparse and Non-ME tasks. In fact, the meta-overfitting in meta-learning includes memorization overfitting and meta-learner overfitting [24]. The former is caused by Non-ME tasks, and the latter is induced by sparse tasks.

To avoid meta-overfitting, the meta-augmentation technique [24] for meta-learning generates different labels by adding noises to the original ones without changing the inputs, and then constructs Mutually-Exclusive(ME) tasks based on both generated and original data, which transfers the original Non-ME tasks to ME tasks by adding generated data into the training dataset. Similarly, in the recommender system shown in Figure1, it is expected to avoid meta-overfitting by adding new potential ratings (invisible) to these items, i.e., adding the user's possible preferable movies, Halloween, Thirteen Ghost, Big world, Big Adventures, etc. With both available and invisible ratings, we can construct ME tasks. Then, the recommender system trained on ME tasks are supposed to avoid the meta-overfitting and provide recommendations for test users.

To address these two forms of meta-overfitting, we propose a cross-domain meta-augmentation for content-aware recommendation (MetaCAR) by generating mutually exclusive tasks. MetaCAR consists of two stages: meta-augmentation and meta-learning. In the meta-augmentation stage, we first adopt a cross-domain component to learn prior knowledge p from the shared users between source and target domains. Then, we transfer the prior to the target domain to generate meaningful plausible ratings by utilizing the content information of the existing users from the target domain. ME tasks therefore can be constructed from the plausible and true ratings. In contrast, other data augmentation techniques for recommendation [25], [26] require that the plausible ratings be very close to the true data distribution, which is uncertain in the real world applications. Furthermore, meta-augmentation cares about both the difference and closeness between plausible ratings and true ratings. Such a constraint relaxes the difficulty of data augmentation. In addition, meta-augmentation generates new ratings for the same user, so we may assign different ratings for the existing (already observed) item in the dataset. Unlike this, other methods generate new ratings for new items that do not exist in the dataset.

Our main contributions are summarized as follows:

- To address the meta-overfitting problem, we propose simple and easy to implement MetaCAR to generate ME tasks.
- To learn prior knowledge that satisfies both closeness and distinguishment requirement of ME tasks, we build a dual CVAE model and impose a multi-view information bottleneck (MIB) [27] to learn domain-shared properties and discard domain-specific properties.
- We develop a novel technique named meta-augmentation with prior knowledge to generate ME tasks by utilizing the same user content. To our knowledge, this is the first study to augment the data by generating plausible ratings for the existing

user-item pair.

- We evaluate MetaCAR on public datasets to demonstrate its superiority for the cold-start user problem over different leading baselines.

The rest of this paper is organized as follows. We first introduce the related work in Section 2. We then present the preliminaries, including problem formulation and meta-learning-based recommendation model in Section 3. Next, the proposed MetaCAR is described in Section 4. Experimental evaluations are reported in Section 5. Finally, we draw our conclusions in Section 6.

2 Related Work

2.1 Content-aware Recommendation Systems

Content-aware recommendation systems learn the representation of items and users based on content data, e.g., user profile, behaviors, and item descriptions. Collaborative topic regression (CTR) [3] concatenates the benefits of collaborative filtering and probabilistic matrix factorization with a topic model. It gains representations from a large collection of articles through an interpretable latent structure. Such representations can be used to deal with existing and newly published articles. Collaborative deep learning (CDL) [4] is a deep learning-based content-aware recommendation system. CDL is implemented with a deep probabilistic model that jointly learns the content data and the preference rating matrices. Unlike CTR, CDL can handle very sparse auxiliary information due to the effective deep hierarchical Bayesian model. Multi-view group representation learning (MGPL) [28] address the cold-start issue in group recommendation by learning the group preferences from multiple views and incorporating the different types of content. Context-aware diversity-oriented knowledge recommendation (CDKR) [29] is proposed to address the in-context accuracy and diversity issue by fully considering item context, user profiles, and problem-solving context.

Content is considered to provide additional information that can describe users' and items' properties. Different from these methods, our approach not only considers content information but also takes into account the gap between content and ratings. This can lead to a distinguishable prior knowledge for creating mutually exclusiveness.

2.2 Cross-domain Recommendation Systems

Cross-domain recommendation methods facilitate the personalization process by exploiting specific domain knowledge into its relevant domains. As a pioneering work, collective matrix factorization (CMF) [6] achieves knowledge integration across domains by factoring several rating matrices and sharing parameters among users factors in multiple domains. Subsequently, multi-domain collaborative filtering (MCF) [7] was proposed by modeling the rating patterns in different domains simultaneously and exploiting link function for different domains for adaptive knowledge transfer across multiple domains. Cross-domain triadic factorization (CDTF) [16] is another milestone that captures the triadic factors of users, items, and domains by tensor factorization with explicit and implicit feedbacks.

Recently, many cross-domain recommendations based on deep learning have been developed to strengthen the representations of target domains. Without relying on any auxiliary information, the deep domain adaptation model (DARec) [17] extracts and transfers only rating patterns by utilizing a domain classifier that shares rating patterns of the same user in different domains. Modeling user behaviors across different domains as a joint distribution, the equivalent transformation learner (ETL) [18] assumes that the shared user's preferences in different domains can be expressed by each other. In particular, ETL converts user's preferences from one domain to another by utilizing an equivalent transformation. However, these methods still suffer severely from data sparsity and cold-start issues when integrating different domains. In our method, we adopt a similar assumption by considering the preferences of a shared user in two different domains as two views with the same unknown label.

For better domain adaptation, content-aware cross-domain models [30], [31], [32] consider both ratings and content information from auxiliary domains to enhance representation. Text-enhanced domain adaptation recommendation (TDAR) [33] adopts the idea of projecting high-dimensional data to subspace [34] by extracting the textual features in word semantic space for each user and item. Then, the textual features are fed into a collaborative filtering model for prediction. TDAR provides an effective textual feature extractor named text memory network (TMN), which is used in our method. In addition, RecSys-DAN [35] learns domain indistinguishable representations by optimizing an adversarial loss.

Compared with these cross-domain methods which transfer data distribution by learning a domain adaptation model, MetaCAR transfers data patterns through the generated ratings. In addition, our main goal is also different. While these methods focus on better domain adaptation, our method focuses on the trade-off between closeness (better domain adaptation) and distinguishable from the true data distribution of target domain. Thus, MetaCAR pays more attention to the domain-shared properties and discards domain-specific properties. This is helpful to construct mutually exclusive tasks, while other methods aim to learn both domain-shared properties and domain-specific properties.

2.3 Meta-Learning-based Recommendation Systems

Meta-learning [36] learns from the training tasks (user's preference) and tests or unseen tasks (new users) with only a few samples (items) to fine-tune the model. In this context, the recommendation generations can be seen as a meta-learning problem. Naturally, meta-learning shows its great potential to solve the data sparsity and cold-start issues in recommendation systems [20]. Existing meta-learning-based recommendation models employ the well-known MAML [19] to address the cold-start issues. MAML is an optimization-based meta-learning method with strong generalization ability, which can be easily deployed without any extra requirement. Due to the strong generalization ability and fast convergence capabilities, the MAML-based recommendation techniques [23], [37] demonstrate satisfactory performance when applied to extremely sparse user-item interactions. Additionally, considering the growing

TABLE 1
Notations.

Symbol	Description
r_s	ratings (implicit feedback) in the source domain
r_t	ratings (implicit feedback) in the target domain
$\mathbf{x}_s$	concatenations of user and item embeddings in the source
$\mathbf{x}_t$	concatenations of user and item embeddings in the target
$\mathbf{z}_s$	latent representations in the source
$\mathbf{z}_t$	latent representations in the target
D_{train}	meta-training task set
D_{test}	meta-testing task set
q_{ϕ_s}	the distribution of latent representations in the source
q_{ϕ_t}	the distribution of latent representations in the target
μ_s, Σ_s	the mean and variance of latent representations $\mathbf{z}_s$
μ_t, Σ_t	the mean and variance of latent representations $\mathbf{z}_t$

need for customized preferences of an individual user in practical recommendations, a MAML-based optimal strategy introduced in MeLU [21] directly considers individual users' item consumption history and performs well with only a limited number of training samples. MeLU shows the powerful performance of meta-learning on three kinds of cold-start issues: new users, new items, and new items for new users. In this paper, we adopt the same setting by treating the user's preference as a task.

Meta-learning is an effective tool to deal with cold-start. Existing meta-learning-based recommendation models directly construct Non-ME tasks from the historical data. Consequently, they lead to poor performance in some cold-start scenarios (see MeLU in Section 5, and shows poor performance in some scenarios due to the serious meta-overfitting). In contrast, MetaCAR carefully addresses the meta-overfitting problem by utilizing the meta-augmentation process to construct mutually exclusive training tasks from historical and meta-augmentation data.

3 Preliminaries

3.1 Meta-Learning for Recommendations

Meta-learning, also named learning to learn, aims to learn good initial weights for a model that can quickly adapt to unseen tasks with a few samples [38]. In meta-learning-based recommendation models [21], [23], user's preferences over items are denoted as a task $\mathcal{T}$ that is divided as a support set $\mathcal{S}$ and a query set $\mathcal{Q}$. Tasks of all users are divided into two parts: one for meta-training denoted as the set D_{train} and the remaining for meta-testing denoted as the set D_{test}. Meta-learning considers optimizing a model with a series of tasks $\mathcal{T}' \in D_{train}$, which are sampled from the task distribution $p(\mathcal{T})$. In this paper, we consider the well-known optimization-based meta-learning method MAML [19] to optimize our recommendation system. The model parameters can be customized via a standard fine-tuning process for each user. The meta-objective function for meta-training phase can generally be formulated as follows:

$$\min_{\theta} \sum_{\mathcal{T}' \sim D_{train}} \mathcal{L}_{\mathcal{T}'}(\theta - \alpha \nabla_\theta \mathcal{L}_{\mathcal{T}'}(\theta, \mathcal{S}), \mathcal{Q}), \tag{1}$$

where ∇_θ denotes the gradient w.r.t. parameters θ of a model and α is the step size.

In the meta-training phase, the task-specific parameters are updated by one gradient step from the global θ as $(\theta - \alpha \nabla_\theta \mathcal{L}_{\mathcal{T}'}(\theta, \mathcal{S})$ in the inner loop, and the gradient is computed with $\mathcal{S}$. The outer loop used to update θ is computed with $\mathcal{Q}$ through task-specific parameters. Finally, the model θ is updated by the gradients of the outer loop from tasks sampled from D_{train}, so that the updated parameters θ are adapted to various tasks.

In the meta-testing phase, we firstly fine-tune the learned model θ with support sets of tasks in D_{test}; we then test the recommendation performance on query sets in D_{test}.

3.2 Problem Formulation

Let $U = \{1, 2, \cdots, n\}$ denote the user index set and $I = \{1, 2, \cdots, m\}$ denote the item index set. $\mathbf{R} = (r_{ui})_{n \times m}$ denotes the observed user-item interaction matrix. The interactions could be explicit ratings (e.g., ratings 1 to 5) or implicit feedback (i.e., binary observations such as 0 or 1). As implicit feedback is more common in real applications [39], we design our model with implicit feedback, where $r_{ui} = 1$ denotes that user u has an interaction (such as click, save, rating action, etc.) with an item i, while $r_{ui} = 0$ denotes that user u has no interactions with item i. Other essential notations in the paper are listed in Table 1.

Cold-start User Problem: Given the rating dataset $\mathbf{R}_s = \{r_{ui}^{(s)}\}_{n_s \times m_s}$ of n_s users over m_s items in a source domain, the sparse rating dataset $\mathbf{R}_t = \{r_{ui}^{(t)}\}_{n_t \times m_t}$ of n_t users over m_t items in a target domain, the content data $\mathbf{X}_s^U(\mathbf{X}_t^U)$ and $\mathbf{X}_s^I(\mathbf{X}_t^I)$ of users and items in the source(target) domain. We divide users(items) into active users(items) and cold-start users(items). Active users(items) refer to the users(items) with no less than k ratings. Instead, cold-start users(items) denote the users(items) with less than k ratings. Our goal is to improve the recommendation performance under the cold-start user setting in the target domain.

Meta-overfitting Problem: In the target domain, suppose the training set is denoted as D_{train}. Each task of D_{train} is denoted as $\mathcal{T} = (\mathbf{x}_u, \mathbf{r}_u)$, where $\mathbf{x}_u(\mathbf{r}_u)$ denotes the content(rating) data of the user u. Since historical user-item ratings collected are the one and only, these single ratings cannot reflect users' diverse interests under different circumstances(e.g., time, mood, age, etc.), which leads to meta-overfitting in meta-learning-based recommendations. In meta-learning, tasks of D_{train} with these single ratings are considered to be Non-Mutually-Exclusive (Non-ME) (i.e., a one-to-one correspondence between the inputs ($\mathbf{x}_u$) and the labels ($\mathbf{r}_u$)), which inevitably leads to meta-overfitting [24], [40]. A feasible solution validated in [24] is to transfer Non-ME into ME tasks by meta-augmentation, that generates new data by adding noises to labels without changing inputs, and then introduce these generated data into constructing a new training set of one-to-many correspondence between the inputs and labels. Similary, our goal is to avoid the meta-overfitting issue in recommender system by transferring Non-ME tasks in D_{train} into ME tasks via meta-augmentation based on the source domain, i.e., constructing a new training set of one-to-many correspondence between $\mathbf{x}_u$ and $\mathbf{r}_u$.

4 Cross-Domain Meta-Augmentation for Content-Aware Recommendations

MetaCAR is proposed to address the cold-start user issue by meta-augmentation with prior technique using an auxiliary

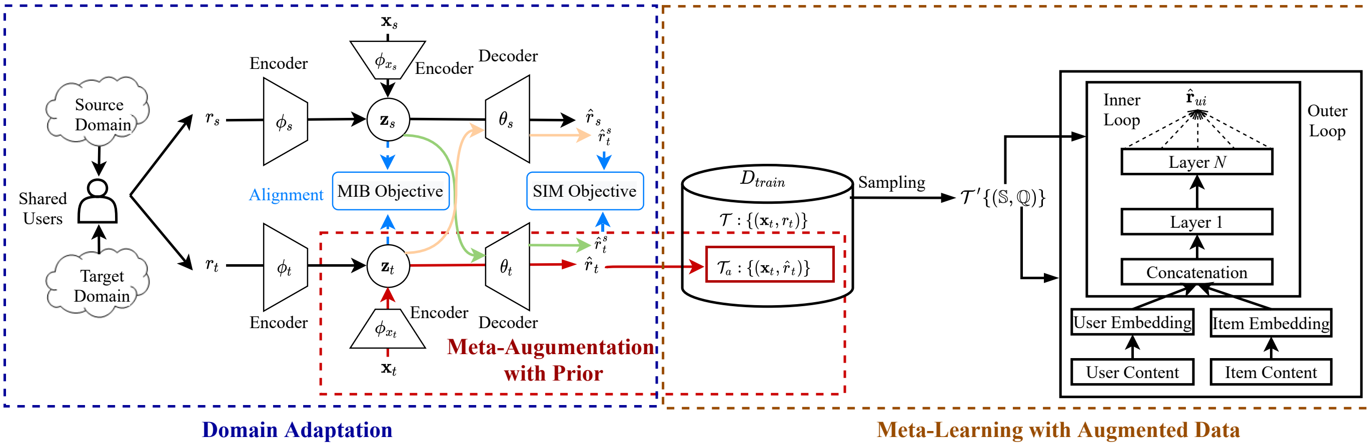


Fig. 2. The MetaCAR framework. We first train the domain adaptation component, implemented by a dual CVAE, with the preference data of shared users between the source and target domains. Then, we generate ratings for users in the target domain with the condition content $\mathbf{x}_t$ by the decoder of CVAE (highlighted with red lines) to augment tasks. This process is denoted as meta-augmentation with prior. Note that we use only one tuple (x, r) to denote the data of a task for simplicity. Finally, we sample a task $\mathcal{T}$ from the training dataset D_{train}, which includes the true and augmented tasks. Each task can be divided into support set $\mathcal{S}$ and query set $\mathcal{Q}$ for a specific user u_i. Then, we introduce them into the MAML component for training the rating prediction model.

source domain and content information. The overall framework is shown in Figure 2. It includes two stages: meta-augmentation and meta-learning. Our goal is to generate ME tasks in the meta-augmentation stage that might lead different users' preferences from the original user. Specifically, firstly, it utilizes the domain adaptation component to transfer users' preferences from the source domain to the target domain by a dual CVAE model. Then, it augments ratings by the decoder from the learned CVAE model for the target domain and constructs mutually exclusive tasks with these plausible ratings. We name this process meta-augmentation with prior. Algorithm 1 shows the detailed procedure of the above discussed. Finally, meta-optimization framework MAML [19] can effectively learn these ME tasks and alleviate meta-overfitting. Conceptually, these settings have high potential to address the data sparsity and cold-start issues.

4.1 Domain Adaptation

In this section, we introduce our domain adaptation component. Its main goal is to transfer prior knowledge that is meaningful but distinguishable from the true data distribution in the target domain. The domain-shared properties sustain knowledge from both source and target domains. Current cross-domain methods consider both domain-shared and domain-specific properties to learn a prior closer to the true data distribution. To trade-off between closeness for meaningful prior and distinguishment for ME, MetaCAR adopts a simple but effective strategy by considering the domain-shared properties and discarding the domain-specific properties. Such a strategy mitigates the difficulty of domain adaptation and suffices for creating ME tasks. In order to learn domain-shared properties only, we build our model as a dual CVAE with the content conditioned on latent representations and impose the multi-view information bottleneck (MIB) constraint to align the two distributions of latent representations for source and target domains. This simple but effective component enables dual knowledge transfer across domains and lets them benefit from each other.

Algorithm 1 MetaCAR algorithm in meta-augmentation stage

Require: D_s and D_t: shared user preference data from source domain and target domain.
Require: D_{train} : meta-training dataset in the target domain.
Random initialize CVAE framework, including ϕ_s, ϕ_{x_s}, θ_s, ϕ_t, ϕ_{x_t}, θ_t.
while not done **do**
  for all shared user u **do**
    Sample data (x_s, r_s) and (x_t, r_t), corresponding to user u, from D_s and D_t.
    Fix ϕ_{x_s} and ϕ_t, and update ϕ_s, θ_s, ϕ_t, θ_t via ELOB loss (Eq. (2) and Eq. (3)), MIB loss (Eq. (5)) and similarity loss (Eq. (8)) with (x_s, r_s) and (x_t, r_t).
    Obtain z_s and z_t from ϕ_s and ϕ_s, where r_s and r_t are the input.
    Fix ϕ_s, θ_s, ϕ_{x_t}, θ_t, and update ϕ_{x_s} and ϕ_t via MSE loss (Eq. (4)) with (x_s, z_s) and (x_t, z_t).
  end for
end while
for all user u in D_{train} **do**
  Obtain the condition terms c_t via ϕ_{x_t}, where x_t is the input.
  Obtain $\hat{r}_t$ via θ_t, where c_t is the input.
  Augment new task $\mathcal{T}_a$: $\{(x_t, \hat{r}_t)\}$, and put it into D_{train}.
end for
return D_{train}

The content information is used to enrich the representations of users/items and transfer the domain-shared properties with meta-augmentation. In addition, the gap between the content and ratings can also help to learn a distinguishable prior knowledge for creating MEness. The content information of users/items is often used to alleviate cold-start issues, but it cannot be shared due to certain model and data dependencies. To align the representations, we explore the domain-invariant reviews [41] as the content data. We first encode the user's content $\mathbf{c}_u^s$ and the item's content $\mathbf{c}_i^s$ into the same dense low-dimensional embeddings and concatenate them together as $\mathbf{x}_{ui}$. We denote $\mathbf{x}_{ui}^{(s)} \in \mathbf{X}_{ui}^s$ and $\mathbf{x}_{ui}^{(t)} \in \mathbf{X}_{ui}^{(t)}$ as the user-item concatenation content in the source domain and the target domain, respectively. For simplicity, we use $\mathbf{x}_s$ and $\mathbf{x}_t$ to represent them. We denote r_s and r_t as ratings rated by the shared user u to the item i.

CVAE is an extension of variational autoencoder, and it can generate some specific data with the data generation process of the condition on encoder and decoder [42]. In this work, we use a dual CVAE network to learn users' representations and reconstruct the input ratings r_s and r_t, and add conditions $\mathbf{x}_s$ and $\mathbf{x}_t$ on the learned distributions of latent representations, as highlighted in the blue dashed part of Figure 2. We use the conditions $\mathbf{x}_s$ and $\mathbf{x}_t$ to control the data generation process in the dual CVAE model to generate some specific data with domain-shared knowledge. Specifically, we train the dense embedding encoders ϕ_{x_s} and ϕ_{x_t} of the conditions $\mathbf{x}_s$ and $\mathbf{x}_t$ with shared users of source and target domains. Then, user's plausible preference can be generated through the encoders ϕ_{x_s} and ϕ_{x_t} with domain-shared knowledge.

In the dual CVAE, the encoder transforms the preference of each user into low-dimensional latent representations z. The input ratings of each user r_s and r_t are encoded into distributions $q_{\phi_s}(\mathbf{z}_s|r_s, \mathbf{c}_s)$ and $q_{\phi_t}(\mathbf{z}_t|r_t, \mathbf{c}_t)$ of latent representations $\mathbf{z}_s$ and $\mathbf{z}_t$, respectively, where the condition terms $\mathbf{c}_s$ and $\mathbf{c}_t$ are the output of a dense embedding encoder parameterized by ϕ_{x_s} and ϕ_{x_t} with corresponding input $\mathbf{x}_s$ and $\mathbf{x}_t$, respectively. The distribution of the latent representations $\mathbf{z}_s$ and $\mathbf{z}_t$ is chosen to be two Gaussian distributions $\mathcal{N}(\mu_s, \Sigma_s)$ and $\mathcal{N}(\mu_t, \Sigma_t)$, respectively.

The decoders reconstruct ratings $\hat{r}_s$ and $\hat{r}_t$ with probability distributions $p_{\theta_s}(r_s|\mathbf{z}_s, \mathbf{c}_s)$ and $p_{\theta_t}(r_t|\mathbf{z}_t, \mathbf{c}_t)$ by sampling $\mathbf{z}_s$ and $\mathbf{z}_t$. The optimization objective of each CVAE is the evidence lower bound (ELOB) [43], which consists of the sum of the reconstruction error and the negative KL divergence between the variational posterior and the prior. Thus, the loss function in the source domain can be written as follows [42]:

$$\begin{aligned}\mathcal{L}_S(r_s, \mathbf{x}_s; \theta_s, \phi_s) = \mathbb{E}_{q_{\phi_s}(\mathbf{z}_s|r_s,\mathbf{x}_s)}[\log p_{\theta_s}(r_s|\mathbf{z}_s, \mathbf{c}_s)] \\ - D_{KL}[q_{\phi_s}(\mathbf{z}_s|r_s, \mathbf{c}_s)||p(\mathbf{z}_s)]. \end{aligned} \tag{2}$$

Similarly, the loss function in the target domain can be written as:

$$\begin{aligned}\mathcal{L}_T(r_t, \mathbf{x}_t; \theta_t, \phi_t) = \mathbb{E}_{q_{\phi_t}(\mathbf{z}_t|r_t,\mathbf{x}_t)}[\log p_{\theta_t}(r_t|\mathbf{z}_t, \mathbf{c}_t)] \\ - D_{KL}[q_{\phi_t}(\mathbf{z}_t|r_t, \mathbf{c}_t)||p(\mathbf{z}_t)]. \end{aligned} \tag{3}$$

We apply alternating optimization to learn the embedding encoders $q_{\phi_{x_s}}$ and $q_{\phi_{x_t}}$ with the mean square error (MSE) as follows:

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

The multi-view information bottleneck (MIB) constraint [27] is used to learn domain-shared properties and discard domain-specific properties in this component. MIB is an effective information-theoretic tool to reserve relevant information of two views for predicting the unknown label and minimizing superfluous information. Therefore, we can treat a shared user from the source and target domain as two views. These two views can be regarded as the same class with the same unknown label. Thus, we can use MIB to reserve domain-shared information and drop out domain-specific information which is not shared by both views. This relies on two basic assumptions: the first is that each domain has domain-specific information, and the second is that two relevant domains share invariant factors that are transferable from the source domain to the target domain [44]. Therefore, the goals of MIB are twofold: (1) aligning the two distributions of latent representations from the source and target domains with domain-shared information and (2) minimizing the irrelevant domain-specific information for a better distinguishable prior learning. Specifically, we regard the ratings r_s from the source domain and r_t from target domains as two views of the shared user's preference. In particular, we assume that r_s and r_t have the same unknown label, i.e., these preferences can be seen as the same class and have some overlapped properties that can be learned by $\mathbf{z}_s$ and $\mathbf{z}_t$.

The MIB objective regarding $\mathbf{z}_s$ and $\mathbf{z}_t$ is as follows:

$$\begin{aligned}\mathcal{L}_{MIB}(\phi_s, \phi_t) = -I_{\phi_s,\phi_t}(\mathbf{z}_s, \mathbf{z}_t) \\ + \beta D_{SKL}(p_{\phi_s}(\mathbf{z}_s|r_s, \mathbf{x}_s)||(p_{\phi_t}(\mathbf{z}_t|r_t, \mathbf{x}_t)), \end{aligned} \tag{5}$$

where the function $I(\cdot)$ denotes the mutual information between inputs [27]. D_{SKL} denotes the symmetrized KL divergence for joint observations of ratings r_s and r_t, and D_{SKL} is computed by averaging the expected value of two KL divergence terms:

$$\begin{aligned}&D_{SKL}(p_{\phi_s}(\mathbf{z}_s|r_s, \mathbf{x}_s)||(p_{\phi_t}(\mathbf{z}_t|r_t, \mathbf{x}_t)) \\ &= \frac{1}{2}D_{KL}(p_{\phi_s}(\mathbf{z}_s|r_s, \mathbf{x}_s)||p_{\phi_t}(\mathbf{z}_t|r_t, \mathbf{x}_t)) \\ &+ \frac{1}{2}D_{KL}(p_{\phi_t}(\mathbf{z}_t|r_t, \mathbf{x}_t)||p_{\phi_s}(\mathbf{z}_s|r_s, \mathbf{x}_s)). \end{aligned} \tag{6}$$

The MIB objective is employed to learn domain-shared representations, which enables the dual CVAE to transfer users' preferences from the source domain to the target domain by the first term of Eq. (5). The second term of Eq. (5) aims to discard domain-specific information for learning compact but informative latent representations. Therefore, the coefficient β of Eq. (5) is a hyper-parameter that is used to trade-off between domain adaptation with shared properties and sufficient latent representations. The symmetrized KL divergence D_{SKL} can be computed directly with the learned distributions $p_{\phi_s}(\mathbf{z}_s|r_s, \mathbf{x}_s)$ and $p_{\phi_t}(\mathbf{z}_t|r_t, \mathbf{x}_t)$. The mutual information between the two learned representations $I_{\phi_s,\phi_t}(\mathbf{z}_s, \mathbf{z}_t)$ can be maximized by using a sample-based differentiable mutual information lower bound. In this paper, we use the InfoNCE estimator [45] implemented by a 3-layer fully connected neural network.

In our dual CVAE component, the MIB objective $\mathcal{L}_{MIB}(\phi_s, \phi_t)$ constrains the encoders ϕ_s and ϕ_t by treating the ratings r_s and r_t as two views with the same label. However, it does not constrain the decoders θ_s and θ_t, which we used to generate the plausible ratings. To align the distributions of both decoders θ_s and θ_t, we add a similarity (SIM) objective to constrain them. Specifically, we treat the latent representations $\mathbf{z}_s$ and $\mathbf{z}_t$ as the same class with the same labels r_s and r_t for the decoders θ_s and θ_t to learn. Therefore, the similarity loss can be defined as:

$$\begin{aligned}&\mathcal{L}_{SIM}(r_s, \mathbf{x}_s, r_t, \mathbf{x}_t; \theta_s, \phi_s, \theta_t, \phi_t) \\ &= \mathbb{E}_{q_{\phi_s}(\mathbf{z}_s|r_s,\mathbf{x}_s)}[\log p_{\theta_s}(r_s|\mathbf{z}_s, \mathbf{c}_s)] \\ &+ \mathbb{E}_{q_{\phi_t}(\mathbf{z}_t|r_t,\mathbf{x}_t)}[\log p_{\theta_t}(r_t|\mathbf{z}_t, \mathbf{c}_t)] \\ &+ \mathbb{E}_{q_{\phi_s}(\mathbf{z}_s|r_s,\mathbf{x}_s)}[\log p_{\theta_t}(r_t|\mathbf{z}_t, \mathbf{c}_t)] \\ &+ \mathbb{E}_{q_{\phi_t}(\mathbf{z}_t|r_t,\mathbf{x}_t)}[\log p_{\theta_s}(r_s|\mathbf{z}_s, \mathbf{c}_s)]. \end{aligned} \tag{7}$$

However, the terms $\mathbb{E}_{q_{\phi_s}(\mathbf{z}_s|r_s,\mathbf{x}_s)}[\log p_{\theta_s}(r_s|\mathbf{z}_s, \mathbf{c}_s)]$ and $\mathbb{E}_{q_{\phi_t}(\mathbf{z}_t|r_t,\mathbf{x}_t)}[\log p_{\theta_t}(r_t|\mathbf{z}_t, \mathbf{c}_t)]$ are the reconstruction losses of $\mathcal{L}_S$ and $\mathcal{L}_T$. We delete them from the similarity objective $\mathcal{L}_{SIM}$, and rewrite the similarity loss as:

$$\begin{aligned}\mathcal{L}_{SIM} &= \mathbb{E}_{q_{\phi_s}(\mathbf{z}_s|r_s,\mathbf{x}_s)}[\log p_{\theta_t}(r_t|\mathbf{z}_t, \mathbf{c}_t)] \\ &+ \mathbb{E}_{q_{\phi_t}(\mathbf{z}_t|r_t,\mathbf{x}_t)}[\log p_{\theta_s}(r_s|\mathbf{z}_s, \mathbf{c}_s)]. \end{aligned} \tag{8}$$

In summary, the cross-domain adaptation objective of MetaCAR is formulated as:

$$\mathcal{L}_{MetaCAR} = \mathcal{L}_S + \mathcal{L}_T + \mathcal{L}_X + \eta\mathcal{L}_{MIB} + \mathcal{L}_{SIM}, \tag{9}$$

where η is a hyper-parameter. We set the hyper-parameter of $\mathcal{L}_{SIM}$ to 1 according to Eq. (7).

4.2 Meta-Augmentation with Prior

In this subsection, we augment ratings in the target domain to enrich interaction data and generate ME tasks for alleviating the meta-overfitting problem. With such a problem alleviated, meta-learning methods show great potential for solving cold-start issues. Simply generating noise labels is proven to be effective for alleviating meta-overfitting [24], as introduced in Section 1. However, the ratings and the user-item pairs have strong connections in recommendations, and noise ratings may lead to a meaningless result. Traditional data augmentation methods such as [25], [26] all generate new samples with new user-item pairs and new ratings, e.g., plausible ratings are augmented for new user and item interactions that are not observed in the true rating matrix. These methods provide each user-item pair with only one rating, and hence we still cannot construct ME tasks by utilizing those traditional data augmentation methods. In contrast, we learn prior knowledge with a dual CVAE trained on shared users, and then generate plausible ratings that are meaningful but distinguishable from true ratings.

Specifically, as shown in the red dashed part of Figure 2, we generate rating $\hat{r}_t$ for a specific user u with learned CVAE by the content $\mathbf{x}_t$ of user u and item i. Note that, we use only one interaction to denote user preference for simplicity. This creates a new rating for the same content $\mathbf{x}_t$, which is reasonable because a user's preferences should vary in the real world, and users and items that have similar content data might lead to different ratings. These ratings denote the preferences of user u in the source domain by transferring preferences, which enrich the interaction data for the target domain. It is worth noting that we augment ratings for existing user-item pairs in the target domain. To facilitate the construction of meta-learning tasks, we treat these different ratings as another task for the same user. That is to say, we have different ratings for the same user and construct different tasks, which are ME. We call this process meta-augmentation with prior.

4.3 Meta-Learning with Augmented Data

Considering the preference of user u of the target domain as a task $\mathcal{T}$, we construct a new task $\mathcal{T}_a$ from the same input $\mathbf{x}_t$ of task $\mathcal{T}$, by using the plausible ratings generated from the dual CVAE with learned prior. Then, task $\mathcal{T}$ and the augmented task $\mathcal{T}_a$ can be expressed as:

$$\mathcal{T} = \{\underbrace{(x_t, r_t)}_{\text{true ratings}}\}, \tag{10}$$

$$\mathcal{T}_a = \{\underbrace{(x_t, \hat{r}_t)}_{\text{plausible ratings}}\}, \tag{11}$$

where $\hat{r}_t$ is the plausible ratings rated by the user u. We consider a meta-learning optimization method MAML [19] that trains the rating prediction model f on training task set D_{train}, which includes the original tasks $\mathcal{T}$ and augmented tasks $\mathcal{T}_a$. As implicit feedback is taken into consideration in our work, we use the binary cross-entropy loss as the objective of rating prediction.

5 EXPERIMENTS

We performed extensive experiments to justify our claim that the proposed MetaCAR is capable of addressing meta-overfitting in meta-learning recommendation over the cold-start user setting. The goal of our experiments is to show how the proposed method alleviates the meta-overfitting problem by evaluating the recommendation performance for cold-start users. To this end, we compare the proposed MetaCAR with four types of competing recommendation baselines: (1) content-aware recommendation methods represented by CDL [4], (2) cross-domain recommendation methods including TDAR [33], ETL [18] and DARec [17], (3) meta-learning-based recommendations including MeLU [21], and (4) matrix-factorization-based methods represented by NeuMF [1].

5.1 Experimental Settings

5.1.1 Datasets

We evaluate the performance of the proposed MetaCAR and the competing baselines on the Amazon dataset[1], which contains user reviews and metadata from the e-commerce site Amazon.com, and the Douban dataset, which was crawled from the Douban website[2]. The Amazon dataset covers user interactions on items as well as item content on 24 product categories. For the Amazon dataset, we chose four different categories: Electronics, Movies, Music, and CDs, which are subsets of the Amazon dataset. The Douban dataset includes three categories: DoubanMovie, DoubanBook, and DoubanMusic. We adopt the same strategy to filter these datasets such that the remaining users and items have at least $k = 5$ reviews each. Statistics are shown in Table 2. Note that these datasets are usually used in recent works for cross-domain recommendation [17], [18], [46], [47]. To test the cross-domain performance, we select Electronics, Movies, and Music as three source (target) domains and CDs as the target (source) domain for the

1. http://jmcauley.ucsd.edu/data/amazon/
2. https://www.douban.com/

TABLE 2
Statistics of datasets.

Dataset		#User	#Item	#Rating	Sparsity
Amazon	Movies	123,960	50,052	1,697,438	99.97%
	Electronics	192,402	63,001	1,687,993	99.98%
	Music	5,541	3,568	64,705	99.67%
	CDs	25,400	24,904	43,903	99.99%
Douban	DoubanMusic	1,672	5,567	69,709	99.25%
	DoubanBook	2,110	6,777	96,041	99.33%
	DoubanMovie	2,712	34,893	1,278,401	98.65%

TABLE 3
Statistics of user overlaps.

Amazon	CDs		
	Shared Users	Ratio	Ratio (CDs)
Movies	18031	14.54%	23.96%
Electronics	6260	3.25%	8.32%
Music	5331	96.21%	7.08%
Douban	DoubanMusic		
	Shared Users	Ratio	Ratio (DoubanMusic)
DoubanBook	1815	66.92%	99.72%
DoubanMovie	1736	78.48%	95.38%

Amazon dataset, and select DoubanMovie and DoubanBook as two source (target) domains and DoubanMusic as the target (source) domain for the Douban dataset. Following [17], we find the shared users between the source and target domains for cross-domain recommendation learning. Statistics of overlaps are shown in Table 3. For each dataset, we split it into training set, validation set, and test set randomly. The training set includes all shared users in target domains.

5.1.2 Meta-Training Tasks

In the training set, we follow the protocol proposed in [21] to construct a meta-learning task in a recommendation system. Specifically, since Amazon has explicit feedback ratings, to construct implicit feedback, we set the rating r_{ij} as "1" if user i interacts with item j and "0" otherwise. According to the experimental results of NeuMF [1] in which the optimal sampling ratio between negative and positive instances is approximately 3 to 6, we choose a ratio of 5 to construct a meta-learning task. Specifically, we implement it by using the tuple that includes one positive instance and five negative instances. Each task consists of tuples. The positive instances are randomly sampled from the interaction history, and the negative instances are randomly sampled from the missing data [48], which are positive instances of other users. We adopt the same strategy to construct the augmented tasks.

For each training task, MeLU [21] samples fewer items as the query set $\mathcal{Q}$ and the rest of (most) items as the support set $\mathcal{S}$. However, the main contribution of the meta-gradient of MAML comes from the outer loop, computed with the query set [49]. Therefore, we use a quarter of the task data as the support set and the remaining data as the query set in the meta-training phase.

5.1.3 Evaluation Protocols

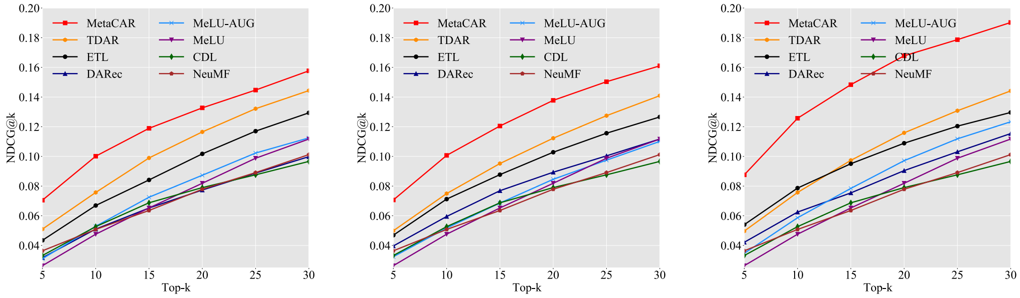
In the meta-testing phase, to evaluate the performance of MetaCAR, we follow the common strategy [1], [50], which adopts the *leave-one-out* evaluation. Specifically, we randomly sample 99 negative items that have no interaction with the user, ranking one test (positive) item among 100 items. These 100 items constitute the query set of the test task. To comply with the few-shot (cold-start) problem setting and simulate real-world applications that contain very few interactions, we limit the number of tuples used for fine-tuning the test task to a maximum of five. These fine-tuning data constitute the support set of the test task. After the test task construction, we fix the test task for performance evaluation. More precisely, all methods in our experiments use exactly the same fine-tuning data and test data.

The performance of a ranked list, which includes 100 items in the query set of the test task, is judged by *hit ratio* (HR), *mean reciprocal rank* (MRR), *normalized discounted cumulative gain* (NDCG) [51] , and *F-score* (F1). We truncate the ranked list at k for each metric and finally use the average metrics of all users to report the performance of the models.

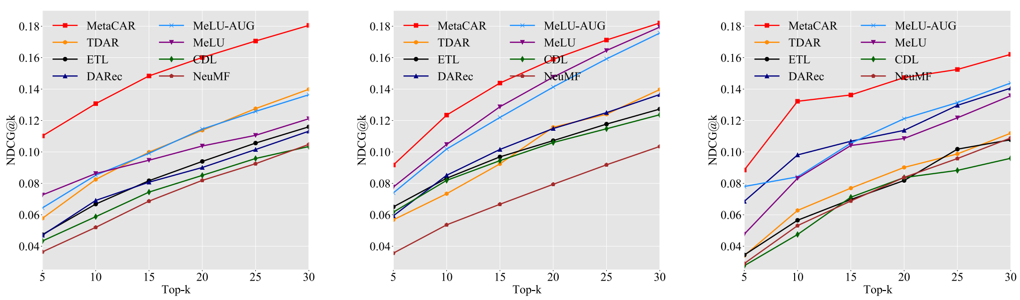
5.1.4 Baselines

As introduced in Section 1, we choose four types of comparison baselines:

- **TDAR**: Text-enhanced domain adaptation recommendation [33] is a distribution pattern transfer method. It adopts domain-invariant textual features as the anchor points to align the latent space embeddings and then feeds them into a domain classifier for domain adaptation.
- **DARec**: Domain adaptation recommendation [17] extracts and transfers patterns with an adversarial learning process from rating matrices of shared users only without relying on any auxiliary information. For a fair comparison, we pretrain the target domain part of DARec with data of the training set and then train DARec with shared user representations across different domains.
- **ETL**: The equivalent transformation learner [18] learns both the overlapped and domain-specific properties for cross-domain recommendation by modeling the joint distribution of user behaviors across different domains. We adopt the same strategy of DARec to train ETL for a fair comparison.
- **CDL**: Collaborative deep learning [4] is a hierarchical Bayesian-based method that has been explored by coupling deep learning for content information and collaborative filtering (CF) for the ratings matrix. For a fair comparison, we replace the content embedding with text memory network (TMN) [33] to extract effective word semantic vectors. Moreover, we obtain the ratings of the items of the query set from the ratings matrix to test the performance.
- **MeLU**: Meta-learned user preference estimator [21] is a state-of-the-art meta-learning method that adapts MAML for solving the cold-start issue by treating it as a few-shot task in recommendation. For a fair comparison, we adopt the same binary cross-entropy loss and neural network architecture with MetaCAR meta-learning component.
- **NeuMF**: Neural collaborative filtering [1] is a neural network-based matrix factorization method that uses a neural network to model the interaction between user and item embeddings by wide & deep structures for prediction ratings.

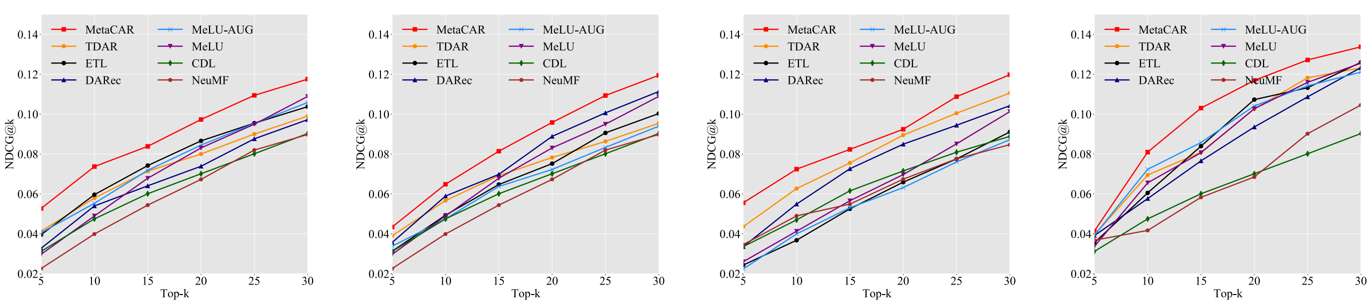


(a) Electronic-to-CDs. (b) Movies-to-CDs. (c) Music-to-CDs.



(d) CDs-to-Electronic. (e) CDs-to-Movies. (f) CDs-to-Music.

Fig. 3. Performance comparison on the Amazon dataset.



(a) DoubanBook-to-DoubanMusic. (b) DoubanMovie-to-DoubanMusic. (c) DoubanMusic-to-DoubanBook. (d) DoubanMusic-to-DoubanMovie.

Fig. 4. Performance comparison on the Douban dataset.

5.1.5 Model and Hyper-Parameter Settings

In the domain adaptation component, the model is optimized with the Adam optimizer. The number of training epochs is set to 50. The batch size is set to 64 for Music and Electronics and 16 for Movies. To better extract domain-invariant features, we adopt a memory structure text memory network (TMN) [33] to extract textual features by mapping user and item content information (reviews) into the word semantic space and linearly combining word semantic vectors. In TMN, the word semantic matrix, used to calculate weights for users and items, is pretrained on GoogleNews corpus by word2vec [52] for English reviews and Tencent AI Lab Embedding Corpus for Chinese Words and Phrases [53] for Chinese reviews. In each domain, we pretrain TMN and fix it to generate embedding vectors of 300 dimensions (English reviews) and 200 dimensions (Chinese reviews) for each user and item. All encoder and decoder networks are implemented with two-layer MLP, and the embedding size is set to 100. The meta-learning component models via a deep fully connected (FC) structure with 2 layers of size 64. The dimension of user and item embedding vectors is set to 300 for English reviews and 200 for Chinese reviews. We set the step size of the inner loop and outer loop to 0.01 and 0.001. The number of inner-loop updates is set to 5. The batch size and the epochs are set to 64 and 20, respectively.

We tune the optimal hyper-parameters for MetaCAR and other comparison methods with a coarse grain linear search. Specifically, for hyper-parameter β in loss function $\mathcal{L}_{MIB}$, we tune it among [0.001, 0.01, 0.1, 1, 10, 100] according to the validation set. The weight factor for loss functions $\mathcal{L}_{MIB}$ is fixed to 1.0 for simplicity.

5.2 Performance Comparison

In this experiment, we test the performance of our method and baselines on bi-directional cross-domain. The results of the performance comparison are shown in Figure 3 and Table 4 for the Amazon dataset, and Figure 4 and Table 5 for the Douban dataset. We use "source-to-target" to denote the transfer of the "source" domain to the "target" domain. In this part, we first demonstrate the meta-overfitting problem via MeLU, and then compare MetaCAR with other baselines to show its superiority.

5.2.1 Discussions of Meta-overfitting

MeLU is a state-of-the-art method designed to address the cold-start recommendation via meta-learning. However, MeLU will undoubtedly meet meta-overffiting because it trained on Non-ME tasks. On the one hand, MeLU demonstrates the superiority of the meta-learning method over the cold-start issue on the Electronics, Movies, Music, and DoubanMusic datasets, as shown in Figure 3 (d), (e) and (f), Figure 4 (a) and (b), Table 4 and Table 5. Especially on Movies, it outperforms all other baselines except MetaCAR. Those results demonstrate the robustness of MAML to the non-mutually exclusive tasks. However, MeLU has not shown the true strength of meta-learning under Non-ME task scenarios. On the other hand, MeLU performs poorly on the CDs and DoubanBook datasets, as shown in Figure 3 (a), (e) and (f), Figure 4 (c) and (d), Table 4 and Table 5, and it is only comparable to NeuMF and CDL, which demonstrate the worst performance. This is because that MeLU meets serious meta-learning overfitting problem when trained on insufficient and non-mutually exclusive tasks. Such problem seriously limits the generalization ability of meta-learning on the cold-start issue.

MeLU-AUG denotes MeLU is directly trained on the combinations of real interactions from both the source and target domains. It can be treated as meta-augmentation with random ratings, which introduces large interactions from the source domain without any guarantee for the correctness of the ratings. For a fair comparison, it adopts the same loss and architecture as MeLU and meta-learning component of MetaCAR. Compared with MeLU, the performance of MeLU-AUG may demonstrate a slight improvement or deterioration on all datasets except DoubanMovie-to-DoubanMusic and DoubanMusic-to-DoubanBook, in which MeLU-AUG shows severe performance decline, as shown in Figure 3 , Figure 4, Table 4 and Table 5, and it is only comparable to NeuMF and CDL, which demonstrate the worst performance. This validates the intuition that assigning random ratings to the user-item pairs in the recommendation system will lead to a meaningless result. Thus, generating meaningful ratings is very important for constructing effective ME tasks.

In the meta-learning stage, MetaCAR can be considered as MeLU with extra meta-augmentation meaningful tasks. In experiments, MetaCAR outperforms MeLU with remarkable improvements in all metrics on all datasets, no matter how good or bad the MeLU performs. Such results validate the effectiveness of constructing meaningful ME tasks in addressing the meta-overfitting problem. The main drawback of these meta-learning-based recommendation methods, e.g., MeLU, may be due to poor task designing, and directly constructing Non-ME tasks from true ratings. In addition, MetaCAR also obtains significant improvement over MeLU-AUG in all metrics on all datasets. These results validate that carefully constructing ME tasks via meta-augmentation with prior is an effective method to address meta-overfitting. In contrast, constructing ME tasks from random ratings is ineffective.

5.2.2 Performance Comparison with Competing Methods

TDAR is a text-enhanced cross-domain method that provides a valuable component named text memory network (TMN) used in our framework. It obtains the outstanding performance on most datasets except CDs-to-Movies and CDs-to-Music and the second-best on Electronic-to-CDs, Movies-to-CDs, Music-to-CDs, and DoubanMusic-to-DoubanBook. This is because of the effectiveness of the domain adaptation component, designed for sparse datasets, and textual features extracted by TMN. DARec is a rating pattern transfer method, and it can extract patterns that depend on the rating matrices only without any content information. It works well on the CDs-to-Movies and DoubanMusic-to-DoubanBook datasets and obtain the second best performance on the DoubanMovie-to-DoubanMusic dataset. This demonstrates the effectiveness of its strategy. ETL learns both domain-specific and shared properties by modeling the joint distribution of user behaviors across different domains. This strategy is useful, and it obtains third best performance on the Electronic-to-CDs, Movies-to-CDs and Music-to-CDs dataset, as shown in Figure 3 (a), (b) and (c), and Table 4. However, both DARec and ETL did not consider the side information and lead to a lower performance than TDAR in many cases.

MetaCAR can also be seen as a cross-domain method by transferring domain-shared properties via generated ratings. Compared with other cross-domain baselines TDAR, DARec, and ETL, MetaCAR also achieved significant performance improvement in all metrics on all datasets. MetaCAR considers both content information and cross domain to learn domain-shared properties and discard domain-specific properties. Adding other effective strategies, e.g., meta-augmentation and the strong generalization ability, MetaCAR obtains superior performance to TDAR, DARec, and ETL with significant improvement. In contrast, due to meta-overfitting, MeLU performs worse in most cases than cross-domain methods TDAR, DARec, and ETL.

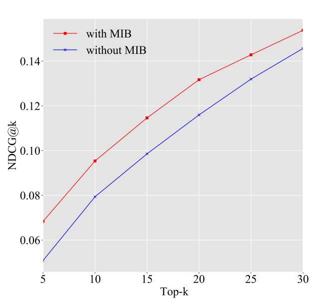
NeuMF is a deep version of the matrix factorization method to learn the latent representations of users and items. CDL is a content-aware recommendation with a deep hierarchical Bayesian model. Both methods are known as collaborative filtering (CF) systems and have proven to be very successful. However, both NeuMF and CDL obtain poor or even worst performance in most cases, as shown in Figure 3 , Figure 4, Table 4 and Table 5. This may be because they both face the serious overfitting problem due to data sparsity and lack of the generalization ability to deal with cold-start.

Compared with NeuMF and CDL, our MetaCAR from all six cross-domains obtained remarkable performance improvement in all metrics, specifically 1.86 times higher in NDCG@20 for the best case (comparing MetaCAR with NeuMF on Music-to-CDs). Moreover, cross-domain baselines generally outperform single-domain baselines except MeLU in most cases of our experiments. This validates the importance of transferring knowledge across domains for the cold-start recommendation.

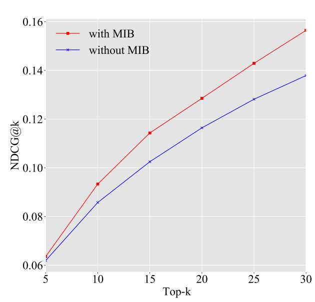
Overall, MetaCAR consistently produces the best performance in all metrics on all datasets with significant improvement except MRR@10 and MRR@20 on the DoubanMusic-to-DoubanMovie dataset, where TDAR obtains the best and

TABLE 4
Performance comparison on the Amazon dataset.

Domain	Metric	HR@10	MRR@10	NDCG@10	F1@10	HR@20	MRR@20	NDCG@20	F1@20
CDs	NeuMF	0.1183	0.0410	0.0583	0.0200	0.2449	0.0497	0.0902	0.0219
	MeLU	0.1139	0.0282	0.0476	0.0207	0.2518	0.0373	0.0819	0.0240
	CDL	0.1183	0.0333	0.0527	0.0215	0.2231	0.0404	0.0790	0.0201
Electronics-to-CDs	TDAR	0.1666	0.0488	0.0757	0.0303	0.3292	0.0597	0.1165	0.0314
	DARec	0.1171	0.0315	0.0510	0.0213	0.2223	0.0386	0.0773	0.0212
	ETL	0.1471	0.0431	0.0669	0.0267	0.2869	0.0523	0.1017	0.0273
	MeLU-AUG	0.1210	0.0330	0.0529	0.0220	0.2579	0.0423	0.0873	0.0246
	MetaCAR	**0.2122**	**0.0665**	**0.1001**	**0.0386**	**0.3425**	**0.0752**	**0.1327**	**0.0326**
Movies-to-CDs	TDAR	0.1658	0.0479	0.0750	0.0301	0.3151	0.0579	0.1123	0.0300
	DARec	0.1320	0.0379	0.0595	0.0240	0.2512	0.0459	0.0894	0.0239
	ETL	0.1552	0.0463	0.0712	0.0282	0.2817	0.0548	0.1028	0.0268
	MeLU-AUG	0.1171	0.0324	0.0517	0.0213	0.2489	0.0412	0.0846	0.0237
	MetaCAR	**0.2074**	**0.0689**	**0.1007**	**0.0377**	**0.3557**	**0.0788**	**0.1378**	**0.0339**
Music-to-CDs	TDAR	0.1708	0.0471	0.0757	0.0311	0.3317	0.0582	0.1159	0.0316
	DARec	0.1330	0.0417	0.0625	0.0242	0.2456	0.0491	0.0904	0.0234
	ETL	0.1711	0.0511	0.0787	0.0311	0.2921	0.0592	0.1089	0.0278
	MeLU-AUG	0.1375	0.0356	0.0587	0.0250	0.2919	0.0459	0.0971	0.0278
	MetaCAR	**0.2684**	**0.0832**	**0.1257**	**0.0488**	**0.4364**	**0.0945**	**0.1678**	**0.0416**
Electronics	NeuMF	0.1106	0.0345	0.0519	0.0154	0.2295	0.0426	0.0818	0.0163
	MeLU	0.1314	0.0728	0.0863	0.0238	0.2022	0.0774	0.1038	0.0193
	CDL	0.1230	0.0394	0.0588	0.0224	0.2270	0.0466	0.0851	0.0216
CDs-to-Electronics	TDAR	0.1689	0.0567	0.0307	0.0825	0.2941	0.0651	0.1138	0.0280
	DArec	0.1468	0.0460	0.0691	0.0267	0.2301	0.0517	0.0901	0.0219
	ETL	0.1385	0.0454	0.0668	0.0252	0.2471	0.0527	0.0939	0.0235
	MeLU-AUG	0.1065	0.1059	0.1060	0.0200	0.1207	0.1067	0.1094	0.0124
	MetaCAR	**0.1964**	**0.1115**	**0.1307**	**0.0357**	**0.3130**	**0.1195**	**0.1601**	**0.0298**
Movies	NeuMF	0.1159	0.0351	0.0536	0.0220	0.2195	0.0420	0.0794	0.0223
	MeLU	0.1931	0.0788	0.1048	0.0351	0.3639	0.0904	0.1476	0.0347
	CDL	0.1604	0.0583	0.0819	0.0292	0.2571	0.0648	0.1060	0.0245
CDs-to-Movies	TDAR	0.1246	0.0931	0.0734	0.0226	0.1405	0.0943	0.1157	0.0134
	DArec	0.1775	0.0576	0.0851	0.0323	0.2961	0.0657	0.1149	0.0282
	ETL	0.1593	0.0601	0.0832	0.0290	0.2547	0.0665	0.1071	0.0243
	MeLU-AUG	0.1917	0.0756	0.1018	0.0348	0.3498	0.0861	0.1413	0.0333
	MetaCAR	**0.2463**	**0.0864**	**0.1234**	**0.0448**	**0.3878**	**0.0961**	**0.1590**	**0.0369**
Music	NeuMF	0.1230	0.0327	0.0531	0.0224	0.2460	0.0410	0.0838	0.0238
	MeLU	0.1868	0.0527	0.0832	0.0340	0.2872	0.0596	0.1085	0.0273
	CDL	0.1094	0.0292	0.0475	0.0199	0.2521	0.0391	0.0837	0.0240
CDs-to-Music	TDAR	0.1516	0.0365	0.0628	0.0276	0.2611	0.0439	0.0902	0.0249
	DArec	0.2142	0.0632	0.0981	0.0390	0.2764	0.0674	0.1137	0.0263
	ETL	0.1310	0.0350	0.0565	0.0238	0.2323	0.0418	0.0818	0.0221
	MeLU-AUG	0.1215	0.0727	0.0841	0.0221	0.2681	0.0828	0.1212	0.0255
	MetaCAR	**0.2333**	**0.1036**	**0.1322**	**0.0424**	**0.2955**	**0.1074**	**0.1473**	**0.0281**



(a) Electronics-to-CDs.



(b) Movies-to-CDs.

Fig. 5. Effect of multi-view information bottleneck.

MetaCAR obtains second best performance. The superior performance of MetaCAR over all competing baselines indicates the importance of meta-augmentation, which can effectively prevent meta-overfitting and overlap the data distribution of new tasks for better performance.

5.3 Effectiveness of Multi-view Information Bottleneck

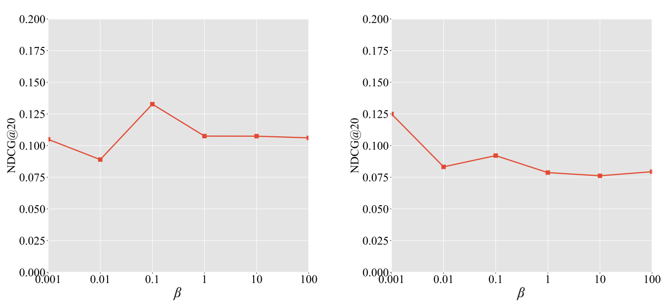
In this part, we discuss the effectiveness of multi-view information bottleneck (MIB). We test the proposed framework with and without the MIB loss on the Electronics-to-CDs, Movies-to-CDs, and Music-to-CDs datasets, and the results are shown in Figure 5. When this constraint is deleted, the performance of MetaCAR significantly declines. It only gains a small performance improvement over TDAR in NDCG@20 (0.1274 versus 0.1159). MetaCAR (0.1434) with MIB outperforms MetaCAR without MIB (0.1274) by more than 12.5% in NDCG@20. This validates the effectiveness of the MIB loss in our framework. However, our framework can still perform well without MIB and demonstrates better performance than baselines. This is because the similarity constraint and the gap between content and ratings can still ensure the meaningfulness and distinguishment of the learned prior. Furthermore, it validates the robustness of the proposed MetaCAR.

5.4 Sensitivity of Hyper-Parameter

The coefficient β is a hyper-parameter to trade-off between sufficient domain-shared properties and robustness

TABLE 5
Performance comparison on the Douban dataset.

Domain	Metric	HR@10	MRR@10	NDCG@10	F1@10	HR@20	MRR@20	NDCG@20	F1@20
DoubanMusic	NeuMF	0.0964	0.0234	0.0399	0.0175	0.2064	0.0307	0.0673	0.0196
	MeLU	0.1151	0.0294	0.0490	0.0209	0.2521	0.0384	0.0831	0.0240
	CDL	0.1000	0.0320	0.0475	0.0182	0.1951	0.0385	0.0715	0.0186
DoubanBook -to- DoubanMusic	TDAR	0.1205	0.0393	0.0580	0.0219	0.2077	0.0454	0.0800	0.0198
	DARec	0.1320	0.0310	0.0540	0.0240	0.2121	0.0363	0.0739	0.0201
	ETL	0.0986	0.0276	0.0439	0.0179	0.2029	0.0344	0.0697	0.0192
	MeLU-AUG	0.1109	0.0388	0.0553	0.0202	0.226	0.0469	0.0845	0.0215
	METACAR	**0.1503**	**0.0509**	**0.0737**	**0.0273**	**0.2461**	**0.0571**	**0.0973**	**0.0234**
DoubanMovie -to- DoubanMusic	TDAR	0.1223	0.0372	0.0567	0.0222	0.2082	0.0430	0.0783	0.0198
	DARec	0.1373	0.036	0.0589	0.0250	0.2585	0.0438	0.0889	0.0245
	ETL	0.1091	0.0319	0.0493	0.0198	0.2120	0.0389	0.0752	0.0201
	MeLU-AUG	0.0951	0.0331	0.0475	0.0173	0.1927	0.0400	0.0723	0.0182
	METACAR	**0.1475**	**0.0400**	**0.0649**	**0.0268**	**0.2721**	**0.0483**	**0.0959**	**0.0259**
DoubanBook	NeuMF	0.0842	0.0374	0.0490	0.0168	0.1421	0.0448	0.0674	0.0202
	MeLU	0.0923	0.0262	0.0413	0.0168	0.2055	0.0339	0.0697	0.0196
	CDL	0.1064	0.0297	0.0476	0.0193	0.1807	0.0350	0.0664	0.0171
DoubanMusic -to- DoubanBook	TDAR	0.1349	0.0410	0.0627	0.0245	0.2426	0.0482	0.0895	0.0230
	DARec	0.0777	0.0247	0.0368	0.0141	0.1948	0.0324	0.0659	0.0185
	ETL	0.1100	0.0326	0.0503	0.0200	0.2125	0.0396	0.0760	0.0201
	MeLU-AUG	0.0890	0.0256	0.0400	0.0162	0.1814	0.0319	0.0632	0.0173
	METACAR	**0.1377**	**0.0529**	**0.0724**	**0.0250**	**0.2179**	**0.0582**	**0.0924**	**0.0208**
DoubanMovies	NeuMF	0.0729	0.0314	0.0417	0.0243	0.1571	0.0423	0.0685	0.0285
	MeLU	0.0944	0.0375	0.0656	0.0172	0.2451	0.0372	0.1027	0.0233
	CDL	0.1119	0.0327	0.0508	0.0203	0.2060	0.0389	0.0742	0.0196
DoubanMusic -to- DoubanMovie	TDAR	0.1175	**0.0553**	0.0695	0.0214	0.2529	**0.0638**	0.1026	0.0240
	DARec	0.1265	0.0372	0.0577	0.0230	0.2707	0.0468	0.0936	0.0257
	ETL	0.1424	0.0366	0.0605	0.0259	0.3174	0.0522	0.1073	0.0312
	MeLU-AUG	0.0811	0.0399	0.0724	0.0148	0.2102	0.0482	0.1041	0.0200
	METACAR	**0.1993**	0.0464	**0.0810**	**0.0362**	**0.3424**	0.0562	**0.1169**	**0.0325**



(a) Electronics-to-CDs. (b) Movies-to-CDs.

Fig. 6. Sensitivity of hyper-parameter β.

representation. We apply a linear search in the coarse-grain range of $\{0.001, 0.01, 0.1, 1, 10, 100\}$. The results of Electronics-to-CDs and Movies-to-CDs are shown in Figure 6. We can see that MetaCAR has varying sensitivity to the hyper-parameter β on different datasets. After grid searching, we obtain $\beta = 0.1$ for Electronics-to-CDs, $\beta = 0.001$ for Movies-to-CDs, $\beta = 1$ for Music-to-CDs, $\beta = 0.1$ for DoubanBook-to-DoubanMusic and $\beta = 0.01$ for DoubanMovie-to-DoubanMusic. In addition, as for bi-directional cross-domain, we set $\beta = 0.1$ for CDs-to-Electronics, $\beta = 0.001$ for CDs-to-Movies, $\beta = 1$ for CDs-to-Music, $\beta = 0.1$ for DoubanMusic-to-DoubanBook and $\beta = 0.01$ for DoubanMusic-to-DoubanMovie.

6 Conclusion

In this paper, we propose a cross-domain meta-augmentation model for content-aware recommendation (MetaCAR) to address two types of meta-overfitting on cold-start recommendations. Specifically, we adopt a domain adaptation component, implemented by dual conditional variational autoencoders (CVAEs) and constrained by MIB, to learn a prior. Then, such a prior is used to plausible ratings to constructing ME tasks for meta-learning. Extensive experiments on ten datasets demonstrate that MetaCAR is an effective method to address the two forms of meta-overfitting and can significantly improve the meta-learning recommendation performance in cold-start scenarios. Although MetaCAR can benefit from its auxiliary domains, what criteria are to be used to determine the optimal domain is still open. We leave this problem for future research.

Acknowledgments

This work was supported by the Major Project for New Generation of AI (grant No. 2018AAA0100400), the National Natural Science Foundation of China (grants No. 62276047, No. 62002052, No. 82121003, No. 61801060 and No. 62176047), Sichuan Science and Technology Program (grant No. 2021JDRC0079, No. 2022YFG0189, No. 2022YFS0600, and No. 2021YFS0374), the Fourth Postdoctoral Innovative Talent Support Program (grant No. BX20190061), Beijing Natural Science Foundation (Z190023), the China Postdoctoral Science Foundation (grants No. 2022M710614, No. 2020M673183, No. 2019TQ0051) and the Australian Research Council (grants No. DP180100106 and No. DP200101328), and Medico-Engineering Cooperation Funds from University of Electronic Science and Technology of China (No. ZYGX2021YGLH208), and the Open Fund of Intelligent Terminal Key Laboratory of Sichuan Province (Grant No. SCITLAB-1017).

REFERENCES

[1] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T. Chua, "Neural collaborative filtering," in *WWW 2017, Perth, Australia, April 3-7, 2017*. ACM, 2017, pp. 173–182.

[2] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 11, pp. 3371–3408, 2010.

[3] C. Wang and D. M. Blei, "Collaborative topic modeling for recommending scientific articles," in *SIGKDD 2011, San Diego, CA, USA, August 21-24, 2011*, 2011, pp. 448–456.

[4] H. Wang, N. Wang, and D. Yeung, "Collaborative deep learning for recommender systems," in *SIGKDD 2015, Sydney, NSW, Australia, August 10-13, 2015*, 2015, pp. 1235–1244.

[5] F. Zhang, N. J. Yuan, D. Lian, X. Xie, and W. Ma, "Collaborative knowledge base embedding for recommender systems," in *SIGKDD, San Francisco, CA, USA, August 13-17, 2016*, 2016, pp. 353–362.

[6] A. P. Singh and G. J. Gordon, "Relational learning via collective matrix factorization," in *SIGKDD 2008, Las Vegas, Nevada, USA, August 24-27, 2008*, 2008, pp. 650–658.

[7] Y. Zhang, B. Cao, and D. Yeung, "Multi-domain collaborative filtering," in *UAI 2010, Catalina Island, CA, USA, July 8-11, 2010*, 2010, pp. 725–732.

[8] Y. Zheng, S. Liu, Z. Li, and S. Wu, "Cold-start sequential recommendation via meta learner," in *AAAI 2021*. AAAI Press, 2021, pp. 4706–4713.

[9] Z. Chen, D. Wang, and S. Yin, "Improving cold-start recommendation via multi-prior meta-learning," in *ECIR 2021, Virtual Event, March 28 - April 1, 2021, Proceedings, Part II*, ser. Lecture Notes in Computer Science, vol. 12657. Springer, 2021, pp. 249–256.

[10] L. Wang, B. Jin, Z. Huang, H. Zhao, D. Lian, Q. Liu, and E. Chen, "Preference-adaptive meta-learning for cold-start recommendation," in *IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*. ijcai.org, 2021, pp. 1607–1614.

[11] L. Zheng, V. Noroozi, and P. S. Yu, "Joint deep modeling of users and items using reviews for recommendation," in *WSDM 2017, Cambridge, United Kingdom, February 6-10, 2017*, 2017, pp. 425–434.

[12] D. Liu, J. Li, B. Du, J. Chang, and R. Gao, "DAML: dual attention mutual learning between ratings and reviews for item recommendation," in *SIGKDD 2019, Anchorage, AK, USA, August 4-8, 2019*, 2019, pp. 344–352.

[13] M. Kolahkaj, A. Harounabadi, A. Nikravanshalmani, and R. Chinipardaz, "Incorporating multidimensional information into dynamic recommendation process to cope with cold start and data sparsity problems," *J. Ambient Intell. Humaniz. Comput.*, vol. 12, no. 10, pp. 9535–9554, 2021.

[14] C. Musto, P. Lops, M. de Gemmis, and G. Semeraro, "Context-aware graph-based recommendations exploiting personalized pagerank," *Knowl. Based Syst.*, vol. 216, p. 106806, 2021.

[15] M. Kolahkaj, A. Harounabadi, A. Nikravanshalmani, and R. Chinipardaz, "A hybrid context-aware approach for e-tourism package recommendation based on asymmetric similarity measurement and sequential pattern mining," *Electron. Commer. Res. Appl.*, vol. 42, p. 100978, 2020.

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

[17] F. Yuan, L. Yao, and B. Benatallah, "Darec: Deep domain adaptation for cross-domain recommendation via transferring rating patterns," in *IJCAI 2019, Macao, China, August 10-16, 2019*, 2019, pp. 4227–4233.

[18] X. Chen, Y. Zhang, I. W. Tsang, Y. Pan, and J. Su, "Towards equivalent transformation of user preferences in cross domain recommendation," *ACM Trans. Inf. Syst.*, 2022. [Online]. Available: https://doi.org/10.1145/3522762

[19] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, 2017, pp. 1126–1135.

[20] M. Vartak, A. Thiagarajan, C. Miranda, J. Bratman, and H. Larochelle, "A meta-learning perspective on cold-start recommendations for items," in *NIPS 2017, December 4-9, 2017, Long Beach, CA, USA*, 2017, pp. 6904–6914.

[21] H. Lee, J. Im, S. Jang, H. Cho, and S. Chung, "Melu: Meta-learned user preference estimator for cold-start recommendation," in *SIGKDD 2019, Anchorage, AK, USA, August 4-8, 2019*, 2019, pp. 1073–1082.

[22] M. Dong, F. Yuan, L. Yao, X. Xu, and L. Zhu, "MAMO: memory-augmented meta-optimization for cold-start recommendation," in *SIGKDD 2020, Virtual Event, CA, USA, August 23-27, 2020*, 2020, pp. 688–697.

[23] Y. Lu, Y. Fang, and C. Shi, "Meta-learning on heterogeneous information networks for cold-start recommendation," in *SIGKDD 2020, Virtual Event, CA, USA, August 23-27, 2020*, 2020, pp. 1563–1573.

[24] J. Rajendran, A. Irpan, and E. Jang, "Meta-learning requires meta-augmentation," in *NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[25] D. Chae, J. Kang, S. Kim, and J. Choi, "Rating augmentation with generative adversarial networks towards accurate collaborative filtering," in *WWW 2019, San Francisco, CA, USA, May 13-17, 2019*, 2019, pp. 2616–2622.

[26] Q. Wang, H. Yin, H. Wang, Q. V. H. Nguyen, Z. Huang, and L. Cui, "Enhancing collaborative filtering with generative augmentation," in *SIGKDD 2019, Anchorage, AK, USA, August 4-8, 2019*, 2019, pp. 548–556.

[27] M. Federici, A. Dutta, P. Forré, N. Kushman, and Z. Akata, "Learning robust representations via multi-view information bottleneck," in *ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*, 2020.

[28] Z. Lyu, M. Yang, and H. Li, "Multi-view group representation learning for location-aware group recommendation," *Inf. Sci.*, vol. 580, pp. 495–509, 2021.

[29] X. Li, C. Chen, P. Zheng, Z. Jiang, and L. Wang, "A context-aware diversity-oriented knowledge recommendation approach for smart engineering solution design," *Knowl. Based Syst.*, vol. 215, p. 106739, 2021.

[30] W. Pan, E. W. Xiang, N. N. Liu, and Q. Yang, "Transfer learning in collaborative filtering for sparsity reduction," in *AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010*, 2010.

[31] Z. Lu, W. Pan, E. W. Xiang, Q. Yang, L. Zhao, and E. Zhong, "Selective transfer learning for cross domain recommendation," in *SIAM 2013, May 2-4, 2013. Austin, Texas, USA*, 2013, pp. 641–649.

[32] T. Man, H. Shen, X. Jin, and X. Cheng, "Cross-domain recommendation: An embedding and mapping approach," in *IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, 2017, pp. 2464–2470.

[33] W. Yu, X. Lin, J. Ge, W. Ou, and Z. Qin, "Semi-supervised collaborative filtering by text-enhanced domain adaptation," in *SIGKDD 2020, Virtual Event, CA, USA, August 23-27, 2020*, 2020, pp. 2136–2144.

[34] C. Yuan, Z. Zhong, C. Lei, X. Zhu, and R. Hu, "Adaptive reverse graph learning for robust subspace learning," *Inf. Process. Manag.*, vol. 58, no. 6, p. 102733, 2021.

[35] C. Wang, M. Niepert, and H. Li, "Recsys-dan: Discriminative adversarial networks for cross-domain recommender systems," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 31, no. 8, pp. 2731–2740, 2020.

[36] J. Zhang, J. Song, L. Gao, Y. Liu, and H. T. Shen, "Progressive meta-learning with curriculum," *IEEE Transactions on Circuits and Systems for Video Technology*, 2022.

[37] H. Bharadhwaj, "Meta-learning for user cold-start recommendation," in *IJCNN 2019 Budapest, Hungary, July 14-19, 2019*, 2019, pp. 1–8.

[38] A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, and T. P. Lillicrap, "Meta-learning with memory-augmented neural networks," in *ICML 2016, New York City, NY, USA, June 19-24, 2016*, 2016, pp. 1842–1850.

[39] L. Chen, L. Wu, K. Zhang, R. Hong, and M. Wang, "Set2setrank: Collaborative set to set ranking for implicit feedback based recommendation," in *SIGIR 2021, Virtual Event, Canada, July 11-15, 2021*. ACM, 2021, pp. 585–594.

[40] M. Yin, G. Tucker, M. Zhou, S. Levine, and C. Finn, "Meta-learning without memorization," in *ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*, 2020.

[41] A. Krishnan, M. Das, M. Bendre, H. Yang, and H. Sundaram, "Transfer learning via contextual invariants for one-to-many cross-domain recommendation," in *SIGIR 2020, Virtual Event, China, July 25-30, 2020*, 2020, pp. 1081–1090.

[42] Y. Wu, C. Macdonald, and I. Ounis, "A hybrid conditional variational autoencoder model for personalised top-n recommendation," in *ICTIR 2020, Virtual Event, Norway, September 14-17, 2020*, 2020, pp. 89–96.

[43] C. Doersch, "Tutorial on variational autoencoders," *CoRR*, vol. abs/1606.05908, 2016.

[44] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks," in *ICML 2015, Lille, France, 6-11 July 2015*, 2015, pp. 97–105.

[45] L. Logeswaran and H. Lee, "An efficient framework for learning sentence representations," in *ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018.

[46] F. Zhu, C. Chen, Y. Wang, G. Liu, and X. Zheng, "DTCDR: A framework for dual-target cross-domain recommendation," in *CIKM 2019, Beijing, China, November 3-7, 2019*. ACM, 2019, pp. 1533–1542.

[47] F. Zhu, Y. Wang, C. Chen, G. Liu, and X. Zheng, "A graphical and attentional framework for dual-target cross-domain recommendation," in *IJCAI 2020*. ijcai.org, 2020, pp. 3001–3008.

[48] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: bayesian personalized ranking from implicit feedback," in *UAI 2009, Montreal, QC, Canada, June 18-21, 2009*, 2009, pp. 452–461.

[49] H. Xu, J. Wang, H. Li, D. Ouyang, and J. Shao, "Unsupervised meta-learning for few-shot learning," *Pattern Recognit.*, vol. 116, p. 107951, 2021.

[50] Y. Koren, "Factorization meets the neighborhood: a multifaceted collaborative filtering model," in *SIGKDD 2008, Las Vegas, Nevada, USA, August 24-27, 2008*, 2008, pp. 426–434.

[51] X. He, T. Chen, M. Kan, and X. Chen, "Trirank: Review-aware explainable recommendation by modeling aspects," in *CIKM 2015, Melbourne, VIC, Australia, October 19 - 23, 2015*, 2015, pp. 1661–1670.

[52] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *NIPS 2013, December 5-8, 2013, Lake Tahoe, Nevada, United States*, 2013, pp. 3111–3119.

[53] Y. Song, S. Shi, J. Li, and H. Zhang, "Directional skip-gram: Explicitly distinguishing left and right context for word embeddings," in *NACAC 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pp. 175–180.

Hui Xu is a joint postdoctor with the University of Electronic Science and Technology of China and Sichuan Artificial Intelligence Research Institute, Yibin, China. His research interests include deep learning, meta-learning and recommendation system.

Changyu Li is currently a Master student with the Center for Future Media, School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China. His research interests include deep learning and computer vision.

Yan Zhang received her Ph.D. degree from the University of Electronic Science and Technology of China in 2019. She is currently working towards the second Ph.D. degree at the University of Technology Sydney. She is also a postdoctor at the University of Electronic Science and Technology of China. Her research interests include recommendation systems, machine learning and data mining. She was a recipient of the fourth Postdoctoral Innovative Talent Support Program of China in 2019.

Lixin Duan received the B.Eng. degree from the University of Science and Technology of China in 2008, and the Ph.D. degree from Nanyang Technological University in 2012. He is currently a Full Professor with the School of Computer Science and Engineering, University of Electronic Science and Technology of China. His research interests include machine learning algorithms (especially in transfer learning and domain adaptation) and their applications in computer vision. He was a recipient of the Microsoft Research Asia Fellowship in 2009 and the Best Student Paper Award at the IEEE Conference on Computer Vision and Pattern Recognition 2010.

Ivor W. Tsang is an ARC Future Fellow and Director of A*STAR Centre for Frontier AI Research (CFAR) since Jan 2022. Previously, he was a Professor of Artificial Intelligence, at University of Technology Sydney (UTS), and Research Director of the Australian Artificial Intelligence Institute (AAII). His research focuses on transfer learning, deep generative models, learning with weakly supervision, big data analytics for data with extremely high dimensions in features, samples and labels. His work is recognised internationally for its outstanding contributions to those fields. In 2013, Prof Tsang received his ARC Future Fellowship for his outstanding research on big data analytics and large-scale machine learning. In 2019, his JMLR paper "Towards ultrahigh dimensional feature selection for big data" received the International Consortium of Chinese Mathematicians Best Paper Award. In 2020, he was recognized as the AI 2000 AAAI/IJCAI Most Influential Scholar in Australia for his outstanding contributions to the field, between 2009 and 2019. His research on transfer learning was awarded the Best Student Paper Award at CVPR 2010 and the 2014 IEEE TMM Prize Paper Award. In addition, he received the IEEE TNN Outstanding 2004 Paper Award in 2007 for his innovative work on solving the inverse problem of non-linear representations. Recently, Prof Tsang was conferred the IEEE Fellow for his outstanding contributions to large-scale machine learning and transfer learning. Prof Tsang serves as the Editorial Board for the JMLR, MLJ, JAIR, IEEE TPAMI, IEEE TAI, IEEE TBD, and IEEE TETCI. He serves as a Senior Area Chair/Area Chair for NeurIPS, ICML, AAAI and IJCAI, and the steering committee of ACML.

Jie Shao received the B.E. degree from Southeast University, Nanjing, China, in 2004 and the Ph.D. degree in computer science from The University of Queensland, Brisbane, Australia, in 2009. He is currently a Professor with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China, and also Executive Deputy Director of Sichuan Artificial Intelligence Research Institute, Yibin, China. He worked as a Research Fellow at the University of Melbourne from 2008 to 2011, and at National University of Singapore from 2012 to 2014. His research interests include database management and multimedia information retrieval.