



PPA++: Preference Prototype-Aware Learning with Large Language Model for Universal Cross-Domain Recommendation

Yuxi Zhang⁶ · Ji Zhang⁶ · Feiyang Xu⁵ · Lvyang Chen⁶ · Bohan Li¹ · Ning Wang⁶ · Huawei Tu² · Lei Guo⁴ · Hongzhi Yin³

Received: 18 March 2025 / Revised: 30 August 2025 / Accepted: 25 September 2025 / Published online: 26 January 2026
© The Author(s) 2026

Abstract

While user preferences are important to cross-domain recommendation (CDR), existing methods primarily discover preferences under specific, yet possibly redundant, item features. To this end, we first propose a novel Preference Prototype-Aware (PPA) learning method to quantitatively learn user preferences while minimizing disturbances from the source domain. It introduces a mix-encoder and a proto-decoder. On the one hand, the mix-encoder learns better general representations of interacted items and captures the intrinsic relationships between items across different domains. On the other hand, the proto-decoder implements a learnable prototype matching mechanism to quantitatively perceive user preferences, avoiding disturbances caused by item features from the source domain. Moreover, through experiments on PPA, we observe another two issues that affect existing CDR methods' performance, i.e., the semantic deficiency caused by sparse item categories and the imbalance weights caused by different user-item distributions. Thus, we further propose a LoRA-based extractor and a domain cross-attention module to alleviate the two issues, respectively. The PPA incorporating with new extractor and attention module is called PPA++. Extensive experiments show that PPA++ outperforms the other state-of-the-art counterparts in four different CDR scenarios.

Keywords Cross-Domain Recommendation · Prototype Learning · Item Similarity Mining · LLM

Yuxi Zhang and Ji Zhang have contributed equally to this work.

✉ Bohan Li
bhli@nuaa.edu.cn

Yuxi Zhang
zyx_xx@nuaa.edu.cn

Ji Zhang
Ji.zhang.nuaa@gmail.com

Feiyang Xu
xufeiyang@zju.edu.cn

Lvyang Chen
cly.ly@nuaa.edu.cn

Ning Wang
nuaawangning@nuaa.edu.cn

Huawei Tu
h.tu@latrobe.edu.au

Lei Guo
leiguoc.cs@gmail.com

Hongzhi Yin
h.yin1@uq.edu.au

- ¹ Key Laboratory of Brain-Machine Intelligence Technology, Ministry of Education, Key Laboratory of Intelligent Decision and Digital Operations, Ministry of Industrial and Information Technology, College of Artificial Intelligence, Nanjing University of Aeronautics and Astronautics, Nanjing, China
- ² Department of Computer Science and Information Technology, La Trobe University, Melbourne, Australia
- ³ The School of Information Technology & Electric Engineering, The University of Queensland, Brisbane, Australia
- ⁴ College of Computer Science and Technology, Shandong Normal University, Jinan, China
- ⁵ Polytechnic Institute, Zhejiang University, Hangzhou, China
- ⁶ College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China

1 Introduction

Recommendation systems (RS) have revolutionized the way users discover and engage with content, by focusing on uncovering latent user preferences and suggesting potentially interesting items. This paradigm has witnessed unprecedented success in various domains, but the understanding of user preferences remains shallow due to the absence of dedicated modules for preference extraction. Cross-domain recommendation (CDR) has emerged as a promising approach to enhance the understanding of user preferences by leveraging data from multiple domains, enabling more personalized and accurate recommendations. However, existing methods primarily focus on dual-domain studies and lack specialized modules for independent user preference modeling. When extracting user preferences across multiple domains, these methods face challenges in identifying genuine user preferences and filtering out redundant item features, rendering them less effective in such scenarios.

To alleviate these issues, several methods construct a dedicated module that can extract user preferences. These methods are built upon unified frameworks, facilitating their application across dual-domain as well as multi-domain real-world scenarios, thereby sorting them as universal CDR methods. The pioneering work of Cao et al. [1] formulates aggregators to obtain specific features of items, while its aggregators merely consist of simple pooling or attention. Inspired by this general and effective framework, Hao et al. [2] exploit fine-tuning to retransfer user preferences. However, this work requires pre-training, which fails to an end-to-end structure and can not directly extract user preferences. Ju et al. [3] propose to mask the user preferences extracted by item representations, aiming to encourage the model to predict user preferences indirectly. The critical limitation of this approach is that the mask mechanism makes it more challenging for datasets with insufficient data.

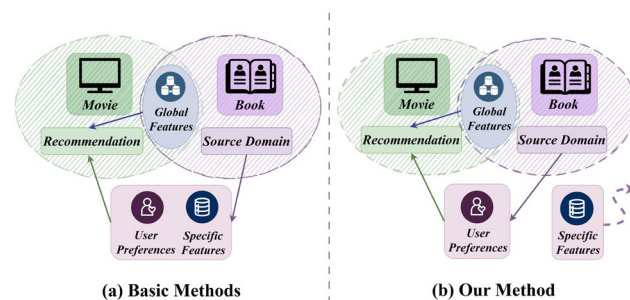


Fig. 1 (a) Existing methods utilize both user preferences and specific features extracted from the source domain. This approach can easily result in suboptimal or even obsolete recommendations due to disturbances from redundant item features; (b) Our method, in contrast, separates user preferences from irrelevant item-specific features, ensuring more reliable recommendations

Despite the promising results achieved, the above works in CDR primarily focus on extracting both user preferences and specific features from the source domain. However, this approach may not effectively capture the true essence of user preferences, as they are more closely associated with interacted items at a higher semantic level rather than specific item features. Overemphasis on item-specific features can lead to suboptimal recommendations and hinder the transferability of user preferences across domains. For instance, consider users who enjoy comedy books. If all the specific features of the books are taken into account, such as the author, genre, or writing style, it can lead to an overly narrow focus on book-related features. This item feature-focused approach can make it challenging to recommend relevant items from other domains, such as movies or TV shows, that align with the user's preference for comedy. The reliance on specific item features can introduce noise and limit the generalizability of the recommendation system. To address this limitation, we advocate for a preference-centric approach to CDR that prioritizes the discovery of user preferences while minimizing the influence of specific item features from the source domain. As illustrated in Fig. 1, our proposed method aims to learn transferable and domain-agnostic user preferences that can effectively bridge the gap between different domains.

In this paper, we first propose a mix-encoder and a proto-decoder that quantitatively learns user preferences while minimizing the impact of source domain disturbances:

Mix-Encoder. Unlike previous simple multiplication between user embedding and item embedding, we introduce the learnable guidance, which extracts the effective specific representation with item scores. This enables the mix-encoder to capture the intrinsic relationships between items accurately and prioritize the most relevant and informative aspects. The final encoded feature is represented by the multiplication between user embedding and item embedding.

Prototype-Aware Decoder. Considering that the recommendation should be separated from irrelevant item features in the source domain, we use prototype learning to represent preferences, rather than directly using item features. The prototype-aware decoder is a learnable component that explicitly models user preferences using representative prototypes. This allows the proto-decoder to effectively capture user preferences while minimizing the influence of redundant or noisy item features.

Armed with the above two modules, we name our architecture as PPA and form a unified end-to-end structure without the need for any pre-training process. While PPA achieves great performance, we also observe two other issues that affect the models' performance. (1) When item categories are scarce, item embeddings are sparsely distributed in the feature space. Existing item embedding methods

can lead to the loss of the semantic content of the items themselves. So the limited item's semantic damages generalization when facing such scenarios. (2) Previous interaction methods between user embedding and item embedding are overly simplistic. The multiplication operations remain unstable when there is a significant disparity in the number of items between users and items. The difference in the density of the two spatial distributions leads to different weight fluctuations, so the model cannot be generalized to scenarios with large distribution differences. Thus, we further propose a LoRA-based extractor and a domain cross-attention module to alleviate the problems.

LoRA-based Extractor. In terms of the previous item extractor methods, we follow the classic embedding approach to project items. Whereas, we observe that the PPA performance is very slightly improved compared to other methods in scenarios with small item categories. Meanwhile, substantial performance disparities exist across all methodologies in scenarios characterized by higher item categories, while their performance converges significantly under limited categories. Hence, we propose to gain more reliable semantics from a large language model to utilize the word's original features and implement Low-rank adaptation (LoRA) to reduce resource consumption.

Domain Cross-Attention Module. In the user-item interaction, the multiplication operations remain unstable as the disparity in the number of users and items grows. As mentioned before, we hope to design an adaptive approach instead of multiplication to balance the weights of user space and item space. To combine such three inputs from different semantic spaces, we designed a domain cross-attention module. Domain cross-attention is inserted after extractors to rebalance the weight distribution.

We improve PPA by introducing these two components, yielding PPA++. Extensive experiments on four different scenarios show that PPA++ outperforms PPA and other existing recommendation methods.

The following sections are organized as follows: the related works and preliminary will be introduced in Sects. 2 and 3. Then, we will retrospect PPA [4] in Sect. 4. PPA++ is introduced in detail in Sect. 5. Thereafter, the experiments will be shown in Sect. 6. Finally, we will present our conclusions and future work in Sect. 7.

2 Related Works

In this section, we provide an overview of the research topics most closely related to our proposed work. To facilitate a good understanding of the CDR field, we categorize the existing approaches into three main directions: dual-domain CDR, multi-domain CDR, and universal CDR. We also

explore the application of prototype learning, particularly in CDR.

2.1 Cross-Domain Recommendation

Conventional cross-domain recommendation (CDR) methods [5, 6] can be roughly classified into two categories: intra-domain (IAD) methods [7, 8] and inter-domain (IED) methods [9–12]. IAD methods mainly focus on addressing data sparsity within the same domain, particularly in unpopular categories where interactions are rare. This challenge leads to difficulties in analyzing the features of these categories. Cao et al. [7] propose decoupling global representations to reinforce user-specific recommendations. Although this method can effectively recommend items to users within the same domain, there is an increasing demand to recommend items from new domains. Recommending items from new domains confronts a more serious challenge known as the cold-start problem. The key limitation of cold-start in IED methods is learning from insufficient historical data about new domains. Man et al. [9] utilize a multi-layer perceptron to capture the cross-domain mapping relationships and select sufficient entities to alleviate the cold-start problem. As the scale of categories and domains increases, it becomes difficult for both of these two types of methods to effectively handle comprehensive scenarios.

Recently, CDR methods have drawn broad attention in addressing multiple target categories, which is a more general and practical problem. Multi-target CDR (MTCDR) aims to concurrently improve recommendations across multiple item categories. These methods primarily employ graph-based approaches and utilize auxiliary information to enhance their recommendations. Cui et al. [13] employ shared graphs and recurrent attention mechanisms to extract multi-target features. Cao et al. [14] incorporate closed-form intra-items and high-order inter-items to describe multiple targets at both fine and coarse levels. However, most of these special MTCDR tricks cannot directly be applied to dual-domain scenarios.

Thus, universal methods have emerged to achieve CDR across various scenarios. UniCDR [1] proposes a unified recommendation framework, which models different scenarios by combining domain-specific and domain-shared features. Inspired by this general and effective framework, Hao et al. [2] utilize motif-based shared embeddings to address the negative transfer problem between domains, effectively improving performance by capturing universal embeddings. Ju et al. [3] randomly mask the user preferences extracted by item representations and train the model to understand them indirectly, thereby reducing the model's reliance on specific item features and enhancing its generalization ability.

However, all of these methods utilize specific item features to roughly represent user preferences while ignoring the significant disturbance from source-domain features and irrelevant specific features. This oversight results in the final suboptimal recommendation over redundant specific features, as the model is unable to effectively distinguish between informative and irrelevant features. Instead, our PPA method can accurately capture user preferences and the generality of interacted items without being disturbed by redundant specific item features. By learning preference prototypes at a higher semantic level, PPA can effectively filter out the noise and focus on the essential aspects of user preferences, which leads to more accurate recommendations across different domains.

2.2 Prototype Learning

Prototype-based learning techniques [15–17] enable deep models to generate output based on a few human-understandable instances. These instances, referred to as prototypes, aim to capture essential characteristics of different classes or groups in the feature space. Therefore, it requires prototypes to be distinguished from other classes and be stable in its own class. Prototype learning was first proposed in the field of image recognition. Chen et al. [15] dissect images by identifying prototype components and integrating their evidence to generate classifications. Xu et al. [17] share global attribute prototypes across different categories to better extend intermediate features to unseen images. The success of prototype learning in the field of image recognition has led to its application in more fields.

In the recommendation field, recent studies have introduced prototype learning to further extract effective representations of users or items. Sankar et al. [18] select each prototype to represent a few-shot item in few-shot recommendations. Melchiorre et al. [19] directly match both users and items to user prototypes and item prototypes, providing effective representations. Gan et al. [20] firstly utilize the target item to pre-train an entire prototype space, which integrates certain semantics of the target embedding. Then, they enhance the final prediction with the prototype-guiding attention mechanism in the fine-tuning phase.

Unlike previous methods in CDR that simply represent prototypes as users or items, our work actively considered the importance of user preferences in cross-domain recommendations. By quantifying user preferences using learnable prototypes, we remove the disturbance from the source-domain items.

2.3 LLM for Cross-Domain Recommendation

Large Language Model (LLM) has recently emerged as a powerful paradigm in numerous research fields. LLM has infused new vitality through its excellent integration capability, driving significant progress in language-related fields. Applying LLMs in recommendation systems (RS) has become a particularly prominent new direction. Leveraging their strength in extracting textual semantics, LLMs are well-suited to capture user interests and preferences required for RS precisely. Wu et al. [21] finetune the domain-specific BERT to address the mismatch between queries and products and employ multi-task learning to optimize multiple objectives simultaneously, thereby developing a general end-to-end learning framework for product search. Additionally, the experimental results of Kang et al. [22] demonstrate that finetuned LLMs with only a small amount of training data could achieve performance comparable to traditional methods, showcasing their potential for data efficiency.

Cross-domain recommendation (CDR), as one of the representative and impactful directions within the field of RS, presents greater challenges and demonstrates higher practical value. CDR necessitates the extraction of user preferences from the source domain and the analysis of item semantic features in the target domain. LLMs offer significant potential to enhance CDR by enabling the extraction and expression of feature correlations across different domains, thereby mitigating key challenges such as cold-start and data sparsity problems. For example, Petruzzelli et al. [23] propose a structured framework to guide LLMs in executing CDR tasks, employing personalized prompts based on user preferences and target items to generate recommendations and explanations in zero-shot and one-shot scenarios. This approach effectively harnesses the knowledge embedded in LLMs to bridge inter-domain gaps, advancing the delivery of personalized and context-aware recommendations.

3 Preliminary

3.1 Problem Statement

For the dual-domain scenario $i \in \{X, Y\}$, the recommendation data consists of users, items and interactions $\mathcal{D}^i = (\mathcal{U}^i, \mathcal{V}^i, \mathcal{E}^i)$. Given the interaction $\mathcal{E}^i$, we follow the previous work [1, 24] to calculate the pre-processed item-item similarity scores $B_g = (B^X, B^Y)$. $B^X \in \mathbb{R}^{|\mathcal{V}^X| \times 1}$ represents the personalized item score within the X domain and B^Y represents ones within the Y domain. These similarity scores are used as prior knowledge to enhance item features Table 1.

Table 1 Key Notations and Descriptions in Sect. 3.1

Notation	Description
i	Domain index
$\mathcal{D}^i$	Recommendation data of domain i
$\mathcal{U}^i$	User set of domain i
$\mathcal{V}^i$	Item set of domain i
$\mathcal{E}^i$	User-item interaction set of domain i
B_g	Global item-item similarity scores
B^X/B^Y	Personalized item score in domain X/Y
B_{i2i}	Item-to-item similarity matrix
A	User-item matrix
λ_F	Frobenius norm
B	Personalized item score
N	Number of sampled items
V_s^i	Specific interacted items
V_g	Global interacted items
u^i/v^i	User/item embedding of domain i
h	Global embedding dimension

Specifically, we introduce a linear model (W_{agg}) to learn the weights B (The detailed derivation process of each step can be referred to [24]) :

$$\begin{aligned}
 & \underset{B_{i2i}}{\operatorname{argmin}} \|A - AB_{i2i}\|_F^2 + \lambda_F \|B_{i2i}\|_F^2 \\
 & B_{i2i} \in \mathbb{R}^{|V| \times |V|}, AB_{i2i} \in \mathbb{R}^{|U| \times |V|} \\
 & \alpha_i = (AB_{i2i})_{u,v_i}, \\
 & \alpha = \operatorname{NORM}(\{\alpha_1, \alpha_2, \alpha_3, \dots\}) \\
 & B = (\{v_1, v_2, v_3, \dots\}, \alpha) \cdot W_{agg},
 \end{aligned} \tag{1}$$

We choose N items from $\mathcal{V}^i$ based on $\mathcal{E}^i$, indicating the specific interacted items V_s^i . We choose $2N$ items from both $\mathcal{V}^X$ and $\mathcal{V}^Y$ based on $\mathcal{E}^i$, indicating the global interacted item V_g . Here, V_s^i with B^i and V_g with B_g constitute the effective item-related inputs.

To further extract their effective representations, we introduce $u^i \in \mathbb{R}^{|U^i| \times h}$ and $v^i \in \mathbb{R}^{|V^i| \times h}$ to denote user and item embedding, where h is the global embedding dimension. After these embedding layers, we get three basic representations for CDR: user embedding u , specific item embedding v_s^i with scores B^i and global item embedding v_g with scores B_g .

We have $u, (v_s^i, B^i)$ and (v_g, B_g) equally in multi-targets scenario. The data distribution characteristics of four different scenarios are summarized in Table 2, which should be considered simultaneously in our universal CDR method.

3.2 Low-Rank Adaptation

Low-rank adaptation (LoRA) [25] is a parameter-efficient fine-tuning technique for LLMs. LoRA integrates trainable low-rank decomposed matrices into transformer layers while retaining original parameters, thereby reducing trainable variables and maintaining computational efficiency without compromising performance.

Specifically, the original weight matrix $W \in \mathbb{R}^{m \times n}$ is decomposed into a product of low-rank submatrices. For example, $W = BA^\top$, where $B \in \mathbb{R}^{r \times k}$ and $A \in \mathbb{R}^{k \times r}$ ($k \ll m, n$). LoRA focuses on adjusting the weights of key pathways (i.e., the submatrices B and A) to update model parameters. So LoRA only fine-tune the decomposed submatrices, reducing both computational complexity and memory usage without compromising overall model performance. Mathematically, the optimization objective for LoRA is defined as:

$$\min_{B,A} \mathcal{L}(x; BA^\top)$$

where $\mathcal{L}$ represents the loss function, x is the input data, and $BA^\top$ denotes the low-rank approximation of the weight matrix. The success of LoRA is evident in its application across various domains such as BERT [26] and GPT [27].

4 Preference Prototype-Aware learning

In this section, we further detail the proposed Preference Prototype-Aware learning (PPA) method. As shown in Fig. 2, PPA consists of two core modules, namely mix-encoder and prototype-aware decoder. In the former module, we employ a mix-network to learn general representations of interacted items and capture the intrinsic relationships between items across different domains. In the latter module, the encoded item features from the source domain are employed to derive the prototype learning. Subsequently, the highest

Table 2 Data format of four common CDR scenarios. Scen. represents Scenario, S. denotes Single and M. denotes Multiple

Scen.	Targets		Overlap		Rec		Model
	S.	M.	User	Item	Intra	Inter	
Scen.1	✓		✓		✓		CoNet
Scen.2	✓		✓			✓	EMCDR
Scen.3		✓		✓	✓		M3Rec
Scen.4		✓	✓		✓		HeroGraph

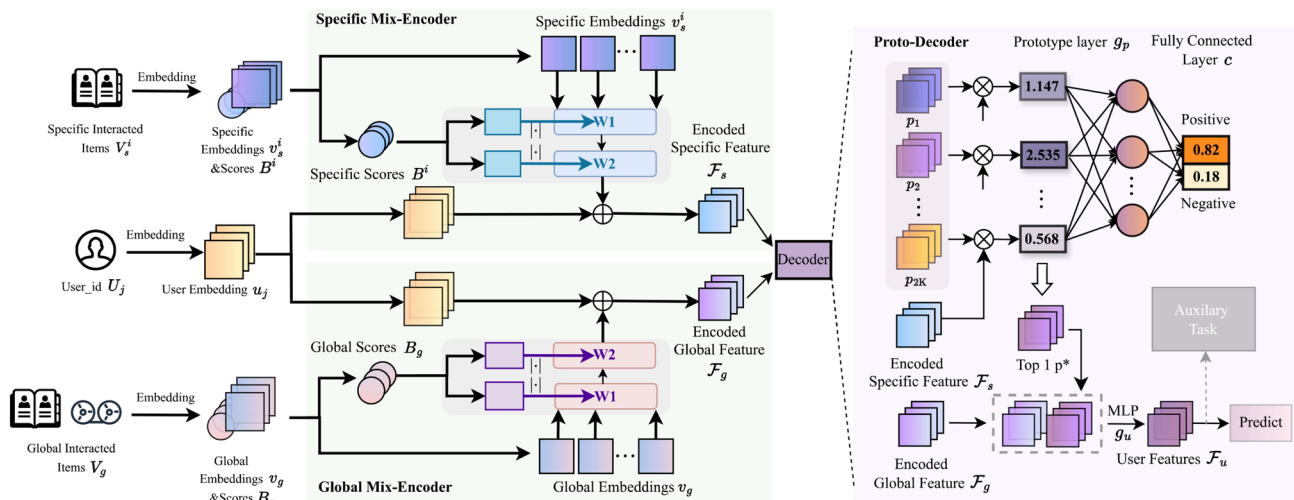


Fig. 2 Illustration of the proposed method PPA. Mix-encoders are employed for extracting the effective specific representation while proto-decoder perceives user preference explicitly and avoids disturbances caused by redundant item features

similarity prototype offers tailed recommendation support to the encoded global features and makes predictions.

4.1 Mix-Encoder

The goal of the mix-encoder is to extract the effective user representations with the guidance of item scores. Unlike previous simple multiplication, we generate the learnable weights through scores to guide the item embedding.

The potential user u_j has interacted in domain X and remains the histories corresponding to specific item embedding $v_s^X \in \mathbb{R}^{N \times h}$ and scores $B^X \in \mathbb{R}^{N \times 1}$, where N is the sample number. To capture the domain representation, we design the mix network to compress item embedding and scores. As depicted in the left column of Fig. 2, we use mix network $f(\cdot; \psi) : \mathbb{R}^{N \times h} \rightarrow \mathbb{R}^{1 \times h}$ parameterized by ψ to encode item embedding following the general mix mechanism as:

$$\begin{aligned} W_1, W_2 &= [|W_{\theta_1}, W_{\theta_2}] \cdot B|, \\ E &= W_2^T \cdot \text{ReLU}(W_1^T \cdot v), \\ \mathcal{F} &= E + u_j, \end{aligned} \quad (2)$$

where W_{θ_1} and W_{θ_2} represent learnable parameters. $|\cdot|$ takes the absolute value. The mix-weights $W_1 \in \mathbb{R}^{N \times h}$ and $W_2 \in \mathbb{R}^{h \times 1}$ are learned by input scores B . After running Eq. 2, the final encoded domain features $\mathcal{F} \in \mathbb{R}^{1 \times h}$ is generated by the domain representation E and current user embedding u_j . Here we adopt two mix-encoder branches to capture two encoded domain features.

In the specific branch, we feed inputs $[u_j, (v_s^X, B^X)]$ to the specific mix-encoder, focusing on capturing latent specific features. The mix-encoder generates specific encoded features as:

$$\mathcal{F}_s = f([u_j, (v_s^X, B^X)]; \psi_s) \in \mathbb{R}^{1 \times h}. \quad (3)$$

In the global branch, we feed inputs $[u_j, (v_g, B_g)]$ to the mix-encoder to generate global encoded features as:

$$\mathcal{F}_g = f([u_j, (v_g, B_g)]; \psi_g) \in \mathbb{R}^{1 \times h}. \quad (4)$$

Here we employ the mix-encoder to learn both specific and global domain representations. The mix-network can capture deeper dependencies between items, further encoding them into highly correlated specific domain features and global domain features. These encoded features $[\mathcal{F}_s, \mathcal{F}_g]$ are further employed by the decoder.

In conclusion, we improve the traditional user-item inner product in three aspects:

- First, the item is replaced with a weight proxy directly mapped from the item itself. The absolute value operation ensures the monotonicity between the item and its weight proxy—meaning the larger the value of the item's own representation, the larger the weight proxy. The weight proxy serves the subsequent nonlinear operations.
- Second, the inner product operation is replaced with a nonlinear function guided by the weight proxy. This modification enhances the nonlinear fitting capability between users and items, allowing the model to capture more complex interaction patterns that linear operations (e.g., inner product) fail to represent.
- Third, the subsequent feature F uniformly denotes the fused weight of user features and item features after interaction. Centered on user representations, this fused weight ensures that sufficient information is retained to support the subsequent modeling of "user

preferences"—laying a foundation for accurately capturing user-specific preference tendencies in downstream recommendation tasks.

4.2 Prototype-Aware Decoder

Existing methods suffer from specific feature-focused recommendations that are easily suboptimal or even obsolete when performing CDR under redundant features from the source domain. The recommendation should be separated from irrelevant item features in the source domain, further focusing on potential user preferences in the global domain. Therefore, we use prototypes to represent preferences, rather than directly using item features. At the heart of the prototype-aware decoder are quantitative prototypes of user interests to target items, which enables the decoder to explicitly perceive user preference.

We partitioned positive preferences into K adaptable prototypes and negative ones into K prototypes, resulting in $2K$ prototypes to precisely quantify the preference from dislike to like. We denote these prototypes as $\mathcal{P} = \{p_1, p_2, \dots, p_{2K} \mid p_i \in \mathbb{R}^{1 \times h}\}$. User preferences are extracted from the source domain, so they only interact with $\mathcal{F}_s$. We match the highest similarity prototype p^* to make PPA aware of the preference, following the calculation of similarity:

$$\text{sim}(p_i, \mathcal{F}_s) = \log \left(\frac{\|p_i - \mathcal{F}_s\|_2^2 + 1}{\|p_i - \mathcal{F}_s\|_2^2 + \epsilon} \right), \quad (5)$$

where ϵ is set as a small value to prevent division by zero. After obtaining the preference prototype p^* , we guide the global encoding features $\mathcal{F}_g$ by introducing this preference of the source domain into the shared domain, effectively extracting final user features $\mathcal{F}_u$ in the target domain. The linear layer g_u with parameters ϕ is employed to dynamically adapt encoded features to evolving preference features, extracting the final user features $\mathcal{F}_u = g_u(\mathcal{F}_g, p^*; \phi) \in \mathbb{R}^{1 \times h}$. The $\mathcal{F}_u$ is obtained through learning that combines the global features containing target domain information and the quantified user preferences. The objective function of prototype learning $\mathcal{L}_{pl}$ is defined as:

$$\mathcal{L}_{pl} = \mathcal{L}_{ce}(c \circ g_p \circ \mathcal{F}_s, \ell) + \lambda_1 \mathcal{L}_{clst} + \lambda_2 \mathcal{L}_{sep} + \lambda_3 \mathcal{L}_{div}, \quad (6)$$

where c is a fully connected layer that predicts positive or negative probability, and g_p is the prototype layer. $\mathcal{L}_{ce}$ represents the cross-entropy loss for preference classification within the prototype branch. ℓ is an all-one array when inputting normal interacted items. $\lambda_1, \lambda_2, \lambda_3$ control the weights of losses and we follow the general value in classification tasks [28].

As shown in Eq. (6), several constraints are proposed to construct the final prototypes [28]. The cluster cost $\mathcal{L}_{clst}$ motivates these interacted items from the source domain to exhibit proximity to one prototype corresponding to their preference. The separation cost $\mathcal{L}_{sep}$ promotes the distancing of encoded item features from prototypes that do not belong to their preference.

$$\begin{aligned} \mathcal{L}_{clst} &= \frac{1}{n} \sum_{i=1}^{2K} \min_{j: p_j \in \mathcal{P}_{y_i}} \|\mathcal{F}_s - p_j\|_2^2, \\ \mathcal{L}_{sep} &= -\frac{1}{n} \sum_{i=1}^{2K} \min_{j: p_j \notin \mathcal{P}_{y_i}} \|\mathcal{F}_s - p_j\|_2^2. \end{aligned} \quad (7)$$

where $\mathcal{P}_{y_i}$ is the set of prototypes under y_i class. Moreover, the diversity loss $\mathcal{L}_{div}$ in Eq. (8) encourages the diversity of the learned prototypes via penalizing prototypes too close.

$$\mathcal{L}_{div} = \sum_{k=1}^2 \sum_{\substack{i \neq j \\ p_i, p_j \in \mathcal{P}_k}} \max(0, \cos(p_i, p_j) - \xi), \quad (8)$$

where $k = \{1, 2\}$ denotes 2 kinds of preferences. ξ is the threshold of the cosine similarity in the diversity loss.

Finally, the user-item prediction score is calculated by $S_p = \mathcal{F}_u \cdot v^i$ to decide which item should be recommended. We exploit the BCE loss to make a basic prediction loss following previous work [1]: $\mathcal{L}_{pred} = \sum_{(u,v) \in \mathcal{E}} [-\log S_p - \log(1 - S_n)]$, where $S_n = \mathcal{F}_u \cdot \bar{v}^i$ denotes the negative prediction scores by dot product uninterested items $\bar{v}^i$.

4.3 Auxiliary Task

Inspired by previous work [1, 29] that applied contrastive learning (CL) [30] in enhancing the model, we select CL as the auxiliary task algorithm to regular the latent space of positive and negative interaction. Suppose on the domain X , the positive input is the same as specific items V_s^X on the main branch. The negative input is randomly sampled from uninterested items $\bar{V}_s^X$. After the main branch, we get the positive features $\mathcal{F}_p = \text{PPA}(V_s^X)$ and negative features $\mathcal{F}_n = \text{PPA}(\bar{V}_s^X)$. The auxiliary loss is as follows:

$$\mathcal{L}_{cl}^X = \sum_{u \in \mathcal{U}^X} [\mathcal{L}_{ce}(\mathcal{F}_p, Z_p) + \mathcal{L}_{ce}(\mathcal{F}_n, Z_n)], \quad (9)$$

where $\mathcal{L}_{ce}$ represents the cross-entropy loss for classification. Z_p is the positive label and stands for 1. Z_n is the

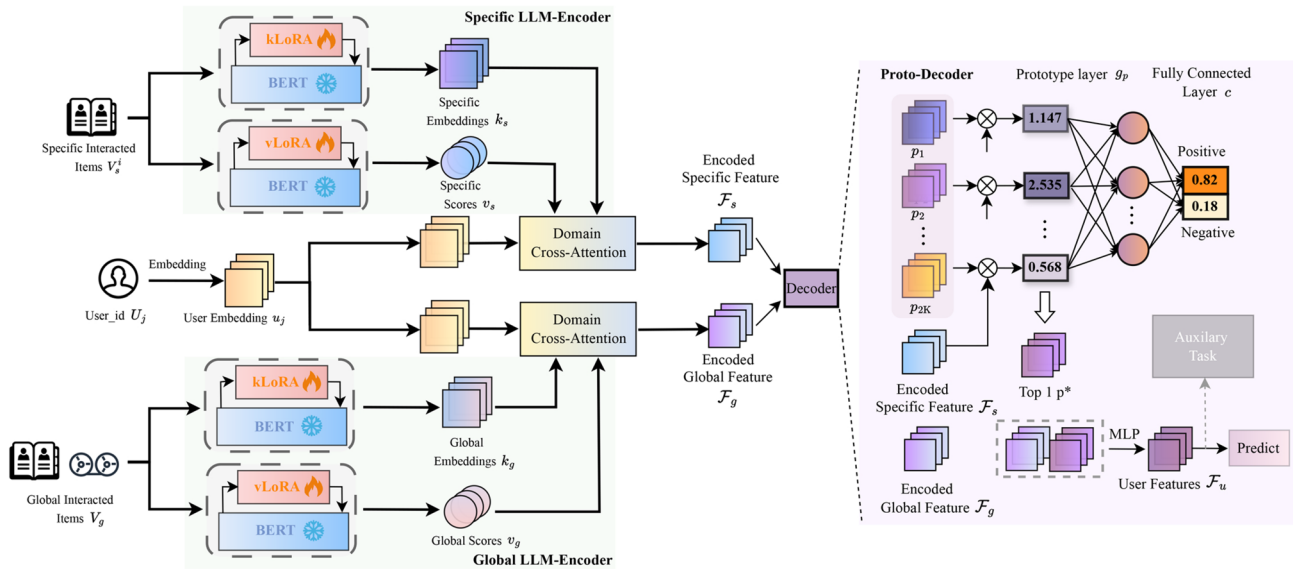


Fig. 3 Illustration of the proposed method PPA++. LLM-encoders (LoRA-based extractors) are proposed to extract more accurate semantic representations based on their substantial pre-training pro-

negative label and stands for 0. Our whole model can be optimized by the total objective function $\mathcal{L}_{tot}$ as:

$$\mathcal{L}_{tot} = \mathcal{L}_{pred} + \lambda_{pl}\mathcal{L}_{pl} + \mathcal{L}_{cl}. \quad (10)$$

5 PPA++: PPA with LLM-Encoder

In this section, we propose a new LLM-based encoder to better adapt to challenging scenarios with different user-item distributions. The extractor architecture adopts LLMs for their superior semantic parsing capacity, specifically leveraging their text-content prior knowledge to facilitate structured knowledge extraction from unstructured words. To address the imbalance distribution between user and items, we introduce a domain cross-attention module that dynamically calibrates the attention entropy between user-space embeddings and item-space embeddings. The improved model is dubbed PPA++.

5.1 LoRA-Based Extractor

Each LoRA-based extractor is utilized to gain reliable embeddings and scores from the original item words. The input is processed by a finetuned LoRA with the frozen BERT model, as the LoRA provides fewer resource consumption and lower accuracy loss.

This extractor is specifically designed to extract embeddings and scores separately, resulting in the derivation of k and v , which correspond to v_s^i and B^i as defined in the problem statement respectively. k stands for the total set of item

spaces. Domain cross-attention mechanism can interact different inputs together effectively, thus gaining an output that contains accurate and rich interaction information

spaces, and v stands for item importance. They correspond to the meaning of K and V of the attention mechanism. Also, the user embedding u_j corresponds to the meaning of Q of the attention mechanism. k and v can be calculated as:

$$k = kLoRA(V_s^X; \theta_{BERT}^{Frozen}, B_k A_k^T) \quad (11)$$

$$v = vLoRA(V_s^X; \theta_{BERT}^{Frozen}, B_v A_v^T) \quad (12)$$

As depicted in the left column of Fig. 3, we use network $f^*(\cdot; \psi^*) : \mathbb{R}^{N \times h} \rightarrow \mathbb{R}^{1 \times h}$ parameterized by ψ^* to gain item features following the general domain cross-attention mechanism. This domain cross-attention mechanism will be introduced in the next subsection.

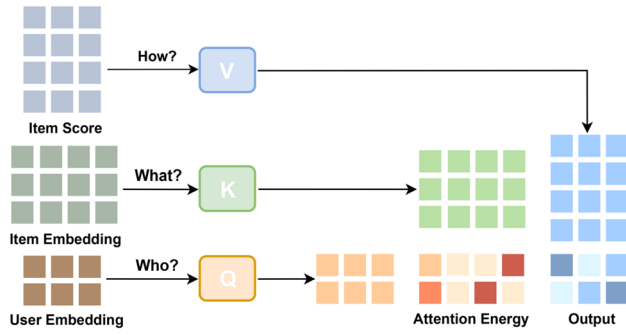
In the specific branch, we feed inputs $[u_j, k_s, v_s]$ to the specific LLM-encoder, focusing on extract accurate semantic features. The LLM-encoder generates specific encoded features as:

$$\mathcal{F}_s = f^*([u_j, k_s, v_s; \psi_s^*) \in \mathbb{R}^{1 \times h}. \quad (13)$$

In the global branch, we feed inputs $[u_j, k_g, v_g]$ to the LLM-encoder to generate global encoded features as:

$$\mathcal{F}_g = f^*([u_j, k_g, v_g; \psi_g^*) \in \mathbb{R}^{1 \times h}. \quad (14)$$

Here we employ the LoRA-based extractor to learn both specific and global domain representations. The LoRA-based extractor can better mine the underlying semantic embeddings under scenarios with low item categories.



5.2 Domain Cross-Attention Module

Previous direct multiplication operations are too simplistic, and when there is a large distribution difference in the number of users and items. The influence of the less numerous side in the embedding space is substantially unstable. Thus, we propose the domain cross-attention mechanism to rebalance the weight distribution, especially in situations of data imbalance. The mechanism of domain cross-attention can be calculated as follows:

$$Q, K, V = [W_u, \theta^{klora}, \theta^{vlora}] \cdot [u_j, k, v] \quad (15)$$

$$\mathcal{F} = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}} \cdot V\right) \quad (16)$$

where W_u , θ^{klora} and θ^{vlora} represent learnable parameters. d_k is a scaling factor used to adjust the scale of attention scores. After running Eq. (2), the LLM-encoder network can generate the final encoded feature as $\mathcal{F}$.

The domain cross-attention mechanism is illustrated in Fig. 4. "Who?" refers to determining the users, "What?" refers to determining the corresponding set of items, and "How?" denotes the assignment of interest scores to each item. This mechanism refines interactions between inputs by dynamically adjusting user attention towards items based on their relevance. It allows the model to accurately capture user preferences by weighing interactions between user history and item features. This dynamic adjustment ensures the model can adapt to shifting interests, improving recommendation precision and maintaining relevance in complex or changing environments.

Here we discuss a special case. When $|V|$ is extremely large, the computational complexity of the original attention mechanism, $O(|U| \cdot |V| \cdot d)$ becomes unacceptable. We propose a Top- k sparse approximation method: first retrieve $k \ll |V|$ candidate entries via ANN (Approximate Nearest Neighbor search), and then perform attention computation over these candidates. Experiments show that when $k=1000$, the decrease in HR/NDCG is less than 0.3%, while the memory usage is reduced to 1/100 of the original.

Require: UID u ;

- 1: Source domain items $V^s = \{v_i^s\}_{i=1}^N$ with scores $B^s \in \mathbb{R}^N$;
- 2: Global items $V^g = \{v_j^g\}_{j=1}^{2N}$ with scores $B^g \in \mathbb{R}^{2N}$;
- 3: Prototype matrix $P \in \mathbb{R}^{2K \times d}$;
- 4: Mask token [MASK] (for LLM encoder)

Ensure: Prediction score vector $\mathbf{s} \in \mathbb{R}^{|V^g|}$ over global items

- 5: **1. LoRA-based Extractor (§5.1)**
- 6: **for** each item $x \in V^s \cup V^g$ **do**
- 7: $\mathbf{k}_x, \mathbf{v}_x \leftarrow \text{LoRA}_{\theta_k, \theta_v}(\text{BERT}_{\text{frozen}}(\text{text}(x)))$
- 8: **end for**
- 9: Stack to obtain K^s, V^s and K^g, V^g
- 10: **2. Domain Cross-Attention (§5.2)**
- 11: $\mathbf{q}_u \leftarrow \text{UserEmbedding}(u) \in \mathbb{R}^d$
- 12: $\mathbf{F}^s \leftarrow \text{CrossAttn}(\mathbf{q}_u, K^s, V^s)$
- 13: $\mathbf{F}^g \leftarrow \text{CrossAttn}(\mathbf{q}_u, K^g, V^g)$
- 14: **3. Prototype-Aware Decoder (§4.2)**
- 15: $\mathbf{p}^* \leftarrow \arg \max_{p \in P} \text{sim}(p, \mathbf{F}^s)$
- 16: $\mathbf{F}_u \leftarrow g_\phi(\mathbf{F}^g, \mathbf{p}^*) \in \mathbb{R}^d$
- 17: **4. Scoring & Loss (§4.2-4.3)**
- 18: **for** $j = 1 \dots |V^g|$ **do**
- 19: $s_j \leftarrow \mathbf{F}_u \cdot \mathbf{k}_j^g$
- 20: **end for**
- 21: $\mathcal{L}_{\text{pred}} \leftarrow \text{BCE}(s_j, y_j)$
- 22: $\mathcal{L}_{\text{cl}} \leftarrow \text{InfoNCE}(\mathbf{F}_u, \mathbf{F}^+, \mathbf{F}^-)$
- 23: $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{pred}} + \lambda_{\text{pl}} \mathcal{L}_{\text{pl}} + \mathcal{L}_{\text{cl}}$
- return** $\mathbf{s} = [s_1, \dots, s_{|V^g|}]$

Algorithm 1 PPA++: Forward Pass Pseudocode

6 Experiments

To illustrate the effectiveness of the proposed PPA and PPA++ method, we aim to answer the following questions: (1) Can PPA/PPA++ outperform the state-of-the-art CDR methods in various scenarios? (Sect. 6.2) (2) How do the different components of PPA/PPA++ contribute to the overall performance? (Sect. 6.3) (3) How do different hyperparameters affect model performance? (Sect. 6.4.1) (4) Can PPA/PPA++ offer prototypes with understandable visualization? (Sect. 6.5)

6.1 Experimental Settings

Tasks and Datasets. We consider four different challenging scenarios under the Amazon benchmark [31]. Specifically, for the dual-intra-domain recommendation in **Scenario**

Table 3 Statistics of four CDR scenarios

Scenarios	Datasets	$ \mathcal{U} $	$ \mathcal{V} $	Training	Test
Scenario 1	Sport	9928	30796	92612	8326
	Cloth	9928	39008	87829	7540
	Elec	3325	17709	50407	2559
	Phone	3325	38706	115554	2560
Scenario 2	Sport	27328	12655	163291	3546
	Cloth	41829	17943	187880	3085
	Game	25025	12319	155036	1304
	Video	19457	8751	156091	1458
Scenario 3	M1	7109	2198	48302	3558
	M2	2697	1357	19615	1310
	M3	3328	1245	23367	1678
	M4	5482	2917	41226	2727
	M5	6466	9762	77173	3154
Scenario 4	D1	231444	2096	491098	13437
	D2	507715	595	1068490	35985
	D3	773188	1312	3785720	92672

1, we employ the Sport&Cloth and Elec&Phone datasets preprocessed by DisenCDR [7]. For the dual-inter-domain recommendation in **Scenario 2**, we utilize the Sport&Cloth and Game&Video datasets preprocessed by CDRIB [32]. For the multi-intra-domain recommendation in **Scenario 3**, we use the dataset preprocessed by M³rec [14], which includes interaction data from five anonymous countries in the electronics domain. For the multi-inter-domain recommendation in **Scenario 4**, we employ the financial dataset created by UniCDR [1], which consists of real-world data from three online banking platforms. The detailed statistics of the datasets in these four scenarios are summarized in Table 3.

Evaluation Metrics. We utilize Hit Ratio (HR) and Normalized Discounted Cumulative Gain (NDCG) to qualitatively evaluate performance, following previous works [1, 14]. HR focuses on whether the recommended list contains items that the user is interested in, while NDCG considers both the user's interest and the relevance ranking of the items in the recommended list. These two metrics are crucial for evaluating recommendation performance. Specifically, we first compute 1000 scores for each validation/test case (including 999 negative items and 1 positive item). Then, we rank the item list and evaluate the Top-10 items using HR and NDCG metrics.

Compared Methods. We compare our methods with the following baselines from four different aspects of CDRs.

- *Single-domain baselines.* We compare our proposed method with several single-domain baselines, including BPRMF [33], NeuMF [34], CML [35], and EASER [24]. These methods are widely used in industrial fields due to their stable performance. We also compare our method with recent graph-based methods: Random Walk[36],

NGCF [37], and LightGCN [38]. These methods mainly utilize high-order neighbor information for both users and items.

- *Intra-domain CDR baselines.* We compare our proposed method with CoNet [39] and DDTCDR [40], which achieve bi-directional transfer through cross-connections. We also compare our method with recent GNN-based models: PPGN [41], Bi-TGCF [8], and DisenCDR [7], which utilize multiple types of neighboring information. Additionally, there have also been some representative efforts in multi-target CDR. GA-MTC-DR [42] and HeroGraph [13] effectively utilize multi-target data through graphs and attention mechanisms. FOREC [43] and M³rec [14] leverage auxiliary rich data. For multi-target CDR, multi-task learning techniques can also be considered as potential approaches. Therefore, our method is also compared with Cross-Stitch [44], MMoe [45], and STAR [46].
 - *Inter-domain CDR baselines.* We compare our proposed method with EMC-DR [9] and SSCDR [47], which aim to improve the utility of limited data. TMC-DR [48] and SA-VAE [49] achieve domain-specific personalized learning through meta-learning and variational autoencoders, respectively. CDRIB [32] employs the information bottleneck principle.
 - *Universal CDR baselines.* Universal methods are proposed to achieve CDR across various scenarios. UniCDR [1] proposes a unified multi-domain recommendation framework, which models different scenarios by combining domain-specific and domain-shared features. This approach aims to capture both the unique characteristics of each domain and the common patterns shared across domains. For fairness, we do not compare our method with models that require pre-training or fine-tuning [2], as these additional steps may introduce bias into the evaluation.
- Implementation Details.** Following previous works [1], we train our model for a total of 100 epochs. The learning rate is $1e^{-3}$ and our batch size is 1024. The hidden dimension is set to 128. All experiments are conducted using the same hardware and software configuration to ensure a fair comparison. We choose $\lambda_{pl} = 1.0$ and $K = 4$ for our PPA method to achieve the best performance. The prototype constraint weights $\lambda_1, \lambda_2, \lambda_3$ in $\mathcal{L}_{pl}$ are set to 0.1, 0.05 and 0.001, following the common setting for classification tasks [28].

6.2 Performance Comparisons

Dual-domain scenarios. We first compare PPA and PPA++ with other baselines under the dual-intra-domain scenario. Test results of the comparison are shown in Table 4, which

Table 4 Performance comparison (%) on dual-intra-domain recommendations. Bold denotes the best results

Datasets	Metric@10	Single-Domain Methods				Cross-Domain Methods				Universal Methods			
		BPRMF	NeuMF	NGCF	LightGCN	CoNet	DDTCDR	PPGN	Bi-TGCF	DisenCDR	UniCDR	PPA	PPA++
Sport	HR	10.43	10.74	13.13	13.19	12.09	11.86	15.10	14.83	17.55	18.37	20.72	21.52
	NDCG	5.41	5.46	6.87	6.94	6.41	6.37	8.03	7.95	9.46	10.98	14.25	15.50
Cloth	HR	11.53	11.18	13.22	13.58	12.40	12.54	14.23	14.68	16.31	17.85	19.92	20.84
	NDCG	6.25	6.02	6.97	7.29	6.62	7.13	7.68	7.93	9.03	11.20	13.89	14.72
Elec	HR	15.71	16.17	18.55	19.17	17.22	18.47	21.68	22.14	24.57	22.92	25.39	26.14
	NDCG	9.19	9.24	10.87	10.28	9.86	11.08	11.63	12.20	14.51	13.83	15.37	15.88
Phone	HR	16.32	15.84	22.79	23.25	17.66	17.23	24.54	25.71	28.76	24.72	31.30	32.74
	NDCG	8.53	8.02	12.38	12.72	9.30	8.58	13.34	13.93	16.13	13.77	17.86	18.69

showcases that the performance of PPA is superior to other baselines across all metrics on different datasets. In particular, on the sport and cloth datasets, our PPA outperforms state-of-the-art models by 3.27% and 2.69% under the NDCG metric. Compared with PPA, PPA++ brings about 0.92% accuracy improvement in average. In this simple scenario, results show that the disturbance caused by source-domain item features greatly deteriorates the perception of preferences in the target domain. We also conduct experiments on PPA in the dual-inter-domain scenario. As shown in Table 5, we find that PPA still outperforms other baselines. Especially on the Game dataset, we exceed the sota method by 2.57% and 1.3% under the HR and NDCG respectively. Moreover, PPA++ outperforms PPA by an average margin of 0.52%. These results highlight that our preference quantification is crucial in simple scenarios, which can outperform baselines by a large margin.

Multi-domain scenarios. To further evaluate our PPA's performance under complex scenarios, we conducted experiments on multi-domain scenarios. Test results for the item-overlapped scenario are shown in Table 6. On average, PPA++ surpasses PPA by 0.724% and it achieved the highest NDCG but ranked second in most HR metrics. The HR metric focuses on whether the recommended list contains items of interest, without considering their ranking or relevance. In contrast, the NDCG metric not only considers whether items appear in the recommended list but also evaluates their ranking and relevance. Therefore, the NDCG metric, with its stronger emphasis on preference weights, accurately reflects the effectiveness of our PPA model in preference learning. The M³rec [14] method, which is specifically designed for multi-target domains, obtained higher HR metrics due to its well-designed multi-domain feature mining. Test results for the user-overlapped scenario are shown in Table 7. PPA outperforms other models in both HR and NDCG. Particularly in the D1 dataset, PPA surpasses the state-of-the-art model by 3.47% and 1.7% in the HR and NDCG. Based on this, PPA++ outperforms PPA by an average margin of 1.02%. These results underscore PPA and PPA++ as universal CDR methods applicable across various scenarios, excelling not only in dual-domain but also in multi-domain tasks.

Universal Methods. We also conducted extensive comparisons with our universal baseline method (i.e., UniCDR [1]) under these four scenarios. In the dual-intra-domain scenario, PPA significantly outperforms UniCDR across all performance metrics on all datasets. Notably, our PPA surpasses UniCDR by 6.58% and 4.09% in HR and NDCG metrics, respectively, on the Phone dataset. In the dual-inter-domain scenario, PPA exceeds UniCDR by 2.57% and 1.3% in HR and NDCG on the Game dataset. These results demonstrate the excellence of PPA as a

Table 5 Performance comparison (%) on dual-inter-domain recommendations. Bold denotes the best results

Datasets	Metric@10	Single-Domain Methods				Cross-Domain Methods				Universal methods			
		CML	BPRMF	NGCF	EMCDR	SSCDR(CML)	TMCDR	SA-VAE	CDRIB	UniCDR	PPA	PPA++	
Sport	HR	5.82	5.75	7.22	7.44	7.27	7.18	7.51	12.04	11.20	13.81	14.73	
	NDCG	3.29	3.16	3.63	3.71	3.75	3.84	3.72	6.22	7.04	7.61	8.05	
Cloth	HR	6.97	6.75	7.07	7.29	6.12	8.11	7.21	12.19	12.48	12.87	13.15	
	NDCG	3.92	3.26	3.48	4.48	3.06	5.05	4.59	6.81	7.52	7.57	7.68	
Game	HR	2.82	3.77	5.14	4.63	3.48	5.36	5.84	8.51	8.78	11.35	12.27	
	NDCG	1.44	1.89	2.73	2.24	1.59	2.58	2.78	4.58	4.63	5.93	6.36	
Video	HR	3.07	4.46	7.41	7.94	5.51	8.85	7.46	13.17	10.74	12.63	13.42	
	NDCG	1.30	2.36	3.87	4.29	2.61	4.41	3.71	6.49	5.89	6.66	6.91	

universal method in simple dual-domain recommendations, benefiting from more effective user preferences and fewer disturbances from the source domain.

To further demonstrate the robustness of PPA in more complex scenarios, we also compared them in multi-domain recommendation scenarios. In the item-overlapped multi-intra-domain scenario, a larger number of users rely on the same item features. Consequently, using item features to represent user preferences can lead to significant errors. Our proposed PPA surpasses UniCDR on every performance metric across all datasets. Moreover, in the user-overlapped scenario, PPA still achieves superior performance results in most cases, especially on the D1 dataset, with improvements of 3.47% and 1.71% on HR and NDCG metrics, respectively.

All experimental results indicate that PPA exhibits excellent user preference awareness under various scenarios, proving the effectiveness of the mix-encoder and prototype-aware decoder. This unified end-to-end framework proposed by PPA demonstrates promising potential, showcasing its adaptability and generalization capabilities in universal recommendation tasks. Moreover, the performance of PPA++ consistently surpasses that of PPA across all aspects. Since PPA++ addresses the two issues we identified and demonstrates superior performance in extreme scenarios, it exhibits its stronger generalizability.

6.3 Ablation Study

6.3.1 Ablation Study of PPA

To analyze the impact of different components in PPA, we carry out ablation studies to quantitatively evaluate their contribution. The comparison results of various ablations on the dual-intra-domain dataset are shown in Table 8.

PPA w/o proto-decoder. To examine the effectiveness of the preference prototype, we remove prototypes and use a linear layer to predict based on encoded features. The results illustrate that the "w/o decoder" variant only obtained the lowest results. Although the encoder extracts effective domain representations, the performance deteriorates due to the redundant item disturbances, which can be removed by the proto-decoder.

PPA w/o mix-encoder. To assess the impact of the mix-network, we utilize the original dot product to replace it. The results show that the mix architecture has a positive effect on the representation extraction. Although simple multiplication operations can directly guide scores to items, the impact of scores is not entirely the same for different domains and categories. Our mix-network can better consider the prior differences between them, making the guidance of scores more comprehensive.

Table 6 Performance comparison (%) on item-overlapped multi-intra-domain recommendation. Bold denotes the best results

Datasets	Metric@10	Single-Domain Methods					Cross-Domain Methods					Universal Methods				
		NeuMF	LightGCN	Random Walk	EASER	Cross-Stitch	MMoE	Bi-TGCF	STAR	FOREC	M3Rec	UniCDR	PPA	PPA++		
M1	HR	62.73	64.73	64.66	70.80	64.46	65.73	66.86	62.93	65.06	73.13	69.08	70.81	71.59		
	NDCG	46.31	48.30	48.05	54.95	49.15	48.98	50.46	46.57	52.05	55.83	59.57	61.27	62.65		
M2	HR	55.60	52.13	50.20	57.40	54.06	56.26	53.46	54.89	58.42	60.86	58.01	58.76	59.34		
	NDCG	34.84	32.70	31.10	37.92	36.65	38.71	33.43	35.25	40.03	40.04	47.52	48.74	50.06		
M3	HR	60.40	56.26	57.53	63.60	59.46	61.53	58.73	60.80	64.13	66.53	64.60	66.27	66.83		
	NDCG	36.57	34.22	34.89	40.13	39.13	41.30	35.77	37.09	41.88	43.35	53.24	53.82	54.35		
M4	HR	40.33	41.14	39.20	45.13	38.93	38.60	42.26	40.20	41.60	48.46	47.52	48.61	49.38		
	NDCG	29.83	31.03	30.08	36.63	29.16	30.16	32.76	29.84	33.52	37.98	42.54	42.69	42.76		
M5	HR	12.26	17.13	17.73	19.13	17.06	16.66	17.86	16.60	17.46	22.66	19.78	20.92	21.73		
	NDCG	9.01	13.16	14.07	16.93	12.51	11.79	14.42	12.48	13.19	18.63	17.04	17.38	17.82		

6.3.2 Ablation Study of PPA++

To analyze the impact of the two newly proposed components introduced in PPA++, we conduct ablation studies to quantitatively assess their contribution. The comparison results of various ablations on Scenario 4 are presented in Table 9. We use Scenario 4 as the dataset for the ablation study because it aligns with the context of the two issues we previously proposed, as it both contains a relatively small number of items and exhibits a significant disparity between the number of items and users.

PPA with attention. To examine the effect of the domain cross-attention module in PPA++, we build upon the PPA model and add the domain cross-attention module. The experimental results show that the "with attention" variant achieved some improvement over the original performance of PPA, demonstrating the effectiveness of the domain cross-attention module in combining different inputs. It helps assign reasonable weights to inputs with varying quantities.

PPA with extractor. To evaluate the impact of the LoRA-based extractor, we conduct experiments by incorporating this module into the PPA framework. The experimental results demonstrate a more effective outcome compared to PPA, thereby validating the efficacy of using the LoRA-based extractor for semantic representation extraction. Particularly in Scenario 4 where the number of items is significantly smaller than that of users, the supplementary information provided by the pre-trained LLMs effectively enhances the accuracy of semantic representation extraction.

6.4 Sensitivity Analysis of Hyperparameters

6.4.1 Analysis of the Loss Factor and Prototype Number

In this section, we explore the hyperparameter influence of the loss factor λ_{pl} and the prototype number K on each class. The comparison results of various settings on the dual-intra-domain dataset are shown in Fig. 5 and Fig. 6, respectively.

Factor λ_{pl} . The total objective function is defined by Eq.(10), where the weight λ_{pl} assigned to $\mathcal{L}_{pl}$ balances the constraints in prototype learning. According to the trajectory of the data points in Fig. 5, we find that our model shows higher and more robust results when $\lambda_{pl} \in [0.8, 1.2]$. Setting the prototype constraint as equally important as the classification constraint is crucial for learning user preferences. We suggest that tuning this factor to around 1.0 is a reasonable way to achieve great performance. Moreover, we observe that under different settings of λ_{pl} , PPA++ outperforms PPA, demonstrating higher and more robust performance. PPA++ is also able to adapt well even when the weight of the prototype component is either too low or too high.

Table 7 Performance comparison (%) on user-overlapped multi-intra-domain recommendation. Bold denotes the best results

Datasets	Metric@10	Single-Domain Methods				Cross-Domain Methods				Universal Methods			
		BPRMF	NeuMF	EASER	LightGCN	MMoE	CoNet	Bi-TGCF	GA-MTCDR	HeroGraph	UniCDR	PPA	PPA++
D1	HR	19.48	20.57	9.15	25.52	21.22	20.60	26.98	26.13	29.73	32.60	36.07	37.52
	NDCG	7.66	7.17	4.04	10.60	8.82	8.46	10.64	10.02	11.74	13.56	15.26	16.73
D2	HR	50.45	52.92	50.07	56.18	56.22	53.53	60.48	59.59	61.49	64.37	62.91	64.68
	NDCG	33.50	35.73	28.53	37.09	38.63	37.66	47.19	47.67	49.57	50.48	50.78	51.27
D3	HR	64.87	64.53	50.40	67.13	65.71	65.90	72.88	73.32	71.77	73.89	74.08	74.41
	NDCG	47.69	48.44	29.02	40.49	47.08	47.51	54.15	57.00	56.81	59.15	59.61	60.23

Number K . The number of prototypes, denoted by K , plays a crucial role in shaping the preference space and capturing user preferences effectively. Figure 6 illustrates the impact of varying K on the performance. The trajectory of the data points reveals that both excessive sparsity and density of prototypes can be detrimental to forming an optimal preference space. When the number of prototypes is too small, the model's capacity to represent diverse user preferences becomes limited. Conversely, when the number of prototypes is excessively large, it can introduce repetition and redundancy, resulting in inefficient utilization. Moreover, an overly dense prototype space can make it difficult to impose meaningful constraints and regularization. Based on the empirical evidence presented in Fig. 6, we suggest choosing 4 prototypes for positive preferences and 4 for negative preferences as a reasonable starting point to achieve a balance between capturing a sufficient range of user preferences and maintaining a manageable prototype space. Furthermore, we observe that under varying numbers of prototypes, PPA++ demonstrates higher and more robust performance compared to PPA. It adapts well both when the number of prototypes is relatively large or small.

6.4.2 Sensitivity Analysis of LoRA Rank

Under Scenario 4 (Dataset D1), we fixed other hyperparameters ($\lambda_{pl} = 1.0$, $K = 4$) and varied the LoRA rank $r \in 4, 8, 16, 32, 64$ to observe changes in HR@10 and NDCG@10. The results are illustrated in Table 10. Through observation, we can conclude that the experiment performance nearly achieves at $r = 8$. Further increase leads to diminished returns, while values below $r < 8$ result in insufficient semantic extraction, causing a noticeable metric decline (HR drops by 0.6 pp). For $r > 8$, almost no improvement is observed, yet it introduces twice the parameters and much longer training time. After comprehensively considering the effectiveness-efficiency trade-off, we uniformly adopt $r=8$ for all experiments in the main text.

6.5 Visualization Analysis

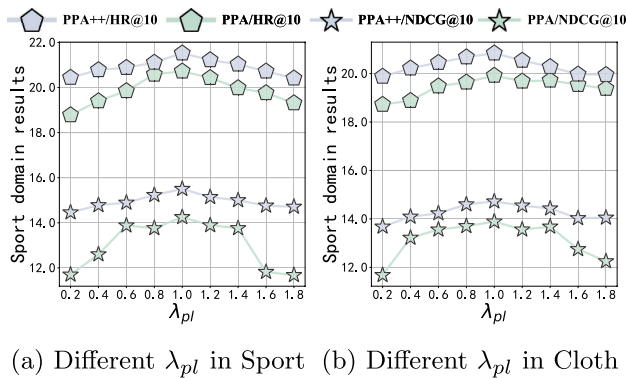
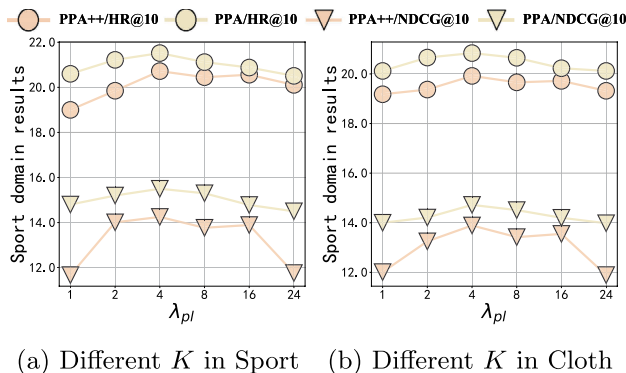
To further investigate the distribution of the prototypes learned by the proto-decoder in the prototype space, we conduct a qualitative analysis with PPA as a case study on dual-intra-domain and multi-intra-domain datasets. The preference prototypes in different scenarios are visualized in Fig. 7. We find that positive prototypes and corresponding negative prototypes can maintain their distance and exhibit similar trends in multi-domain situations, suggesting that the proposed method can successfully learn different sparse preferences for different domain classes, which significantly extracts user potential preferences from

Table 8 Test results of PPA and its ablations on Scenario1 and Scenario 2

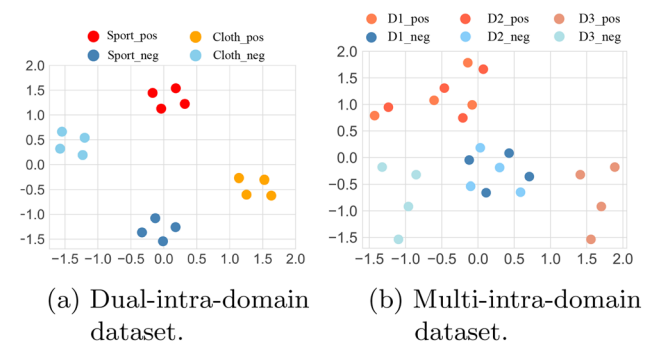
Scenario	Datasets	Metrics@10	w/o decoder	w/o encoder	PPA	PPA++
Scenario1	Sport	HR	19.21	20.11	20.72	21.52
		NDCG	11.49	13.27	14.25	15.50
	Cloth	HR	18.06	18.65	19.92	20.84
		NDCG	11.75	12.03	13.89	14.72
	Elec	HR	22.79	24.33	25.39	26.14
		NDCG	14.22	15.05	15.37	15.88
Scenario 2	Sport	HR	25.42	28.48	31.30	32.74
		NDCG	14.24	16.82	17.86	18.69
	Cloth	HR	11.37	13.16	13.81	14.73
		NDCG	6.71	7.17	7.61	8.05
	Game	HR	12.32	12.77	12.87	13.15
		NDCG	6.98	7.18	7.57	7.68
	Video	HR	9.78	9.79	11.35	12.27
		NDCG	4.82	4.92	5.93	6.36
			11.69	12.28	12.63	13.42
			6.47	6.34	6.66	6.91

Table 9 Test results of PPA++ and its ablations on scenario 4

Scenario	Datasets	Metrics@10	PPA	with attention	with extractor	PPA++
Scenario 4	D1	HR	36.07	36.38	37.32	37.52
		NDCG	15.26	15.49	16.24	16.73
	D2	HR	62.91	63.35	64.29	64.48
		NDCG	50.78	51.03	51.16	51.27
	D3	HR	74.08	74.12	74.33	74.41
		NDCG	59.61	59.95	60.17	60.23

**Fig. 5** Impact of the loss weight λ_{pl} **Fig. 6** Impact of the prototypes number K on each class**Table 10** Experiments under D1 datasets, Inf. denotes inference

Methods	Params	HR@10	NDCG@10	Inf.(ms)
PPA	73MB	36.07	15.26	7.8
PPA++ ($r=4$)	+12MB	37.15	15.71	9.4
PPA++ ($r=8$)	+23MB	37.52	16.73	9.6
PPA++ ($r=16$)	+46MB	37.48	16.85	9.6
PPA++ ($r=32$)	+92MB	37.41	16.79	9.8

**Fig. 7** Visualization of the learnable prototypes by t-SNE. We set the prototype number $K = 4$ in both scenarios

irrelevant source-domain features. By learning sparse representations, PPA is able to focus on the most informative and discriminative features, thereby reducing the impact of redundant or noisy information. Moreover, PPA also learns the relationship across various scenarios, which enables the

generalizable module to handle more multi-domain problems and improve its performance in various scenarios.

7 Conclusions

In this paper, we explore the challenge of discovering user preferences under the disturbance of redundant item features. Unlike previous methods that only focus on discovering user preferences under redundant specific item features, we first propose a Preference Prototype-Aware (PPA) method based on mix-encoder to learn user preferences quantitatively while minimizing this disturbance. Mix-encoder can extract general representations of interacted items with the guidance of item scores. Experimental results show that utilizing preference prototype-aware method can significantly improve the performance in all real-world scenarios considered in this study. Besides, LLMs with lora fine-tuning can extract better item semantic representations without too much computational resources while previous direct addition operations are too simple and not dominant when there is a large difference in the number of users and items. Based on LLMs and domain cross-attention mechanism, PPA++ achieves better performance than PPA and outperforms all the existing universal cross-domain RS methods. Moreover, PPA and PPA++ are unified end-to-end training frameworks and demonstrate the reliability of user preferences represented by prototypes across different scenarios.

In our future work, we will consider setting the number of prototypes as an additional learnable parameter in the learning pipeline to achieve a higher degree of automated learning. Additionally, we aim to explore the interpretability of the learned preference prototypes, enabling users to gain insights into their own preferences and the factors influencing them.

Acknowledgements This research is supported by the Natural Science Foundation of China (No. 62172372 and U23B2057, 62176185), Postgraduate Education and Teaching Reform Special Project (Outstanding Engineers) of Nanjing University of Aeronautics and Astronautics(2024YJXGG-Z12).

Author Contributions YZ and JZ: Methodology, Writing-original draft, Data analysis and Paper writing. FX: Methodology and Paper writing. LC: Data analysis and Paper writing. BL: Supervision, Project administration, Funding acquisition. NW and HT: Formal analysis, Investigation, Resource. LG and HY: Writing-review, editing.

Funding The Natural Science Foundation of China (No. 62172372 and U23B2057, 62176185), Postgraduate Education and Teaching Reform Special Project (Outstanding Engineers) of Nanjing University of Aeronautics and Astronautics(2024YJXGG-Z12).

Data Availability The data and material that support the findings of this study are available from the corresponding author upon reasonable request.

Declarations

Conflict of interest The authors have no competing interests to declare that are relevant to the content of this article.

Ethics approval and consent to participate Approve.

Consent for publication Approve.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

References

1. Cao J, Li S, Yu B, Guo X, Liu T, Wang B (2023) Towards universal cross-domain recommendation. In: ACM Int Conf Web Search and Data Mining, pp 78–86
2. Hao B, Yang C, Guo L, Yu J, Yin H (2024) Motif-based prompt learning for universal cross-domain recommendation. In: ACM Int Conf Web Search and Data Mining, pp 257–265
3. Ju H, Kang S, Lee D, Hwang J, Jang S, Yu H (2024) Multi-domain recommendation to attract users via domain preference modeling. AAAI Conf Art Intell 38(8):8582–8590
4. Zhang Y, Zhang J, Xu F, Chen L, Li B, Guo L, Yin H (2024) Preference prototype-aware learning for universal cross-domain recommendation. In: ACM Int Conf Inf Know Manage pp 3290–3299
5. Yufang L, Shaoqing W, Keke L, Xueting L, Fuzhen S (2024) Channel-enhanced contrastive cross-domain sequential recommendation. Data Sci Eng 9(3):325–340
6. Liu Y, Wang S, Li X, Sun F (2024) A meta-adversarial framework for cross-domain cold-start recommendation. Data Sci Eng 9(2):238–249
7. Cao J, Lin X, Cong X, Ya J, Liu T, Wang B (2022) Disentangled representations for cross-domain recommendation: an embedding and mapping approach. Int Joint Conf Art Intell 17:2464–2470
8. Liu M, Li J, Li G, Pan P (2020) Cross domain recommendation via bi-directional transfer graph collaborative filtering networks. In: ACM Int Conf Inf Knowl Manage, pp 885–894
9. Man T, Shen H, Jin X, Cheng X (2017) Cross-domain recommendation: an embedding and mapping approach. Int Joint Conf Art Intell 17:2464–2470
10. Guo L, Lu Z, Yu J, Nguyen QVH, Yin H (2024) Prompt-enhanced federated content representation learning for cross-domain recommendation. Proc ACM Web Conf 2024:3139–3149
11. Liu H, Guo L, Zhu L, Jiang Y, Gao M, Yin H (2024) Merpl: a pretrain, prompt, and fine-tune paradigm for non-overlapping many-to-one cross-domain recommendation. ACM Trans Inf Syst 42(4):1–24

12. Guo L, Zhang J, Tang L, Chen T, Zhu L, Yin H (2022) Time interval-enhanced graph neural network for shared-account cross-domain sequential recommendation. *IEEE Trans Neural Netw Learn Syst* 35(3):4002–4016
13. Cui Q, Wei T, Zhang Y, Zhang Q (2020) Herograph: A heterogeneous graph framework for multi-target cross-domain recommendation. In: *ACM Conf Recomm Syst*
14. Cao J, Cong X, Liu T, Wang B (2022) Item similarity mining for multi-market recommendation. In: *Int ACM SIGIR Conf Res Devel Inf Retr*, pp 2249–2254
15. Chen C, Li O, Tao D, Barnett A, Rudin C, Su JK (2019) This looks like that: deep learning for interpretable image recognition. *Conf Neural Inf Proc Syst* 32
16. Nauta M, Jutte A, Provoost J, Seifert C (2021) This looks like that, because... explaining prototypes for interpretable image recognition. In: *Joint European Conf Mach Lear Knowl Disc Datab*, Springer pp 441–456
17. Xu W, Xian Y, Wang J, Schiele B, Akata Z (2020) Attribute prototype network for zero-shot learning. *Adv Neural Inf Process Syst* 33:21969–21980
18. Sankar A, Wang J, Krishnan A, Sundaram H (2021) Protocf: Prototypical collaborative filtering for few-shot recommendation. In: *ACM Conf Recomm Syst*, pp 166–175
19. Melchiorre AB, Rekabsaz N, Ganhör C, Schedl M (2022) Protomf: Prototype-based matrix factorization for effective and explainable recommendations. In: *ACM Conf Recomm Syst*, pp 246–256
20. Gan C, Huang B, Hu B, Ma J, Zhang Z, Zhou J, Zhang G, Zhong W (2024) Peace: Prototype learning augmented transferable framework for cross-domain recommendation. In: *ACM Int Conf on Web Search and Data Mining*, pp 228–237
21. Wu X, Magnani A, Chaidaroon S, Puthenpuhussery A, Liao C, Fang Y (2022) A multi-task learning framework for product ranking with bert. In: *ACM Web Conf*, pp 493–501
22. Kang W-C, Ni J, Mehta N, Sathiamoorthy M, Hong L, Chi E, Cheng DZ (2023) Do llms understand user preferences? evaluating llms on user rating prediction. *arXiv preprint [arXiv:2305.06474](https://arxiv.org/abs/2305.06474)*
23. Petruzzelli A, Musto C, Laraspatla L, Rinaldi I, Gemmis M, Lops P, Semeraro G (2024) Instructing and prompting large language models for explainable cross-domain recommendations. In: *ACM Conf Recomm Syst*, pp 298–308
24. Steck H (2019) Embarrassingly shallow autoencoders for sparse data. In: *World Wide Web*, pp. 3251–3257
25. Hu EJ, Shen Y, Wallis P, Allen-Zhu Z, Li Y, Wang S, Wang L, Chen W (2021) Lora: Low-rank adaptation of large language models. *arXiv preprint [arXiv:2106.09685](https://arxiv.org/abs/2106.09685)*
26. Devlin J (2018) Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint [arXiv:1810.04805](https://arxiv.org/abs/1810.04805)*
27. Radford A (2018) Improving language understanding by generative pre-training
28. Zhang Z, Liu Q, Wang H, Lu C, Lee C (2022) Protgnn: towards self-explaining graph neural networks. *Proceedings of the AAAI Conference on Artificial Intelligence* 36:9127–9135
29. Li C, Xie Y, Yu C, Hu B, Li Z, Shu G, Qie X, Niu D (2023) One for all, all for one: Learning and transferring user embeddings for cross-domain recommendation. In: *ACM Int Conf on Web Search and Data Mining*, pp 366–374
30. Chen T, Kornblith S, Norouzi M, Hinton G (2020) A simple framework for contrastive learning of visual representations. In: *Int Conf Mach Lear*, pp 1597–1607. PMLR
31. He R, McAuley J (2016) Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In: *International World Wide Web Conf*, pp 507–517
32. Cao J, Sheng J, Cong X, Liu T, Wang B (2022) Cross-domain recommendation to cold-start users via variational information bottleneck. In: *IEEE Int Conf Data Eng*, pp 2209–2223. IEEE
33. Rendle S, Freudenthaler C, Gantner Z, Schmidt-Thieme L (2012) Bpr: Bayesian personalized ranking from implicit feedback. *arXiv preprint [arXiv:1205.2618](https://arxiv.org/abs/1205.2618)*
34. He X, Liao L, Zhang H, Nie L, Hu X, Chua T-S (2017) Neural collaborative filtering. In: *World Wide Web*, pp 173–182
35. Hsieh C-K, Yang L, Cui Y, Lin T-Y, Belongie S, Estrin D (2017) Collaborative metric learning. In: *World Wide Web*, pp 193–201
36. Barkan O, Koenigstein N (2016) Item2vec: neural item embedding for collaborative filtering. In: *IEEE Int Work Mac Lear Sig Proc (MLSP)*, pp 1–6. IEEE
37. Wang X, He X, Wang M, Feng F, Chua T-S (2019) Neural graph collaborative filtering. In: *Int ACM SIGIR Conf Res Devel Nform Retr*, pp 165–174
38. He X, Deng K, Wang X, Li Y, Zhang Y, Wang M (2020) Lightgcn: Simplifying and powering graph convolution network for recommendation. In: *Int ACM SIGIR Conf Res Devel Inf Retr*, pp 639–648
39. Hu G, Zhang Y, Yang Q (2018) Conet: Collaborative cross networks for cross-domain recommendation. In: *ACM Int Conf Inf Knowl Manage*, pp 667–676
40. Li P, Tuzhilin A (2020) Dtdcdr: Deep dual transfer cross domain recommendation. In: *ACM Int Conf Web Search and Data Mining*, pp 331–339
41. Zhao C, Li C, Fu C (2019) Cross-domain recommendation via preference propagation graphnet. In: *ACM Int Conf Inf Knowl Manage*, pp 2165–2168
42. Zhu F, Wang Y, Zhou J, Chen C, Li L, Liu G (2021) A unified framework for cross-domain and cross-system recommendations. *IEEE Trans Knowl Data Eng*
43. Bonab H, Aliannejadi M, Vardasbi A, Kanoulas E, Allan J (2021) Cross-market product recommendation. In: *ACM Int Conf Inf Knowl Manage*, pp 110–119
44. Misra I, Shrivastava A, Gupta A, Hebert M (2016) Cross-stitch networks for multi-task learning. In: *IEEE/CVF Comput Vision and Pattern Recognit Conf*, pp 3994–4003
45. Ma J, Zhao Z, Yi X, Chen Y, Hong L, Chi EH (2018) Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In: *ACM SIGKDD Conf Knowl Disc and Data Mining*, pp 1930–1939
46. Sheng X-R, Zhao L, Zhou G, Ding X, Dai B, Luo Q, Yang S, Lv J, Zhang C, Deng H, et al (2021) One model to serve all: Star topology adaptive recommender for multi-domain ctr prediction. In: *ACM Int Conf Inf Knowl Manage*, pp 4104–4113
47. Kang S, Hwang J, Lee D, Yu H (2019) Semi-supervised learning for cross-domain recommendation to cold-start users. In: *ACM Int Conf Inf Knowl Manage*, pp 1563–1572
48. Zhu Y, Ge K, Zhuang F, Xie R, Xi D, Zhang X, Lin L, He Q (2021) Transfer-meta framework for cross-domain recommendation to cold-start users. In: *Int ACM SIGIR Conf Res Devel Inf Retr*, pp 1813–1817
49. Salah A, Tran TB, Lauw H (2021) Towards source-aligned variational models for cross-domain recommendation. In: *ACM Conf Recomm Syst*, pp 176–186