Mixed Attention Network for Cross-domain Sequential Recommendation

Guanyu Lin
Carnegie Mellon University
Pittsburgh, PA, United States
guanyul@andrew.cmu.edu

Chen Gao†
Tsinghua University
Beijing, China
chgao96@gmail.com

Yu Zheng
Tsinghua University
Beijing, China

Jianxin Chang
Kuaishou Technology
Beijing, China

Yanan Niu
Kuaishou Technology
Beijing, China

Yang Song
Kuaishou Technology
Beijing, China

Kun Gai
Unaffiliated
China

Zhiheng Li
Tsinghua University
Beijing, China

Depeng Jin
Tsinghua University
Beijing, China

Yong Li
Tsinghua University
Beijing, China

Meng Wang
Hefei University of Technology
Hefei, China

ABSTRACT

In modern recommender systems, sequential recommendation leverages chronological user behaviors to make effective next-item suggestions, which suffers from data sparsity issues, especially for new users. One promising line of work is the cross-domain recommendation, which trains models with data across multiple domains to improve the performance in data-scarce domains. Recent proposed cross-domain sequential recommendation models such as PiNet and DASL have a common drawback relying heavily on overlapped users in different domains, which limits their usage in practical recommender systems. In this paper, we propose a **M**ixed **A**ttention **N**etwork (MAN) with local and global attention modules to extract the domain-specific and cross-domain information. Firstly, we propose a local/global encoding layer to capture the domain-specific/cross-domain sequential pattern. Then we propose a mixed attention layer with item similarity attention, sequence-fusion attention, and group-prototype attention to capture the local/global item similarity, fuse the local/global item sequence, and extract the user groups across different domains, respectively. Finally, we propose a local/global prediction layer to further evolve and combine the domain-specific and cross-domain interests. Experimental results on two real-world datasets (each with two domains) demonstrate the superiority of our proposed model. Further study also illustrates that our proposed method and components are model-agnostic and effective, respectively. The code and data are available at https://github.com/tsinghua-fib-lab/WSDM24-MAN.

CCS CONCEPTS

• **Information systems** → **Personalization**.

KEYWORDS

Cross-domain Sequential Recommendation, Mixed Attention Network, Recommender Systems

1 INTRODUCTION

Widespread in online platforms such as news, video, e-commerce, etc., recommender systems that vastly improve the efficiency of information distribution and diffusion are of great importance in today's Web. Sequential recommendation [38] is one of the most important research problems in recommender systems, which aims at predicting a user's next interacted item based on their historical interaction sequence. Though recent representative models of sequential recommendation such as GRU4REC [11], SASRec [15] and SURGE [2] etc. have achieved decent performance, they suffer from the issue of data sparsity [7, 40, 43], limiting the performance.

To address the data sparsity issue, cross-domain recommendation [6, 32, 44] is a widely adopted approach, which leverages the data from multiple domains to boost the performance of the data-scarce domain by parameter-sharing [6] or multi-task learning [27]. Particularly, a few early attempts [3, 19, 29] were proposed to achieve *cross-domain sequential recommendation*, which leverages cross-domain technique to address the data sparsity of sequential modeling. However, as illustrated in Figure 1(a), these methods rely heavily on the overlapped users and require pairwise inputs from two domains of the same bridge users, which is hardly satisfied
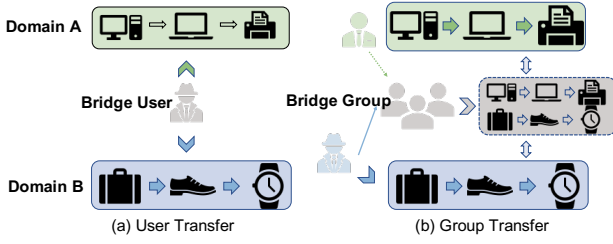
Figure 1: Illustration of (a) user transfer learning relies on overlapped users and (b) group transfer learning without previous assumptions on user overlap.

in practical scenarios. For example, in our experimental benchmark datasets (Micro Video and Amazon), there is only a small part (at most 8.37%) of overlapped users, as Table 1, which violates the assumption of existing approaches. In fact, in many real-world applications, users are not overlapping across different domains [21, 24]. Thus, it is challenging for the existing cross-domain sequential recommendation to work in real-world scenarios. Actually, there are three key challenges for cross-domain sequential recommendation:

- **Different item characteristics across domains.** There are not always overlapped items across different domains. Even if items are shared across domains, items reflect different characteristics. For example, for a higher-end e-commerce website, the price aspect takes less effect when users purchase items, while it plays an important role in a lower-end website. Such difference brings difficulty in learning accurate item representations across different domains.
- **Various sequential patterns across domains.** Similar to the item, the sequential behaviors vary in different domains. For example, users may be more decisive in a higher-end E-commerce website, leading to very short sequences with very brief sequential patterns. Therefore sequential patterns are various, and the modeling is challenging.
- **User preference transferring without overlapped user.** We focus on the general cross-domain recommendation task, where users may not fully overlap. Therefore, it is challenging to capture the common preference shared by users across domains, especially when there is even no overlapped user.

To address these challenges, we develop a novel group-based method with the group transfer to avoid dependence on the overlap of users and global space to capture the item characteristics and sequential patterns across different domains as Figure 1(b). Note that the group-prototype attention here can capture group information in an unsupervised manner, without further requiring additional input information compared with Figure 1(a). Specifically, we propose a novel solution named MAN (short for **M**ixed **A**ttention **N**etwork for Cross-domain Sequential Recommendation), consisting of local and global modules, mixing three types of designed attention network from item level, sequence level, and group level. First, we generate separate representations for each item, including the local representation capturing domain-specific characteristics and the global representation shared by different domains. We then design an item similarity attention module to capture the similarity between local/global item representation and the target item representation. Second, we propose a sequence-fusion attention module

to fuse the local and global item sequential representations. Most importantly, although user information cannot be directly shared, the group information can be shared across domains. Therefore, we propose a group-prototype attention module, which utilizes multiple group prototypes to transfer the information at the group level. Finally, the obtained local and global embeddings are fed into the corresponding prediction layers to evolve the domain-specific and cross-domain interests.

The contributions of this paper can be summarized as follows.

- We approach the problem of cross-domain sequential recommendation from a more practical perspective that there is no prior assumption of overlapped users across domains, which is far more challenging.
- We propose a solution named MAN and address the key challenges by mixing three attention modules: item similarity attention, sequence-fusion attention, and group-prototype attention. Besides, local and global designs are proposed to capture the domain-specific and cross-domain patterns.
- We conduct extensive experiments on a collected large-scale industrial dataset and a public benchmark dataset, where the results show significant performance improvements compared with the state-of-the-art models. Further studies illustrate that our proposed method is model-agnostic, and group prototypes can capture the group patterns across domains without overlapping users.

2 PROBLEM FORMULATION

In our problem of cross-domain sequential recommendation, we first use A and B to denote the two domains. Let $\mathcal{I}^A$ and $\mathcal{I}^B$ denote the sets of items in domain A and B, respectively. More specifically, supposing $i_t^A \in \mathcal{I}^A$ or $i_t^B \in \mathcal{I}^B$ is the t-th item that a given user has interacted with in the A or B domain, the t-length sequence of historical items can be represented as $(i_1^A, i_2^A, \ldots, i_t^A)$ or $(i_1^B, i_2^B, \ldots, i_t^B)$. The goal of our problem is to improve the recommendation accuracy of the following item *i.e.*, i_{t+1}^A or i_{t+1}^B, of all users across all domains simultaneously. The problem can be formulated as follows.
Input: Item sequence $(i_1^A, i_2^A, \ldots, i_t^A)$ and $(i_1^B, i_2^B, \ldots, i_t^B)$ for users in domain A and B, respectively.
Output: The cross-domain recommendation model estimating the probability that target item i_{t+1}^A and i_{t+1}^B will be interacted by the given users with item sequence $(i_1^A, i_2^A, \ldots, i_t^A)$ and $(i_1^B, i_2^B, \ldots, i_t^B)$ in the domain A and B, respectively.

3 METHODOLOGY

Figure 2 illustrates our proposed MAN model, encoding the item sequence with local/global encoding layer, mixing three attention modules, and evolving the interests by local/global prediction layer.

- **Local/Global Encoding Layer**. We build both domain-specific local and cross-domain global embeddings for items. We further encode them with local and global encoders, respectively, to capture the domain-specific and cross-domain item sequential patterns.
- **Mixed Attention Layer**. We propose Item Similarity Attention, Sequence-fusion Attention, and Group-prototype Attention to capture the cross-domain patterns at the item, sequence, and group levels.
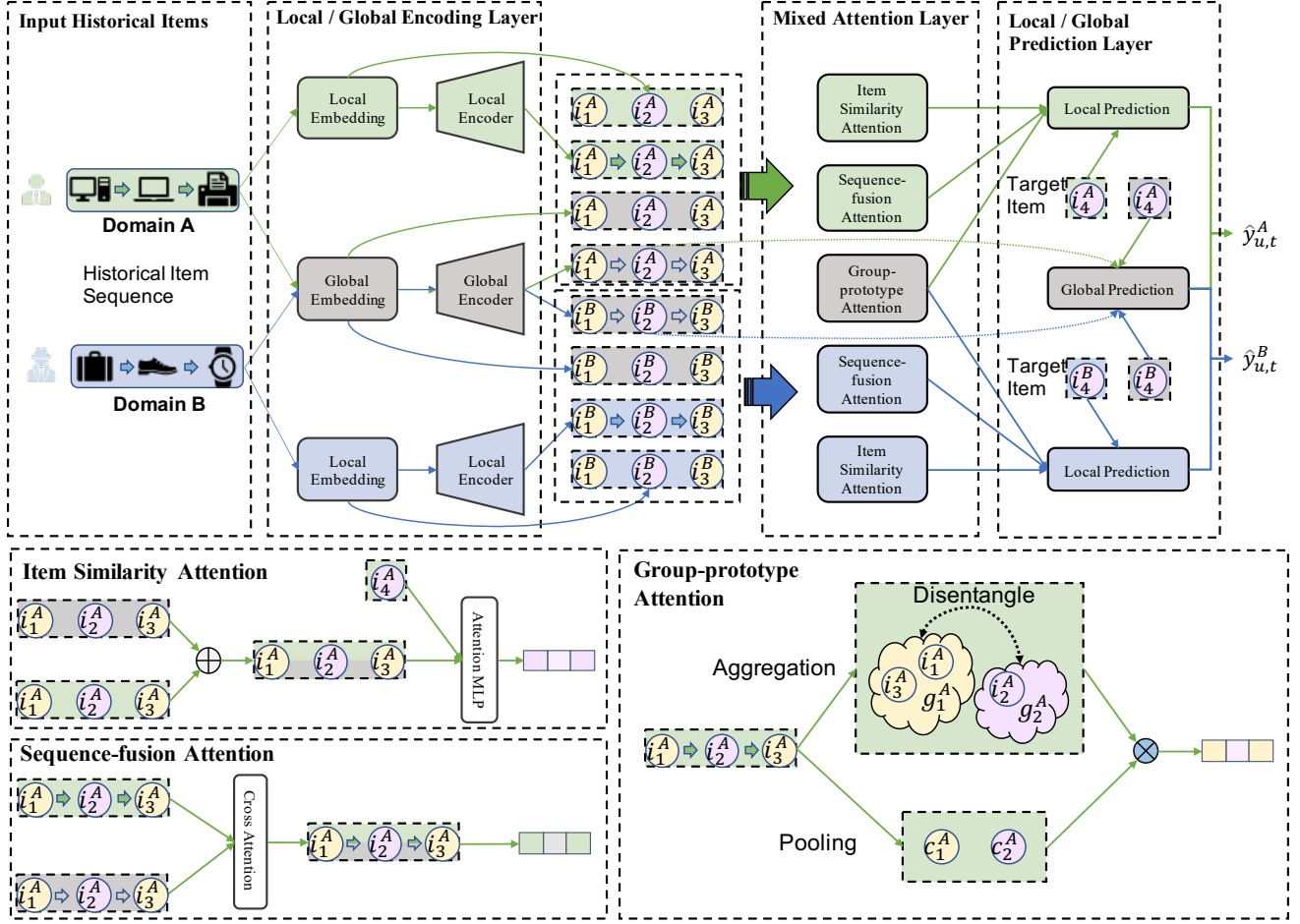
Figure 2: Illustration of our proposed MAN model. (1) The item sequences are first input into the Local/Global Encoding Layer, which builds local and global embeddings for each item and encodes them to extract the local and global sequential patterns; (2) In the Mixed Attention Layer, Item Similarity Attention is fed with local and global item embeddings to capture the item-level relation; Sequence-fusion Attention fuses the encoded local and global sequential representations to capture the sequence-level relation; Group-prototypes attention leverages the shared group prototypes to capture the group-level relation. Here we take domain A to illustrate each proposed attention component in detail. (4) The aggregated embeddings will be fed into the local prediction layer and global prediction layer, respectively, for the final prediction.

- **Local/Global Prediction layer**. To evolve the interests and predict the probability of the candidate's next item that the user will interact with in each domain, we propose a local prediction layer and a global prediction layer.

3.1 Local/Global Encoding Layer

We first build local and global item embeddings. Then we further look up item embeddings and encode them with local and global encoders at the sequence level.

*3.1.1 **Local and Global Item Embeddings**.* To capture the domain-specific patterns for different domains, we create two item embedding matrices $\mathbf{M}^A \in \mathbb{R}^{|\mathcal{I}^A| \times D}$ and $\mathbf{M}^B \in \mathbb{R}^{|\mathcal{I}^B| \times D}$ where D denotes the latent dimensionality. Then, to capture the shared item

characteristics across different domains, from the perspective of representation learning, we assume there exists a shared latent space [39] where different domains have common representation; Thus, we create a shared embedding matrix $\mathbf{M} \in \mathbb{R}^{|\mathcal{I}^A \cup \mathcal{I}^B| \times D'}$.

Here we use $(i_1^A, i_2^A, \ldots, i_t^A)$ and $(i_1^B, i_2^B, \ldots, i_t^B)$, to denote the historical item sequences of domain A and domain B, specifically. Note that we pad sequences shorter than T with a constant zero vector, following existing works [2, 15]. To further capture the position of items in the sequence, we also integrate learnable *positional embeddings* into item embeddings (domain A example) as:

$$\begin{aligned}
\mathbf{E}^A &= \left[\mathbf{M}_{i_1^A}^A, \mathbf{M}_{i_2^A}^A, \cdots, \mathbf{M}_{i_n^A}^A\right] + \left[\mathbf{P}_1^A, \mathbf{P}_2^A, \cdots, \mathbf{P}_n^A\right], \\
\mathbf{E}^{A_g} &= \left[\mathbf{M}_{i_1^A}, \mathbf{M}_{i_2^A}, \cdots, \mathbf{M}_{i_n^A}\right] + \left[\mathbf{P}_1, \mathbf{P}_2, \cdots, \mathbf{P}_n\right],
\end{aligned} \quad (1)$$

$$\mathbf{E}^B = \left[\mathbf{M}_{i_1^B}^B, \mathbf{M}_{i_2^B}^B, \cdots, \mathbf{M}_{i_n^B}^B\right] + \left[\mathbf{P}_1^B, \mathbf{P}_2^B, \cdots, \mathbf{P}_n^B\right],$$
$$\mathbf{E}^{Bg} = \left[\mathbf{M}_{i_1^B}, \mathbf{M}_{i_2^B}, \cdots, \mathbf{M}_{i_n^B}\right] + \left[\mathbf{P}_1, \mathbf{P}_2, \cdots, \mathbf{P}_n\right], \quad (2)$$

where $\mathbf{E}^A, \mathbf{E}^B \in \mathbb{R}^{T \times D}$ ($\mathbf{E}^{Ag}, \mathbf{E}^{Bg} \in \mathbb{R}^{T \times D'}$) denote the local (global) embeddings for domain A and domain B, respectively. Besides, g means global. Here $\mathbf{P}^A, \mathbf{P}^B \in \mathbb{R}^{T \times D}$ and $\mathbf{P} \in \mathbb{R}^{T \times D'}$ are the learnable positional embeddings.

*3.1.2 **Local Encoder and Global Encoder of Sequences**. After obtaining $\mathbf{E}^A, \mathbf{E}^B, \mathbf{E}^{Ag}$ and $\mathbf{E}^{Bg}$ from the embedding layers, we then apply sequential encoders to learn the sequential patterns. Here we propose the local encoder and global encoder as follows,

$$S^A = \mathbf{Encoder}(\mathbf{E}^A), S^B = \mathbf{Encoder}(\mathbf{E}^B), \quad (3)$$

$$S^{Ag} = \mathbf{Encoder}_g(\mathbf{E}^{Ag}), S^{Bg} = \mathbf{Encoder}_g(\mathbf{E}^{Bg}), \quad (4)$$

where $\mathbf{Encoder}$ and $\mathbf{Encoder}_g$ are the sequential backbone models (i.e., SASRec [15] or SURGE [2]) with independent and shared parameters, respectively, across domains.

Based on it, we obtain S^A (S^B) and S^{Ag} (S^{Bg}), which capture local sequential patterns and global sequential patterns, respectively, in domain A (B).

3.2 Mixed Attention Layer

In this section, we first propose item similarity attention to extract similar items from local and global spaces. Then we propose sequence-fusion attention to further fuse the local and global item sequence representations, which will combine the domain-specific and cross-domain sequential patterns. Finally, we propose group-prototype attention to extract the group pattern across domains.

*3.2.1 **Item Similarity Attention**. To capture the similarity between local/global item embeddings and target item embedding, we first fuse the item embedding from local space (i.e., $\mathbf{E}_j^A$ and $\mathbf{E}_j^B$) and global space (i.e., $\mathbf{E}_j^{Ag}$ and $\mathbf{E}_j^{Bg}$) together. Specifically, given a user in domain A (B), we can calculate the item similarity scores $\mathbf{F}^A$ ($\mathbf{F}^B$) between his/her historical items and the target item as follows,

$$\mathbf{F}^A = \mathrm{MLP}\left(\mathbf{M}_{i_{t+1}^A} \| \mathbf{E}^A + \mathbf{E}^{Ag}\right), \mathbf{F}^B = \mathrm{MLP}\left(\mathbf{M}_{i_{t+1}^B} \| \mathbf{E}^B + \mathbf{E}^{Bg}\right), \quad (5)$$

where $\mathbf{M}_{i_{t+1}^A}$ ($\mathbf{M}_{i_{t+1}^B}$) denotes the embedding of the target item for domain A (B) and $\|$ denotes the concatenation operation. Based on the item similarity scores, we can then weigh similar historical items' embeddings to refine item embeddings as follows,

$$\mathbf{E}^{A_i} = \mathbf{softmax}\left(\mathbf{F}^A\right)(\mathbf{E}^A + \mathbf{E}^{Ag}), \mathbf{E}^{B_i} = \mathbf{softmax}\left(\mathbf{F}^B\right)(\mathbf{E}^B + \mathbf{E}^{Bg}), \quad (6)$$

where $\mathbf{E}^{A_i}$ and $\mathbf{E}^{B_i} \in \mathbb{R}^{T \times D}$ are the representations of target items' similar historical items weighted by similarity scores of $\mathbf{F}^A$ and $\mathbf{F}^B$ in domain A and domain B, respectively. Here A_i and B_i mean item similarity of domain A and B, respectively.

Note that all functions are with dependent parameters except that they are subscript with g w.r.t. *global*.

*3.2.2 **Sequence-fusion Attention**. After obtaining S^A, S^B, S^{Ag}, and S^{Bg}, we then fuse them to combine the domain-specific and cross-domain sequential patterns together as follows,

$$S^{A_s} = \mathbf{MLP}(\mathbf{CA}(S^A, S^{Ag}) + S^A); S^{B_s} = \mathbf{MLP}(\mathbf{CA}(S^B, S^{Bg}) + S^B); \quad (7)$$

where the **cross-attention (CA)** layer [37] is defined as follows (take S^A as an example),

$$\mathbf{CA}(S^A, S^{Ag}) = \mathbf{Atten}\left(S^A \mathbf{W}_{A_s}^Q, S^{Ag} \mathbf{W}_{A_s}^K, S^{Ag} \mathbf{W}_{A_s}^V\right), \quad (8)$$

where $\mathbf{W}_{A_s}^Q, \mathbf{W}_{A_s}^K, \mathbf{W}_{A_s}^V \in \mathbb{R}^{D \times D}$ are parameters to be learned and **Atten** function is defined as below.

$$\mathbf{Atten}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \mathbf{softmax}\left(\frac{\mathbf{Q}\,\mathbf{K}^T}{\sqrt{D}}\right)\mathbf{V}, \quad (9)$$

where $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ are the query matrix, key matrix, and value matrix, respectively.

*3.2.3 **Group-prototype Attention**. Although we can not leverage overlapped user IDs across domains, there often exist user groups with similar preferences. Specifically, we first pool each sequence to obtain relevance to each group. Then we leverage multiple group prototypes to aggregate the item groups and weigh them based on their relevance.

Group Interest Pooling. For an item sequence of a user, it actually does not belong to only one group prototype. Instead, it can be a hybrid combination of several prototypes with different weights. For example, a user can be both an adolescent and a basketball lover at the same time. Thus, we propose a learnable soft cluster assignment matrix [33, 41], to calculate the importance of N_g groups. Specifically, the item sequence of each user is firstly pooled by a pooling matrix $\mathbf{W}_A^P \in \mathbb{R}^{N_g \times T}$ ($\mathbf{W}_B^P \in \mathbb{R}^{N_g \times T}$), based on which the relevance of the user to each group can be calculated as follows,

$$\mathbf{C}^A = \mathbf{MLP}\left(\mathbf{W}_A^P S^A\right), \mathbf{C}^B = \mathbf{MLP}\left(\mathbf{W}_B^P S^B\right), \quad (10)$$

where $\mathbf{C}^A$ and $\mathbf{C}^B \in \mathbb{R}^{N_g \times 1}$ are relevance scores for each group.

Group Interest Aggregation. We then create N_g group-prototype embeddings $\mathbf{G} \in \mathbb{R}^{N_g \times D}$ to represent the interest groups. These embeddings can then be transformed to each domain, aggregating the typically related items as follows,

$$\mathbf{G}^A = \mathbf{MLP}\left(\mathbf{CA}(\mathbf{G}, S^A)\right), \mathbf{G}^B = \mathbf{MLP}\left(\mathbf{CA}(\mathbf{G}, S^B)\right), \quad (11)$$

where $\mathbf{G}^A$ and $\mathbf{G}^B \in \mathbb{R}^{N_g \times D}$ are the obtained group-prototype representations for the sequences of domain A and domain B, respectively. Here **CA** layer is similar to Eqn.(8). We can then weigh all group-prototype representations based on the relevance scores as follows,

$$\mathbf{G}^{A_u} = \mathbf{softmax}\left(\mathbf{C}^A\right)\mathbf{G}^A, \mathbf{G}^{B_u} = \mathbf{softmax}\left(\mathbf{C}^B\right)\mathbf{G}^B, \quad (12)$$

where $\mathbf{G}^{A_u}$ and $\mathbf{G}^{B_u} \in \mathbb{R}^{N_g \times D}$ are the weighted group-prototype representations for each user.

Group-prototype Disentanglement. Each group prototype obviously should be distinct, according to its definition. Therefore, inspired by the advances of disentangled representation learning [28], we propose the prototype disentanglement regularization as:

$$\mathcal{L}^g = -\lambda_g \sum_{i=1}^{N_g} \sum_{j=i+1}^{N_g} \left(\mathbf{G}_i - \mathbf{G}_j \right)^2 \tag{13}$$

where λ_g is the penalty hyper-parameter. This loss function will be jointly learned with the main loss function later.

3.3 Local/Global Prediction Layer

In this section, we first evolve the local and global interests via corresponding prediction layers. Then we optimize them with the objective function for each domain.

*3.3.1 **Local and Global Prediction Layer**.* With the proposed mixed-attention network (item similarity attention, sequence-fusion attention, and group-prototype attention), we concatenate the outputs together and feed them into the proposed local prediction layer and global prediction layer based on MLP [2, 46, 47], which can be formulated as follows,

$$\hat{y}_{u,t}^A = \mathbf{MLP} \left(\mathbf{e}^{A_i} \| \mathbf{s}^{A_s} \| \mathbf{g}^{A_u} \| \mathbf{s}^A \| \mathbf{M}_{i_{t+1}^A}^A \right) + \mathbf{MLP}_g \left(\mathbf{s}^{A_g} \| \mathbf{M}_{i_{t+1}^A} \right), \tag{14}$$

$$\hat{y}_{u,t}^B = \mathbf{MLP} \left(\mathbf{e}^{B_i} \| \mathbf{s}^{B_s} \| \mathbf{g}^{B_u} \| \mathbf{s}^B \| \mathbf{M}_{i_{t+1}^B}^B \right) + \mathbf{MLP}_g \left(\mathbf{s}^{B_g} \| \mathbf{M}_{i_{t+1}^B} \right), \tag{15}$$

where $\mathbf{MLP}_g$ is the MLP layer with shared parameters across domains, and the concatenated embeddings are obtained via,

$$\mathbf{e}^{A_i} = \sum_{t=1}^T \mathbf{E}_t^{A_i}, \mathbf{s}^{A_s} = \sum_{t=1}^T \mathbf{S}_t^{A_s}, \mathbf{g}^{A_u} = \sum_{k=1}^{N_g} \mathbf{G}_k^{A_u}, \mathbf{s}^A = \sum_{t=1}^T \mathbf{S}_t^A, \mathbf{s}^{A_g} = \sum_{t=1}^T \mathbf{S}_t^{A_g},$$

$$\mathbf{e}^{B_i} = \sum_{t=1}^T \mathbf{E}_t^{B_i}, \mathbf{s}^{B_s} = \sum_{t=1}^T \mathbf{S}_t^{B_s}, \mathbf{g}^{B_u} = \sum_{k=1}^{N_g} \mathbf{G}_k^{B_u}, \mathbf{s}^B = \sum_{t=1}^T \mathbf{S}_t^B, \mathbf{s}^{B_g} = \sum_{t=1}^T \mathbf{S}_t^{B_g},$$

which denotes average pooling before being fed into MLPs.

*3.3.2 **Objective Function with Independent Updating**.* We then exploit the negative log-likelihood function [2, 46, 47] for optimization, which can be formulated as follows,

$$\mathcal{L}^A = -\frac{1}{|\mathcal{R}^A|} \sum_{(u,i_t^A) \in \mathcal{R}^A} \left(y_{u,t}^A \log \hat{y}_{u,t}^A + \left(1 - y_{u,t}^A\right) \log \left(1 - \hat{y}_{u,t}^A\right) \right), \tag{16}$$

$$\mathcal{L}^B = -\frac{1}{|\mathcal{R}^B|} \sum_{(u,i_t^B) \in \mathcal{R}^B} \left(y_{u,t}^B \log \hat{y}_{u,t}^B + \left(1 - y_{u,t}^B\right) \log \left(1 - \hat{y}_{u,t}^B\right) \right), \tag{17}$$

where $\mathcal{R}^A$ and $\mathcal{R}^B$ are the training sets of domain A and domain B, respectively. Here $y_{u,t}^A = 1$ ($y_{u,t}^B = 1$) and $y_{u,t}^A = 0$ ($y_{u,t}^B = 0$) indicate a positive sample and a negative sample, respectively, and $\hat{y}_{u,t}^A$ and $\hat{y}_{u,t}^B$ stand for predicted click probability of the next item.

To optimize jointly across two domains, the final objective function is a linear combination of $\mathcal{L}^A$, $\mathcal{L}^B$ and $\mathcal{L}^g$ calculated in Eqn.(13), Eqn.(16) and Eqn.(17), respectively, as follows,

$$\mathcal{L} = \mathcal{L}^A + \mathcal{L}^B + \lambda^A \|\Theta^A\|_2 + \lambda^B \|\Theta^B\|_2 + \mathcal{L}^g \tag{18}$$

where Θ^A and Θ^B are the sets of learnable parameters with λ^A and λ^B as the regularization penalty hyper-parameters of domain A and domain B, respectively.

Table 1: Data statistic for two datasets. Here Avg. Length is the average number of users' history interacted items.

Dataset	Micro Video		Amazon	
Domain	**A**	**B**	**Video Games**	**Toys**
#Users	43,919	37,692	826,767	1,342,911
#Items	147,813	131,732	50,210	327,698
#Records	18,011,737	14,908,625	1,324,753	2,252,771
Overlap Items	71.22%	79.91%	7.66%	4.72%
Overlap users	7.18%	8.37%	0.27%	0.04%
Ave. length	212.50	244.95	19.55	18.23
Density	0.2775%	0.3003%	0.0032%	0.0005%

4 EXPERIMENTS

In this section, we conduct extensive experiments with two real-world datasets, investigating the following research questions (RQs).

- **RQ1:** How does the proposed method perform compared with the state-of-the-art single-domain recommenders and cross-domain recommenders?
- **RQ2:** What is the effect of different components in the method?
- **RQ3:** Is the proposed method model-agnostic? What about the performance on different backbones? Is the method still effective with the solely local or global module?
- **RQ4:** How do the group prototypes represent different groups?

4.1 Experimental Setup

*4.1.1 **Datasets**.* We evaluate the recommendation performance on an industrial Micro Video dataset and a public e-commerce dataset. The statistics of the datasets are shown in Table 1.

*4.1.2 **Baselines and Evaluation Metrics**.* To demonstrate the effectiveness of our model, we compare it with two categories of competitive baselines: single-domain models and cross-domain models. Specifically, single-domain models are DIN [47] Caser [36], GRU4REC [11], DIEN [46], SASRec [15], SLi-Rec [42] and SURGE [2]. These single-domain models are trained on each domain independently following existing work [19, 29].

Besides, cross-domain models are NATR [5], PiNet [29] and DASL [19]. PiNet and DASL are adapted to our settings without fully-overlapped users (with the item sequence of another domain as empty). Other cross-domain models like MiNet [30] and CoNet [13] are not included in experiments because they are non-sequential models and will be much poor than sequential models [29].

All models are evaluated on two popular accuracy metrics AUC and GAUC [9], as well as two ranking metrics, MRR and NDCG [2].

*4.1.3 **Hyper-parameter Settings**.* The initial learning rate for Adam [16] is 0.001 with Xavier initialization [8] to initialize the parameters. Regularization coefficients are searched in [$1e^{-7}$, $1e^{-5}$, $1e^{-3}$]. The batch size is set as 200 and 20, respectively, for the Micro Video dataset and Amazon dataset. The embedding sizes of all models with 40 and 20 are fixed for the Micro Video dataset and Amazon dataset, respectively. MLPs with layer size [100, 64] and [20, 10] are exploited for the prediction layer on the Micro Video dataset and Amazon dataset, respectively. Item sequence length of 250 is set for the Micro Video dataset, and 20 is set for the Amazon

Table 2: Performance comparisons for MAN on Micro Video dataset and Amazon dataset.

Domain	Metric	Single-domain							Cross-domain			
		DIN	Caser	GRU4REC	DIEN	SASRec	SLi-Rec	SURGE	NATR	PiNet	DASL	Ours
Micro Video A	AUC	0.5673	0.7744	0.7838	0.6666	0.7730	0.7558	0.7959	<u>0.7972</u>	0.7834	0.7879	**0.8285**
	MRR	0.5544	0.5740	0.5417	0.5264	0.5359	0.5337	0.5888	<u>0.5899</u>	0.5273	0.5568	**0.6167**
	NDCG	0.6628	0.6780	0.6531	0.6409	0.6488	0.6461	0.6892	<u>0.6921</u>	0.6422	0.6651	**0.7112**
	WAUC	0.7837	0.8053	0.7910	0.7654	0.7911	0.7729	0.8170	<u>0.8197</u>	0.7880	0.8075	**0.8435**
Micro Video B	AUC	0.5613	0.7308	0.7625	0.6581	<u>0.7794</u>	0.7620	0.7605	0.7727	0.7595	0.7665	**0.8094**
	MRR	0.4526	0.4971	0.5285	0.4768	<u>0.5472</u>	0.5418	0.5042	0.5462	0.5037	0.5288	**0.5756**
	NDCG	0.5843	0.6184	0.6431	0.6025	<u>0.6574</u>	0.6529	0.6239	0.6571	0.6240	0.6431	**0.6797**
	WAUC	0.7246	0.7533	0.7860	0.7420	<u>0.7957</u>	0.7845	0.7645	0.7939	0.7705	0.7858	**0.8215**
Domain	Metric	Single-domain							Cross-domain			
		DIN	Caser	GRU4REC	DIEN	SASRec	SLi-Rec	SURGE	NATR	PiNet	DASL	Ours
Amazon Video Games	AUC	0.5577	0.5766	0.5303	<u>0.6059</u>	0.5234	0.5750	0.5975	0.5617	0.5740	0.5527	**0.6559**
	MRR	0.3736	0.3284	0.2953	0.3526	0.2833	0.3503	<u>0.4667</u>	0.3388	0.3419	0.3053	**0.4755**
	NDCG	0.5171	0.4854	0.4582	0.5046	0.4488	0.5009	<u>0.5917</u>	0.4918	0.4957	0.4667	**0.5986**
	WAUC	0.5587	0.5805	0.5395	0.6115	0.5257	0.5721	<u>0.6311</u>	0.5629	0.5847	0.5652	**0.6686**
Amazon Toys	AUC	0.6372	0.5138	<u>0.6576</u>	0.6321	0.5707	0.6106	0.6455	0.6127	0.5402	0.6237	**0.6712**
	MRR	0.5946	0.3293	<u>0.5949</u>	0.5669	0.3110	0.5292	0.5566	0.5070	0.3140	0.3515	**0.6385**
	NDCG	0.6879	0.4836	<u>0.6889</u>	0.6676	0.4721	0.6389	0.6602	0.6211	0.4729	0.5045	**0.7221**
	WAUC	0.6398	0.5058	<u>0.6540</u>	0.6421	0.5784	0.6184	0.6504	0.6217	0.5419	0.6305	**0.6788**

dataset. The numbers of group prototypes are searched from [1, 5, 10, 20].

4.2 Overall Performance (RQ1)

The performance comparisons over all models are as shown in Table 2, where SASRec and SURGE with better performance are leveraged as backbones on these two datasets, respectively. It can be observed that:

- **Our approach performs best**. Our model MAN significantly outperforms all baselines under all metrics. Specifically, our model improves AUC against all baselines by 4.10% and 3.85% on Micro Video A and Micro Video B, respectively, while by 8.25% and 2.07% on Amazon Video Games and Amazon Toys, respectively. In general, the improvement is more consistent across evaluation metrics on the Micro Video dataset with more overlapped users. The Amazon dataset with extremely sparse data sees the highest improvement (8.25%), which verifies that our approach can address the sparse data problem, promoting the sequential learning of both domains simultaneously and that of less interacted domains even more sharply.
- **Existing cross-domain sequential recommenders rely heavily on overlapped users or items**. PiNet and DASL are based on fully-overlapped user datasets [19, 29], but they are indeed comparable with GRU4REC under datasets without fully-overlapped users, either outperforming or even underperforming. In contrast to them, our proposed approach outperforms all baselines and improves the backbones significantly, which illustrates the effectiveness of our cross-domain modeling without user overlapping. Though NATR achieves decent performance on the Micro Video dataset with a lot of overlapped items, it fails to achieve effective cross-domain modeling on the Amazon dataset with limited overlapped items.

- **Sequential recommenders are effective but with data sparsity bottleneck**. Based on the Micro Video dataset, comparing the sequential models (i.e., Caser, GRU4REC, DIEN, SASrec, SLi-Rec, and SURGE) with the non-sequential model (i.e., DIN), it is necessary for us to model the chronological relationship between items. Besides, SASRec and SURGE are comparable and outperform all other single-domain sequential models, which illustrates the capacity of self-attention to handle long-term information and verifies the effectiveness of compressing information with metric learning. The observation of sequential models is consistent with the experimental results of the SURGE [2] paper. Based on the Amazon dataset, DIN even outperforms some sequential models, i.e., SASRec, which also drops a lot under such a short sequence scene. Though sequential models are the potential for capturing the chronological relationship between items, they are blocked by the data sparsity. Besides we have also attempted to train them with two domains simultaneously ("Shared" models in the backbone study), but the results show that one domain's optimization will have a negative impact on another domain, leading to negative transfer. Thus it is necessary to design cross-domain modeling to avoid negative transfer.

4.3 Impact of Each Component (RQ2)

To study the impact of our proposed components, we compare our model with that detaching Item Similarity Attention (ISA) module, Sequence-fusion Attention (SFA) module, and Group-prototype Attention (GPA) module on two datasets under four evaluation metrics, as shown in Table 3. Firstly, it can be observed that the shared group prototypes of GPA are most effective in both Micro Video and Amazon Video datasets, illustrating that there are similar interest groups across different domains. Besides, the performance also drops a bit when removing the sequence-fusion component (most effective in Micro Video B), i.e., SFA for fusing the local and

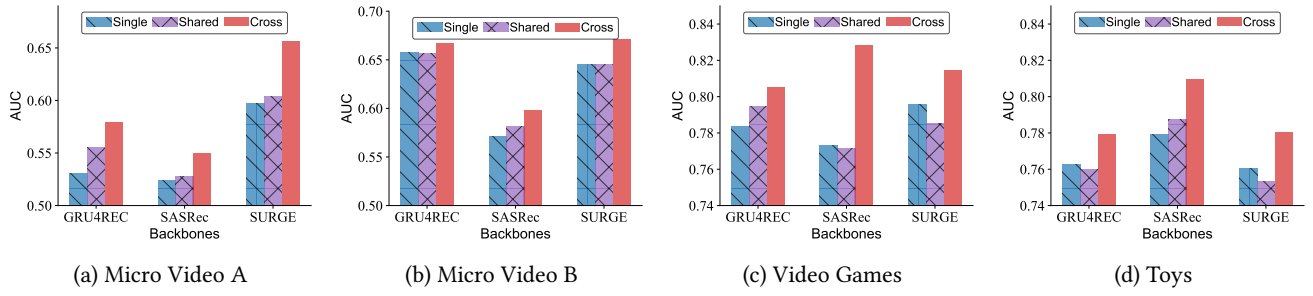(a) Micro Video A   (b) Micro Video B   (c) Video Games   (d) Toys

Figure 3: AUC performance of MAN with different backbones on Micro Video dataset and Amazon dataset. Here "Single" means backbone models trained with single domain data, which refers to the local module. "Shared" means shared backbone model trained with cross-domain data, which refers to the global module. "Cross" is the backbone equipped with our method.

(a) Micro Video A   (b) Micro Video B   (c) Video Games   (d) Toys
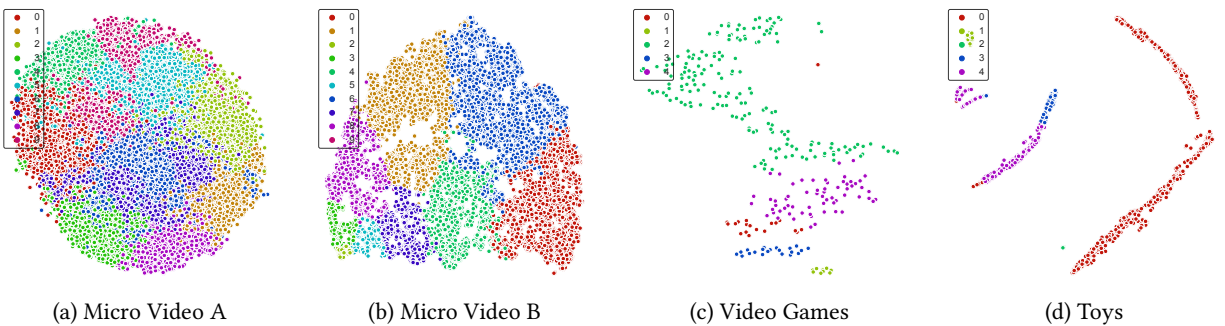
Figure 4: K-Means and t-SNE visualization of pooled group representations on Micro Video dataset and Amazon dataset, with different colors representing different groups. Group patterns across domains of two datasets are captured by the different distribution of group representations. (Best view in color.)

Table 3: Ablation study of the proposed components on Micro Video dataset and Amazon dataset.

Domain	Model	AUC	MRR	NDCG@10	WAUC
Micro Video A	w/o ISA	0.8136	0.5795	0.6826	0.8216
	w/o SFA	0.8133	0.5895	0.6904	0.8292
	w/o GPA	0.7983	0.5714	0.6762	0.8134
	w all	**0.8285**	**0.6167**	**0.7112**	**0.8435**
Micro Video B	w/o ISA	0.8059	0.5644	0.6711	0.8162
	w/o SFA	0.7939	0.5559	0.6643	0.8073
	w/o GPA	0.7996	0.5631	0.6701	0.8147
	w all	**0.8094**	**0.5756**	**0.6797**	**0.8215**
Amazon Video Games	w/o ISA	0.6195	0.4437	0.5743	0.6352
	w/o SFA	0.642	0.4426	0.5735	0.6523
	w/o GPA	0.6195	0.4198	0.5549	0.6288
	w all	**0.6559**	**0.4755**	**0.5986**	**0.6686**
Amazon Toys	w/o ISA	0.6499	0.5497	0.6548	0.6516
	w/o SFA	0.6502	0.6126	0.7021	0.6583
	w/o GPA	0.6546	0.6183	0.7074	0.6606
	w all	**0.6712**	**0.6385**	**0.7221**	**0.6788**

global sequential patterns, which means there are truly common sequential patterns across different domains. There are also similar

items across different domains when the performance decreases after the detaching item similarity attention module (most effective in Amazon Toys).

In short, Group-prototype Attention is the most important among the three proposed attentions.

4.4 Backbone Study (RQ3)

Here, we study whether GRU4REC, SASRec, and SURGE can be boosted under our proposed method. That is to say, whether our proposed method is model-agnostic. The reason why we choose these three models is that they perform better on the experimented datasets. Figure 3 shows the results of our method with different backbones on two datasets under AUC evaluation, where we can observe that:

- **Our proposed method is model-agnostic**. The selected backbones are all boosted by our proposed MAN, which means our proposed method is model-agnostic. The backbones selected here are RNN-based, attention-based, and even graph-based models. Thus our method can be applied in various state-of-the-art sequential recommendation models to boost their performance.
- **Our proposed method performs better on larger datasets**. The improvement on the Micro Video dataset is generally more obvious than that on the Amazon dataset. This is because a large dataset can provide rich cross-domain information.

4.5 User Group Visualization (RQ4)

In this section, the embeddings of all users' pooled group representations will be visualized to show the patterns our group-prototype attention module has captured.

The pooled group representation (calculated in Eqn.(12)) for each user is visualized with K-Means and t-SNE, as shown in Figure 4. More specifically, we first apply K-Means on the pooled group representations to cluster data into N_g groups. Then t-SNE is exploited to reduce the group representations into two-dimensional space, and the clustered groups by K-Means are used to label each user. It can be observed that: (1) for each dataset, the group patterns vary across different domains, where the users under Micro Video A are distributed evenly while the users under Micro Video B mostly belong to groups 0, 1, and 6. On the Amazon dataset, the users mostly belong to group 2, and group 0 under Video Games and Toys, respectively; (2) for two datasets, the users on the Amazon dataset are distributed more unbalanced and dispersed than those on the Micro Video dataset, which may because the Amazon dataset is more sparse.

5 RELATED WORK

Sequential Recommendation Sequential Recommendation [38] is the fundamental model of our work, which models the user's historical behaviors as a sequence of time-aware items, aiming to predict the probability of the next item. Initially, the Markov chain is exploited to model the sequential pattern of item sequence as FPMC [34]. To further extract the high-order interaction between the historical items, researchers have also applied deep learning models such as recurrent neural network [4, 12], convolution neural network [18] and attention network [37] in recommender systems [11, 15, 36, 46, 47]. However, recurrent neural network-based and convolution neural network-based methods often pay attention to the recent items before the next item, failing to model the long-term interest. Recently, researchers have also combined the sequential recommendation model and traditional recommendation model such as matrix factorization [17] to model the long and short-term interest [42, 45] while SURGE [2] exploits metric learning to compress the item sequence. Some recent works like DFAR [23] and DCN [22] focus on capturing more complex relations behind sequential recommendation. In this paper, we perform cross-domain learning based on sequential recommendation models to achieve knowledge transfer between different domains.

Cross-Domain Recommendation Cross-domain recommender systems [1] are an effective solution to the highly sparse data problem and cold-start problem that sequential recommendation meets. Early cross-domain recommendation models are based on single-domain recommendation, assuming that auxiliary user behaviors across different domains will benefit the target domain's user modeling [14, 26, 35]. Indeed, the most popular approaches are often based on transfer learning [31] to transfer the user embedding or item embedding from the source domain to improve the target domain's modeling, including MiNet [30], CoNet [13] and itemCST [32] etc.

However, industrial platforms tend to improve all domains of their products simultaneously instead of improving the target domain without consideration of the source domain. Thus, dual learning [10, 25], which can achieve simultaneous improvements across

both source domain and target domain, grabs researchers' attention and has already been applied in cross-domain recommender systems [20, 48]. Moreover, to enhance the recommendation performance across all domains simultaneously, researchers have proposed some dual-target approaches focusing on sequential modeling [3, 19, 29], which addresses the sparse data problem and cold-start problem promisingly and considers the performance of both source domain and target domain. Specifically, PiNet [29] tackles the shared account problem and transfers account information from one domain to another domain where the account also has historical behaviors; DASL [19] proposes dual embedding to interact embeddings and dual attention to mix the sequential patterns for the same users across two domains. Besides PiNet and DASL, DAT-MDI [3] applies dual attention like DASL on session-based recommendation without relying on user overlapping. However, requiring the item sequence pairs in two domains as input is unreasonable because the item sequences of two domains are often independent of each other despite belonging to the same user. Hence such a dual attention manner by mixing the sequence embedding of two domains will not result in a promising performance, theoretically speaking, under a non-overlapped user scene. Though NATR [5] tends to avoid user overlapping, it is a non-sequential and single-target model.

In this paper, we perform cross-domain learning in a dual-target manner to achieve simultaneous improvements across different domains without any prior assumption of overlapped users or items.

6 CONCLUSIONS AND FUTURE WORK

In this work, we studied the task of sequential recommender systems in a cross-domain manner from a more practical perspective without any prior assumption of overlapped users. Such exploration brought us three key challenges from the item, sequence, and group levels. To address these three challenges, we proposed a novel solution named MAN with local and global modules, mixing three attention networks and transferring at the group level. The first one was the local/global encoding layer that captures the sequential pattern from domain-specific and cross-domain perspectives. Secondly, we further proposed the item similarity attention that captured the similarity between local/global item embeddings and target item embedding, the sequence-fusion attention that fused sequential patterns across global encoder and local encoder, and the group-prototype attention with several group prototypes to share the sequential user behaviors implicitly without leveraging the user ID. Finally, we proposed a local/global prediction layer to evolve the domain-specific and cross-domain interests. As for future work, we plan to conduct online A/B tests to further evaluate our proposed solution's recommendation performance.

REFERENCES

[1] Iván Cantador, Ignacio Fernández-Tobías, Shlomo Berkovsky, and Paolo Cremonesi. 2015. Cross-domain recommender systems. In *Recommender systems handbook*. Springer, 919–959.

[2] Jianxin Chang, Chen Gao, Yu Zheng, Yiqun Hui, Yanan Niu, Yang Song, Depeng Jin, and Yong Li. 2021. Sequential Recommendation with Graph Neural Networks. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 378–387.

[3] Chen Chen, Jie Guo, and Bin Song. 2021. Dual attention transfer in session-based recommendation with multi-dimensional integration. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 869–878.

[4] Junyoung Chung, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS 2014 Workshop on Deep Learning, December 2014*.

[5] Chen Gao, Xiangning Chen, Fuli Feng, Kai Zhao, Xiangnan He, Yong Li, and Depeng Jin. 2019. Cross-Domain Recommendation Without Sharing User-Relevant Data. In *The World Wide Web Conference* (San Francisco, CA, USA) *(WWW '19)*. Association for Computing Machinery, New York, NY, USA, 491–502. https://doi.org/10.1145/3308558.3313538

[6] Chen Gao, Yong Li, Fuli Feng, Xiangning Chen, Kai Zhao, Xiangnan He, and Depeng Jin. 2021. Cross-domain Recommendation with Bridge-Item Embeddings. 16, 1 (2021), 1–23. Publisher: ACM New York, NY.

[7] Chen Gao, Yu Zheng, Wenjie Wang, Fuli Feng, Xiangnan He, and Yong Li. 2024. Causal inference in recommender systems: A survey and future directions. *ACM Transactions on Information Systems* (2024).

[8] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *AISTATS*. 249–256.

[9] Asela Gunawardana and Guy Shani. 2015. Evaluating Recommender Systems. In *Recommender Systems Handbook*, Francesco Ricci, Lior Rokach, and Bracha Shapira (Eds.). Springer US, 265–308. https://doi.org/10.1007/978-1-4899-7637-6_8

[10] Di He, Yingce Xia, Tao Qin, Liwei Wang, Nenghai Yu, Tie-Yan Liu, and Wei-Ying Ma. 2016. Dual learning for machine translation. 29 (2016), 820–828.

[11] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2016. Session-based recommendations with recurrent neural networks. In *ICLR*.

[12] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[13] Guangneng Hu, Yu Zhang, and Qiang Yang. 2018. Conet: Collaborative cross networks for cross-domain recommendation. In *Proceedings of the 27th ACM international conference on information and knowledge management*. 667–676.

[14] Liang Hu, Jian Cao, Guandong Xu, Longbing Cao, Zhiping Gu, and Can Zhu. 2013. Personalized recommendation via cross-domain triadic factorization. In *Proceedings of the 22nd international conference on World Wide Web*. 595–606.

[15] Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In *2018 IEEE International Conference on Data Mining (ICDM)*. IEEE, 197–206.

[16] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*.

[17] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009).

[18] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. Imagenet classification with deep convolutional neural networks. 25 (2012), 1097–1105.

[19] Pan Li, Zhichao Jiang, Maofei Que, Yao Hu, and Alexander Tuzhilin. 2021. Dual Attentive Sequential Learning for Cross-Domain Click-Through Rate Prediction. *arXiv preprint arXiv:2106.02768* (2021).

[20] Pan Li and Alexander Tuzhilin. 2020. Ddtcdr: Deep dual transfer cross domain recommendation. In *Proceedings of the 13th International Conference on Web Search and Data Mining*. 331–339.

[21] Feng Liang, Weike Pan, and Zhong Ming. 2021. Fedrec++: Lossless federated recommendation with explicit feedback. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. 4224–4231.

[22] Guanyu Lin, Chen Gao, Yinfeng Li, Yu Zheng, Zhiheng Li, Depeng Jin, and Yong Li. 2022. Dual contrastive network for sequential recommendation. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*. 2686–2691.

[23] Guanyu Lin, Chen Gao, Yu Zheng, Jianxin Chang, Yanan Niu, Yang Song, Zhiheng Li, Depeng Jin, and Yong Li. 2023. Dual-interest Factorization-heads Attention for Sequential Recommendation. In *Proceedings of the ACM Web Conference 2023*. 917–927.

[24] Guanyu Lin, Feng Liang, Weike Pan, and Zhong Ming. 2020. Fedrec: Federated recommendation with explicit feedback. *IEEE Intelligent Systems* 36, 5 (2020), 21–30.

[25] Mingsheng Long, Jianmin Wang, Guiguang Ding, Wei Cheng, Xiang Zhang, and Wei Wang. 2012. Dual transfer learning. In *Proceedings of the 2012 SIAM International Conference on Data Mining*. SIAM, 540–551.

[26] Babak Loni, Yue Shi, Martha Larson, and Alan Hanjalic. 2014. Cross-domain collaborative filtering with factorization machines. In *European conference on information retrieval*. Springer, 656–661.

[27] Yichao Lu, Ruihai Dong, and Barry Smyth. 2018. Why I like it: multi-task learning for recommendation and explanation. In *Proceedings of the 12th ACM Conference on Recommender Systems*. 4–12.

[28] Jianxin Ma, Chang Zhou, Hongxia Yang, Peng Cui, Xin Wang, and Wenwu Zhu. 2020. Disentangled Self-Supervision in Sequential Recommenders. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 483–491.

[29] Muyang Ma, Pengjie Ren, Yujie Lin, Zhumin Chen, Jun Ma, and Maarten de Rijke. 2019. π-Net: A parallel information-sharing network for shared-account cross-domain sequential recommendations. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 685–694.

[30] Wentao Ouyang, Xiuwu Zhang, Lei Zhao, Jinmei Luo, Yu Zhang, Heng Zou, Zhaojie Liu, and Yanlong Du. 2020. MiNet: Mixed Interest Network for Cross-Domain Click-Through Rate Prediction. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2669–2676.

[31] Sinno Jialin Pan and Qiang Yang. 2009. A survey on transfer learning. 22, 10 (2009), 1345–1359. Publisher: IEEE.

[32] Weike Pan, Evan Xiang, Nathan Liu, and Qiang Yang. 2010. Transfer learning in collaborative filtering for sparsity reduction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 24. Issue: 1.

[33] Ekagra Ranjan, Soumya Sanyal, and Partha Talukdar. 2020. Asap: Adaptive structure aware pooling for learning hierarchical graph representations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 5470–5477.

[34] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. Factorizing personalized markov chains for next-basket recommendation. In *WWW*. 811–820.

[35] Ajit P. Singh and Geoffrey J. Gordon. 2008. Relational learning via collective matrix factorization. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. 650–658.

[36] Jiaxi Tang and Ke Wang. 2018. Personalized top-n sequential recommendation via convolutional sequence embedding. In *WWW*. 565–573.

[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NeurIPS*. 5998–6008.

[38] Shoujin Wang, Liang Hu, Yan Wang, Longbing Cao, Quan Z. Sheng, and Mehmet Orgun. 2019. Sequential recommender systems: challenges, progress and prospects. (2019).

[39] Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S Yu. 2019. Heterogeneous graph attention network. In *The World Wide Web Conference*. 2022–2032.

[40] Zhenlei Wang, Jingsen Zhang, Hongteng Xu, Xu Chen, Yongfeng Zhang, Wayne Xin Zhao, and Ji-Rong Wen. 2021. Counterfactual data-augmented sequential recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 347–356.

[41] Rex Ying, Jiaxuan You, Christopher Morris, Xiang Ren, William L Hamilton, and Jure Leskovec. 2018. Hierarchical graph representation learning with differentiable pooling. *arXiv preprint arXiv:1806.08804* (2018).

[42] Zeping Yu, Jianxun Lian, Ahmad Mahmoody, Gongshen Liu, and Xing Xie. 2019. Adaptive User Modeling with Long and Short-Term Preferences for Personalized Recommendation. In *IJCAI*. 4213–4219.

[43] An Zhang, Leheng Sheng, Zhibo Cai, Xiang Wang, and Tat-Seng Chua. 2023. Empowering Collaborative Filtering with Principled Adversarial Contrastive Loss. In *NeurIPS*.

[44] Cheng Zhao, Chenliang Li, Rong Xiao, Hongbo Deng, and Aixin Sun. 2020. Catn: Cross-domain recommendation for cold-start users via aspect transfer network. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 229–238.

[45] Wei Zhao, Benyou Wang, Jianbo Ye, Yongqiang Gao, Min Yang, and Xiaojun Chen. 2018. PLASTIC: Prioritize Long and Short-term Information in Top-n Recommendation using Adversarial Training. In *IJCAI*. 3676–3682.

[46] Guorui Zhou, Na Mou, Ying Fan, Qi Pi, Weijie Bian, Chang Zhou, Xiaoqiang Zhu, and Kun Gai. 2019. Deep interest evolution network for click-through rate prediction. In *AAAI*. 5941–5948.

[47] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. Deep interest network for click-through rate prediction. In *KDD*. 1059–1068.

[48] Feng Zhu, Chaochao Chen, Yan Wang, Guanfeng Liu, and Xiaolin Zheng. 2019. Dtcdr: A framework for dual-target cross-domain recommendation. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 1533–1542.