# MLoRA: Multi-Domain Low-Rank Adaptive Network for Click-Through Rate Prediction

Zhiming Yang\* yangzhiming@mail.nwpu.edu.cn Northwestern Polytechnical University Xi'an, China

Luwei Yang<sup>†</sup> luwei.ylw@alibaba-inc.com Alibaba Group Hangzhou, China Haining Gao\*
gaohaining.ghn@alibaba-inc.com
Alibaba Group
Hangzhou, China

Libin Yang<sup>†</sup>
{libiny}@nwpu.edu.cn
Northwestern Polytechnical
University
Xi'an, China

Dehong Gao\* dehong.gdh@nwpu.edu.cn Northwestern Polytechnical University Xi'an, China

Xiaoyan Cai xiaoyanc@nwpu.edu.cn Northwestern Polytechnical University Xi'an, China

Wei Ning wei.ningw@alibaba-inc.com Alibaba Group Hangzhou, China

#### **ABSTRACT**

Click-through rate (CTR) prediction is one of the fundamental tasks in the industry, especially in e-commerce, social media, and streaming media. It directly impacts website revenues, user satisfaction, and user retention. However, real-world production platforms often encompass various domains to cater for diverse customer needs. Traditional CTR prediction models struggle in multi-domain recommendation scenarios, facing challenges of data sparsity and disparate data distributions across domains. Existing multi-domain recommendation approaches introduce specific-domain modules for each domain, which partially address these issues but often significantly increase model parameters and lead to insufficient training. In this paper, we propose a Multi-domain Low-Rank Adaptive network (MLoRA) for CTR prediction, where we introduce a specialized LoRA module for each domain. This approach enhances the model's performance in multi-domain CTR prediction tasks and is able to be applied to various deep-learning models. We evaluate the proposed method on several multi-domain datasets. Experimental results demonstrate our MLoRA approach achieves a significant improvement compared with state-of-the-art baselines. Furthermore, we deploy it in the production environment of the

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Conference acronym 'XX, June 03-05, 2018, Woodstock, NY

© 2018 Copyright held by the owner/author(s). Publication rights licensed to ACM. ACM ISBN 978-1-4503-XXXX-X/18/06 https://doi.org/XXXXXXXXXXXXXXX

Guannan Zhang zgn138592@alibaba-inc.com Alibaba Group Hangzhou, China

Alibaba.COM <sup>1</sup>. The online A/B testing results indicate the superiority and flexibility in real-world production environments. The code of our MLoRA is publicly available <sup>2</sup>.

#### CCS CONCEPTS

• Information systems → Learning to rank; Content ranking; Novelty in information retrieval; Social recommendation.

## **KEYWORDS**

Low-Rank Adaptive, Click-Through Rate Prediction, Multi-domain

## **ACM Reference Format:**

Zhiming Yang, Haining Gao, Dehong Gao, Luwei Yang, Libin Yang, Xiaoyan Cai, Wei Ning, and Guannan Zhang. 2018. MLoRA: Multi-Domain Low-Rank Adaptive Network for Click-Through Rate Prediction. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation emai (Conference acronym 'XX)*. ACM, New York, NY, USA, 11 pages. https://doi.org/XXXXXXXXXXXXXXXX

## 1 INTRODUCTION

Click-Through Rate (CTR) prediction is currently one of the most crucial estimation tasks, serving as a fundamental aspect in comprehending user behavior and driving personalized experiences [2, 40, 57]. It plays an essential role across various application areas such as personalized recommendation systems [26, 32, 33, 36], information retrieval [27, 38, 39, 44], and online advertising [20, 30, 47, 50, 56, 57, 60]. For instance, in the content recommendation field, platforms like TikTok leverage CTR prediction algorithms to curate personalized feeds for users, ensuring captivating and relevant browsing experiences [3, 10, 40, 42, 46, 57]. While in the products recommendation area, platforms like Alibaba.com utilize CTR prediction to boost conversion rate which in turn increases website revenues [54, 55].

<sup>\*</sup>These authors have contributed equally to this work.

<sup>&</sup>lt;sup>†</sup>Corresponding authors.

<sup>1</sup>https://www.alibaba.com/

<sup>&</sup>lt;sup>2</sup>https://github.com/gaohaining/MLoRA

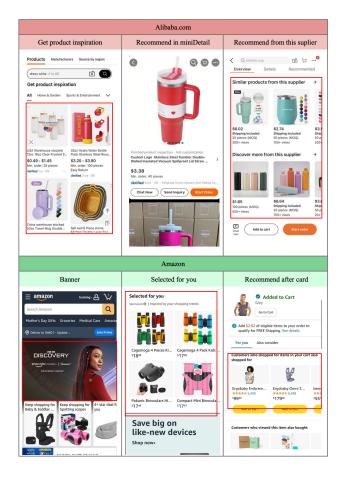


Figure 1: Upper: three representative business domains of Alibaba.com, Get product inspiration, Recommend in miniDetail and Recommend from this supplier. Lower: three key business sectors of Amazon, Banner, Selected for you and Recommend after card.

For the sake of simplification, traditional CTR prediction approaches usually focus on one single domain, where the CTR model utilizes samples collected from a particular domain [3, 10, 12, 46]. However, the **multi-domain CTR prediction** is ubiquitous in reality, especially for large-scale platforms (e.g., Taobao or Amazon e-commerce platforms), which is still a hard nut to crack [5, 15, 22]. Taking Alibaba.com recommendation system as an example, there are hundreds of domains (e.g., *Get product inspiration, Recommend in miniDetail*, and *Recommend from this supplier*) to satisfy the user demands and preferences. Each domain showcases a series of related products under a given topic as seen in the upper part of Figure 1. Another Amazon example can be seen in the lower part.

Currently, two types of approaches are being researched to address the problem of multi-domain CTR prediction. The first type treats multiple domains independently and builds separate CTR prediction models for each domain as seen in Figure 2 (a) [3, 40, 60]. These approaches suffer from the problems of **data sparsity** since there is a limited volume of data for training a domain. Meanwhile,

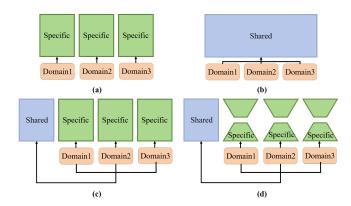


Figure 2: (a) Training separated model for each domain. (b) Training a unified model by mixing multi-domain data. (c) Multi-domain Framework [29, 45, 48]. (d) Our Proposed MLoRA.

these approaches take less consideration of the relationships between multiple domains [3, 10, 12, 40, 42, 46]. The second type trains a unified model by merging multi-domain data as seen in Figure 2 (b) [6, 23]. Although it solves the issue of data sparsity to a certain degree, this approach often fails to capture **domain diversity** as the mixed data neglects the differences in data distribution between domains. In response to the above two issues of multi-domain recommendation, researchers have proposed corresponding solutions shown as Figure 2 (c). Researchers [29, 45, 48] alleviated the issue of sparse data by employing the shared part to learn an overall data distribution and introducing specific parts for each domain, thereby capturing unique data distribution information for each domain. However, these approaches still lead to a sharp increase in **model parameters** and suffer from **insufficient learning** in some domains with small data volumes.

Inspired by the success of Low-Rank Adaptor(LoRA) in large language model finetuning [13], we propose the Multi-domain LoRA (i.e., MLoRA) approach for multi-domain CTR prediction, and its code is publicly available. The LoRA adaptor utilizes a low-rank structure to finetune Large Language Models (LLMs), effectively learning knowledge from domain data with fewer parameters. Meanwhile, we build multiple LoRA adaptors for each domain, which learn data distribution of each domain more efficiently. Firstly, the Multi-domain LoRA approach effectively captures the unique data distributions of each domain, which facilitates the model in recognizing the diversity of information across different domains. Secondly, the LoRA adaptor requires significantly fewer parameters, thus avoiding the potential issues of inadequate training on domains with sparse data and high computational cost. The MLoRA approach is scalable and flexible to new domains by applying additional LoRA adaptors. In addition, the proposed approach is a general framework, which is able to be applied to various CTR models such as WDL, PNN, NFM, etc [3, 10, 12, 14, 24, 40, 46, 49].

We have conducted extensive experiments on several public datasets. The experimental results indicate the superiority across multiple datasets by equipping current mainstream CTR models with MLoRA. Furthermore, our MLoRA approach has been deployed

on the Alibaba.COM e-commerce website, which integrated 10 core recommendation domains and the additional parameter size is only increased by 1.76%. It contributes to a 1.49% increase in CTR and a 3.37% increase in order conversion rate, as well as a 2.71% increase in the number of paid buyers across the entire site. The main contributions of this paper are summarized as follows:

- We propose a novel approach MLoRA to address the problem of multi-domain CTR prediction by incorporating LoRA adaptors. The MLoRA approach is model-agnostic, which can be applied to nearly all deep learning-based CTR approaches.
- We introduce multiple LoRA networks, where the integration of mixed data mitigates data sparsity, while the implementation of multiple LoRA adaptors captures the diversity within each domain.
- We evaluate our proposed MLoRA on several datasets with state-of-the-art methods. Extensive experimental results demonstrate the effectiveness and superiority of our MLoRA approach.
- We deploy MLoRA in a real-world production e-commerce website. The A/B testing results verify its advancement in CTR and conversion rate only at a small cost of parameter increase.

## 2 RELATED WORK

In this section, we will review the related work from the following two respects: CTR Prediction and Low-Rank Adaptor.

# 2.1 CTR Prediction

The prediction of Click-Through Rate is a fundamental task in recommendation, advertising and search systems, where an effective CTR prediction model is crucial for enhancing commercial value. We will describe existing methods in two groups: Single-Domain Approaches and Multi-Domain Approaches.

**Single-Domain Approaches.** In recent years, the methodology for CTR tasks has shifted from traditional shallow [9, 16, 17, 42, 63] approaches to deep learning techniques [3, 10, 40, 57, 59, 60]. Deep learning-based CTR models typically follow an architecture consisting of embedding and multi-layer perceptron (MLP) layers. The embedding layer initially transforms sparse ID features into dense vectors and continuous numeric features are also discretized through bucketing before embedding. The MLP layer takes all embeddings as inputs to perform deep feature interactions. PNN [40] adds a product layer between the embedding and MLP layers to explicitly carry out feature interactions. WDL [3] combines a linear layer for memorizing information on the wide side with a deep MLP network for generalization on the deep side. DCN [49] introduces a cross-network to facilitate feature interactions on the wide side, while DeepFM [10] incorporates a Factorization Machine to boost the capability for feature interaction on the wide side. NFM [12] improves the deep side by introducing the Bi-Interaction Layer structure to process second-order crossing information, enabling deep MLP networks to better capture high-order combinations between features. Another innovation for the multi-layer MLP is AFM [53], which implements an attention mechanism, allowing

each feature interaction to be weighted differently, thereby filtering valuable combination information. AutoInt [46] and CAN [58] further propose the self-attention mechanism for comprehensive feature interactions. FiBiNET [14] dynamically learns feature importance and employs a Bilinear method to learn fine-grained feature interactions. xDeepFM [24] builds on the WDL [3] model by introducing the Compressed Interaction Network (CIN) module, which generates vector-wise level feature interactions explicitly. The aforementioned methods are mainly from the perspective of feature interactions, another modeling approaches start from the angle of user behavior modeling. DIN [60] innovatively incorporates the user behavior sequence and employs an attention mechanism to capture diverse interests related to the target item. DIEN [59] introduces GRU to model the evolution and transformation of user interests over time. MIMN [37] proposes a novel memory-based architecture, which is designed to capture user interests from long sequential behavior sequences. It provides the capability to process sequences extending to thousands in length.

Multi-Domain Approaches. In real-world applications, data is often collected from multiple domains. Multi-domain data typically presents two challenges: firstly, some domain data is sparse, which cannot meet the requirements for model training. Secondly, there are differences in data between different domains, making it difficult to apply a single-domain approach rigidly. Facing the challenge of CTR prediction in multi-domain data scenarios, researchers have proposed some solutions. Domain generalization (DG) methods [62] offer a solution approach to the multi-domain CTR prediction problem by extracting common knowledge from multiple domains and learning features that generalize to unknown domains. Li et al. [21] suggest that this approach can effectively alleviate the issue of data sparsity. STAR [45] proposed Partitioned Normalization(PN) and domain-specific FCN to capture the uniqueness of each domain's features. MMOE [29] adopts a Mixture of Experts (MoE) architecture, sharing expert modules across all domains while training a gating network for each domain. PLE [48] achieves joint learning of information from different domains by explicitly separating shared and task-specific components. However, the task-specific component is almost the same as the shared part, it will cause a significant increase in model parameters. Moreover, it still suffers from insufficient training in sparse data domains.

## 2.2 Low-Rank Adaptor

Over the past year, the pretraining [4, 18, 34, 41] and finetuning [25, 31, 35, 51, 52] architecture has become a fundamental modeling approach in the NLP field. Pretraining allows the model to acquire generalizable knowledge from a large and diverse dataset, while finetuning makes the model more closely match the data distribution of a specific domain. However, in many domains, the amount of data available for finetuning is very limited, sometimes even much less than the number of parameters in the model, making it a challenge for finetuning tasks to learn sufficiently. In light of this, a new perspective based on analyzing intrinsic dimensionality [19] is proposed. The intrinsic dimension describes the minimum number of dimensions needed to solve the problem. Measuring the intrinsic dimension will tell us how many free parameters are required to closely approximate the solution to the problem while finetuning

for each target task. Studies [1, 19] show that in some cases, the intrinsic dimension is several orders of magnitude smaller than the full parameterization.

Therefore, a finetuning approach named LoRA [13] is proposed, which freezes the pretrained model weights and injects trainable rank decomposition matrices, named A and B, into each layer. Benefiting from the low-rank, the number of parameters in the LoRA module that need to be finetuned is significantly lower than that of the pretrained parameters. This enables downstream tasks to be finetuned very efficiently, achieving commendable results while also reducing the amount of training data required. A pretrained model can be shared and employed to construct numerous small LoRA modules for various tasks across different domains. By freezing the shared model and efficiently switching tasks through the replacement of low-rank matrices A and B, we can significantly reduce the storage requirements and the overhead related to task switching.

## 3 METHODOLOGY

In this section, we present the proposed MLoRA in detail. Figure 3 illustrates the overall network architecture of our model.

## 3.1 Problem definition

We define the notations and problem settings in this paper. The model uses sparse/dense inputs such as user features, item features, context features, and so on. The predicted target  $\hat{y}$  is the probability that user u will interact with item i, which is calculated via:

$$\hat{y} = F(E(u_1), ..., E(u_n); E(i_1), ..., E(i_m); E(c_1), ..., E(c_k)),$$
 (1)

where  $\{u_1, ..., u_n\}$  is the set of user features.  $\{i_1, ..., i_m\}$  is the set of target item feature and  $\{c_1, ..., c_k\}$  is the set of context features. The  $E(i) \in \mathbb{R}^d$  means the embedding layer, which maps the sparse feature into learnable dense vectors. Continuous numeric features are discretized through bucketing before embedding.

When jointly modeling across multiple domains, we need to integrate the data from all domains and take the domain ID as input, which is calculated via:

$$\hat{y_t} = F(E(u_1), ..., E(u_n); E(i_1), ..., E(i_m); E(c_1), ..., E(c_k); E(t)),$$
 (2)

where  $\hat{y_t}$  is the probability that user u will interact with item i in domain t (i.e., the domain ID).

# 3.2 Low-Rank Adaptor

LoRA is a finetuning approach based on an improvement of a single fully connected layer which has a full-rank parameter matrix  $\mathbf{W} \in R^{d_{out} \times d_{in}}$ , where  $d_{in}$  is the dimension of the input and  $d_{out}$  is the dimension of the output. Assuming that after finetuning, the parameter matrix  $\mathbf{W}$  is updated to  $\mathbf{W}'$ . Without loss of generality, finetuning can be described as learning  $\Delta \mathbf{W}$ , where

$$\Delta \mathbf{W} = \mathbf{W}' - \mathbf{W},\tag{3}$$

When finetuning a large pretrained model for a specific task, the full-rank matrix often has considerable parameter redundancy [1, 13, 19], with a rank  $r \ll min(d_{in}, d_{out})$ , then  $\Delta \mathbf{W}$  can be represented as :

$$\Delta \mathbf{W} = \mathbf{W'} - \mathbf{W} = \mathbf{B}\mathbf{A},\tag{4}$$

where  $\mathbf{B} \in R^{d_{out} \times r}$  and  $\mathbf{A} \in R^{r \times d_{in}}$ . It is particularly important to note that in order to ensure that  $\Delta \mathbf{W}$  is zero at the beginning of training,  $\mathbf{A}$  is initialized with a Gaussian distribution, and  $\mathbf{B}$  is initialized with zeros. Therefore, the forward pass of LoRA can be represented as:

$$h = W'x = Wx + BAx, (5)$$

where x is the input and h is the output.

## 3.3 MLoRA for Multi-domain CTR

When training a CTR model, there are two parts that need to be fitted: the generalizable common information and the personalized distinctive information. Thus, the predicted target  $\hat{y}$  can be calculated via:

$$\hat{y} = F(\mathbf{x}) = F_0(\mathbf{x}) + \Delta F(\mathbf{x}), \tag{6}$$

Considering the sparsity of individual domains, a model L(x) utilizing low-rank matrices as parameters can be used to learn personalized information for each domain. Then the  $\hat{y}$  can be calculated via:

$$\hat{\mathbf{y}} = F_0(\mathbf{x}) + L(\mathbf{x}),\tag{7}$$

When modeling across different domains, the primary focus is on the commonalities and differences between domains. Therefore, a low-rank matrix model is constructed separately for each domain. Then the  $\hat{y_t}$  can be calculated via:

$$\hat{\mathbf{y}}_t = F_0(\mathbf{x}) + L_t(\mathbf{x}),\tag{8}$$

From this perspective, in order to achieve a more comprehensive fitting, we propose the approach named MLoRA, as shown in Figure 3, which splits each layer of the model into a common part and a personalized part, rather than treating the model as a whole. A single layer can be represented as:

$$\mathbf{h}_t = \mathbf{W}\mathbf{x} + \Delta \mathbf{W}_t = \mathbf{W}\mathbf{x} + \mathbf{B}_t \mathbf{A}_t \mathbf{x},\tag{9}$$

Unlike NLP models, CTR models usually exhibit significant differences in network width between layers, hence using a fixed r value across different layers is not equitable. In the MLoRA approach, a fixed scaling factor, referred to as a temperature coefficient  $\alpha$ , is employed. Thus r can be calculated via:

$$r = \max(\frac{d_{out}}{\alpha}, 1). \tag{10}$$

MLoRA applies a two-phase training strategy. In the pretraining phase, the backbone network is trained with large-scale pretraining data to learn information that can be generalized across various domains. During the finetuning phase, we add the MLoRA networks while freezing the backbone network. The finetuning phase is focused solely on **A** and **B** updates to learn personalized information for each domain.

## 4 EXPERIMENTS

Extensive experiments are conducted to evaluate our MLoRA approach. The datasets, baselines, and experimental settings are described in Section 4.1, Section 4.2, and Section 4.3, respectively. In Section 4.4, we detail the experimental design for evaluating MLoRA, conducting thorough experiments across three datasets and performing detailed analyses of the results. Finally, in Section 4.6, we discuss real-world production applications and online A/B testing results.

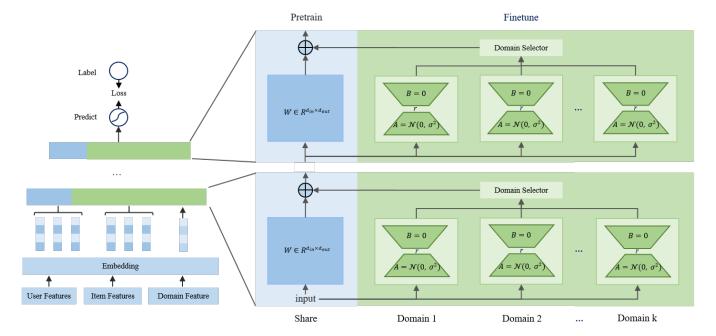


Figure 3: The overview architecture of the proposed MLoRA, which consists of input features, embedding layer, shared network, multi-domain network, and domain selector. Each domain network consists of low-rank matrix A and B, and the outputs of the domain network and share network are added together to serve as the output of the entire layer. The embedding layer and shared network are trained during the pretraining phase, while the domain network is trained during the finetuning phase.

# 4.1 Datasets

In our experiments, we construct multi-domain recommendation benchmark datasets based on three publicly available real-world datasets: Taobao [7], Amazon [11], and Movielens [11]. The statistics of them are summarized in Table 1.

The Taobao dataset consists of user click logs from various themed scenarios in Taobao APP <sup>3</sup>. These scenarios are already partitioned into domains based on different thematic topics, such as what to take when traveling, how to dress up yourself for a party and things to prepare when a baby is coming, etc. We randomly select 10, 20, and 30 domains to construct multi-domain datasets which are named **Taobao-10**, **Taobao-20**, and **Taobao-30**.

The Amazon dataset is a large-scale dataset comprising reviews (ratings, text, and helpfulness votes) collected from Amazon.com <sup>4</sup>, along with product metadata (descriptions, category information, price, brand, and image features), and links (also viewed/also bought graphs). We split domains based on product categories such as Musical Instruments and Video Games. We select 6 domains to form the dataset **Amazon-6**.

The Movielens dataset, published in February 2003, contains 1 million movie ratings collected from 6000 users across 4000 movies. We partition the Movielens dataset using gender as a categorical feature, forming the multi-domain dataset **Movielens-gen**.

# 4.2 Baselines

Both state-of-the-art single-domain and multi-domain methods are selected as baselines. For fairness, the hidden layers in all models are set to [256, 128, 64].

- MLP is a basic neural network model, consisting of multiple fully connected layers linked sequentially.
- WDL [3] combines traditional feature engineering with deep models to enable the model to possess characteristics of both memorization and generalization.
- NFM [12] utilizes the Bi-Interaction Layer structure to process second-order crossing information, allowing the DNN structure to better learn the information of cross features and reduce the difficulty of learning higher-order cross-feature information.
- AutoInt [46] achieves automatic feature cross through ingenious model design by leveraging multi-layered multihead self-attention modules stacked together combined with residual network to extract low-order and high-order crossinformation from input features.
- PNN [40] uses embedding layers to learn distributed representations of the data, captures interactive patterns between inter-field categories using product layers, and ultimately explores high-dimensional feature interactions using fully connected layers.
- DCN [49] proposes a cross-network to explicitly encode features for better model representation.

<sup>3</sup>https://world.taobao.com/

<sup>4</sup>https://www.amazon.com/

Dataset	Domain	User	Item	Train	Validation	Test
Amazon-6	6	445789	172653	9968333	3372666	3585877
Taobao-10	10	23778	6932	92137	37645	43502
Taobao-20	20	58190	16319	243592	96591	106500
Taobao-30	30	99143	29945	394805	151369	179252
Movielens-gen	2	6041	3953	600125	200042	200042

Table 1: Overall statistics of the datasets.

Table 2: The WAUC (%) results of CTR prediction on different datasets. Note Base refers to the original results of the corresponding methods and Base+MLoRA refers to the results with the assistance of MLoRA. Avg is the average results across all approaches.  $\Delta$  refers to the improvement of Base+MLoRA compared to Base.

Dataset	Approach	MLP	STAR	WDL	NFM	AutoInt	PNN	DCN	FiBiNET	DeepFM	xDeepFM	Avg
	Base	72.92	75.51	73.12	76.88	75.56	76.42	72.07	76.51	75.01	75.26	74.93
Taobao-10	Base+MLoRA	74.53	76.17	73.51	77.08	75.83	76.74	72.14	76.77	75.66	75.64	75.41
	Δ	+1.61	+0.66	+0.39	+0.20	+0.27	+0.32	+0.07	+0.26	+0.65	+0.38	+0.48
	Base	75.07	77.26	73.25	65.07	74.10	74.03	75.61	74.93	73.55	74.64	73.75
Amazon-6	Base+MLoRA	77.48	77.28	74.25	67.41	74.46	75.18	75.62	75.27	74.08	74.83	74.58
	Δ	+2.41	+0.02	+1.00	+2.34	+0.36	+1.15	+0.01	+0.34	+0.53	+0.19	+0.83
	Base	80.15	80.18	80.17	80.57	80.25	80.26	80.28	80.47	80.24	80.26	80.28
Movielens-gen	Base+MLoRA	80.39	80.36	80.41	80.59	80.41	80.39	80.42	80.74	80.46	80.44	80.46
	Δ	+0.24	+0.18	+0.24	+0.02	+0.16	+0.13	+0.14	+0.27	+0.22	+0.18	+0.18
-	$Avg(\Delta)$	+1.42	+0.29	+0.54	+0.85	+0.26	+0.53	+0.07	+0.29	+0.47	+0.25	+0.50

- FiBiNET [14] dynamically learns the importance of features using the squeeze-excitation network structure and models cross features better using a bilinear function.
- DeepFM [10] integrates the advantages of factorization machines and deep neural networks by simultaneously considering interactions between lower-order and higher-order features, enabling end-to-end learning of all feature interactions.
- xDeepFM [24] designs a compressed interaction network to explicitly learn higher-order feature interactions, combining it with traditional DNNs to jointly learn explicit and implicit higher-order features.
- STAR [45] is a multi-domain recommendation method that divides parameters into shared and specific parts. Meanwhile, it proposes Partitioned Normalization to integrate differentiated information across different domains.

## 4.3 Settings

For a fair comparison, we implement all CTR prediction approaches on TensorFlow 1.12 and run them on two RTX 2080Ti GPUs. For the Taobao dataset, as the input embeddings are provided already, there is no need to train the input embedding. For Amazon and Movielens datasets, the input embedding size is set as 8. We set the learning rate to 0.001, dropout rate to 0.5, batch size to 1024, and employed binary cross-entropy [43] as the loss function. All approaches employ a three-layer feedforward neural network with hidden layer sizes of [256, 128, 64], and temperature coefficient  $\alpha$ 

is set to 32. We divided the dataset into training, validation, and testing sets, where about three-fifths of the dataset is randomly selected as the training set, and about one-fifth is randomly chosen as the validation set, and the rest as the testing set following [28].

Our training strategy will save the parameters of the model with the best performance during training. Training would stop when the model's performance failed to exceed the best performance achieved in previous training epochs for a consecutive number of times. It is determined by a patience parameter following [28]. In practice, most training schemes set patience to 3. Due to the possibility of some schemes terminating training prematurely, to ensure fairness, we set patience=5 to maintain consistency across all schemes.

AUC (Area Under the ROC Curve) is one of the most commonly used metrics for evaluating CTR prediction [8]. Note 0.1% absolute AUC gain is regarded as significant for the CTR task [3, 46, 60]. As the data scale varies across different domains in the dataset, we utilized WAUC (Weighted AUC) [61] as the metric for evaluating model performance:

$$WAUC = \sum_{i=1}^{n} \omega_i * AUC_i, \tag{11}$$

where  $AUC_i$  refers to the AUC result for the i-th domain and  $\omega_i$  represents the proportion of the data volume for the i-th domain compared to the total data volume. It is defined as

$$\omega_i = \frac{s_i}{\sum_{i=1}^n s_i} \times 100\%,$$
 (12)

where  $s_i$  represents the size of the data for the i-th domain.

# 4.4 Main Experiments

We conduct extensive experiments on three public datasets: Taobao-10, Amazon-6, and Movielens-gen as shown in Table 1. The MLoRA approach is model-agnostic, which allows it to be applied to various CTR prediction methods. Therefore, to evaluate the performance of MLoRA, we applied it to a wide range of existing deep CTR prediction models based on deep learning and compared the results of the original models (denoted as Base) with those equipped with MLoRA. Here, WAUC(%) score is reported as the metric.

**Observation 1**: From Table. 2, it can be observed that with the assistance of MLoRA, all approaches achieve performance improvements on the Taobao-10, Amazon-6, and Movielens-gen datasets. The improvements across all methods on the three datasets are 0.49%, 0.83%, and 0.18%, respectively, with an average of 0.5%. These improvements are statistical significant by applying a paired T-test. This may be due to the base models (except for STAR) training a single network by mixing multi-domain data, which fails to capture the diversity among domains. Introducing LoRA adaptors helps alleviate this limitation. The STAR approach copies the shared network to each specific domain and trains the domain network again. Due to the large parameter size of the domain network and data sparsity, the overall model tends to be under-fitting. Therefore, introducing MLoRA can also improve the performance of STAR by reducing the complexity and parameterization of specific FCNs through the introduction of LoRA. This claims that MLoRA is a model-agnostic framework that can be easily deployed on various deep CTR prediction networks. It effectively alleviates data sparsity issues and captures inter-domain diversity information, demonstrating good transferability and feasibility.

**Observation 2**: By comparing the results of MLoRA on three datasets, we find that on the Amazon-6 dataset, MLoRA achieves the highest average performance improvement of 0.83% (from 73.75% to 77.48%). We attribute this to the larger dataset size of Amazon-6 and the more pronounced differences between domains, which the base model fails to learn sufficiently. Introducing specific LoRA adaptors for each domain allows the model to capture differentiated information better, leading to a significant performance boost. Furthermore, the performance improvement on the Movielens dataset is the lowest at 0.18% (from 80.28% to 80.46%), with all base models achieving similar performance levels, ranging from 80.1% to 80.6% on Movielens-gen. Even after deploying MLoRA, the performance remains in the range of 80.3% to 80.8%. This is because the Movielens dataset has a smaller data size, and the complexity and parameterization of the base models are already sufficient for the CTR prediction task on this dataset. Additionally, the differences between domains are relatively small, resulting in minimal improvement with MLoRA.

**Observation 3**: To demonstrate the effectiveness of MLoRA's LoRA adaptors on each domain, we analyze the performance improvement of deep models on each domain of Taobao-10 dataset. As MLP serves as the most basic model structure, and FiBiNET and DeepFM exhibit relatively lower and higher performance improvements on the Taobao-10 dataset, respectively, we select these three approaches for analysis. The results are presented in Table 3.

It can be observed that MLoRA enhances consistent improvement across all domains, with the most significant improvements observed in Domain\_1 and Domain\_6. One of the reasons is that Domain\_1 and Domain\_6 have smaller data sizes and greater dissimilarities compared with other domains. Therefore, we conclude that by introducing LoRA adaptors, we effectively alleviate the issues of data sparsity and domain diversity. Particularly, the performance improvement becomes more pronounced when these issues are severe. The same phenomenon is observed for the Amazon-6 and the Movielens-gen. Thus, it can be concluded that when the dataset size increases and the distributional differences between domains become more pronounced, MLoRA achieves better results. This highlights its ability to not only learn overall data distribution information but also capture relational information between different domains.

# 4.5 Ablation Studies

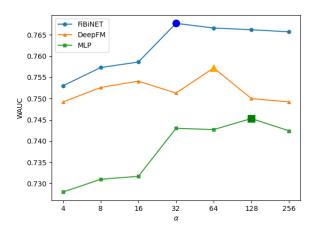


Figure 4: As temperature coefficient  $\alpha$  changes, the performance of MLP, FiBiNET, and DeepFM with MLoRA varies on the Taobao-10 dataset.

**Evaluation of** r. Note that r is able to influence the performance of MLoRA. Thus, we conduct ablation experiments by varying temperature coefficient  $\alpha$  to adjust the r value. Same as before, we take MLP, FiBiNET, and DeepFM as examples. The ablational results are shown in Figure 4. It can be observed that the performance of all three methods follows a trend of first increasing and then decreasing as  $\alpha$  is increases, which indicates the existence of an optimal  $\alpha$ . Additionally, the optimal r value corresponding to different methods varies. Specifically, MLP, DeepFM, and FiBiNET achieve optimal performance at  $\alpha$  values of 128, 64, and 32, respectively. This suggests that more complex models require larger r values, which in turn implies smaller  $\alpha$  values.

**Evaluation of domain size.** To analyze the performance of MLoRA with varying numbers of domains, we conduct ablation experiments using three multi-domain datasets: Taobao-10, Taobao-20, and Taobao-30. From Table 4, it can be observed that MLoRA achieves performance improvements of 0.49%, 0.35%, and 0.29% on

+0.78

+1.57

+0.56

+1.08

 $Avg(\Delta)$ 

+0.84

2 9 WAUC Approach 0 1 3 4 5 6 7 8 MLP 69.09 58.23 68.17 77 29 79.70 74.21 56.56 75.47 64.43 70.21 72.92 MLP+MLoRA 70.40 62.24 69.03 79.11 80.87 74.47 60.12 76.61 65.90 72.46 74.53 Λ +1.31+4.02 +0.86+1.82+1.17+0.26+3.55+1.14+1.47+2.25+1.61**FiBiNET** 71.90 63.13 70.43 79.28 81.19 78.11 68.04 82.54 65.42 76.08 76.51 FiBiNET+MLoRA 72.56 63.70 70.65 79.35 81.20 78.58 68.58 83.25 65.55 76.08 76.77 Λ +0.67+0.57+0.22+0.07+0.01+0.47+0.54+0.71+0.13+0.00+0.26DeepFM 72.44 64.27 66.60 76.46 81.36 76.56 66.85 83.53 67.34 72.69 75.01 DeepFM+MLoRA 72.81 64.41 67.21 77.80 81.65 77.41 67.11 83.65 67.58 72.79 75.66 +0.37+0.29 +0.85 +0.26 +0.24 +0.10Δ +0.14+0.61+1.34+0.12+0.65

Table 3: On Taobao-10 dataset, we obtained the AUC (%) results for each domain using the MLP, FiBiNET, and DeepFM schemes.

Table 4: The WAUC (%) results of CTR prediction on a multi-domain Taobao dataset classified by 10, 20, and 30 domains.

+0.49

+0.53

+1.45

+0.66

+0.61

+0.78

Dataset	Approach	MLP	STAR	WDL	NFM	AutoInt	PNN	DCN	FiBiNET	DeepFM	xDeepFM	Avg
	Base	72.92	75.51	73.12	76.88	75.56	76.42	72.07	76.51	75.01	75.26	74.93
Taobao-10	Base+MLoRA	74.53	76.17	73.51	77.08	75.83	76.74	72.14	76.77	75.66	75.64	75.41
	Δ	+1.61	+0.66	+0.39	+0.20	+0.27	+0.32	+0.07	+0.26	+0.65	+0.38	+0.48
	Base	77.05	77.89	77.33	78.55	77.50	79.13	79.32	79.27	79.52	79.13	78.27
Taobao-20	Base+MLoRA	77.54	78.66	77.83	78.84	78.09	79.21	79.49	79.29	79.81	79.45	78.62
	Δ	+0.49	+0.77	+0.50	+0.29	+0.59	+0.08	+0.17	+0.02	+0.29	+0.32	+0.35
	Base	76.96	77.03	77.74	78.09	78.34	78.66	75.81	76.38	76.13	79.27	77.44
Taobao-30	Base+MLoRA	77.15	78.29	77.90	78.32	78.36	78.72	76.46	76.59	76.25	79.29	77.73
	Δ	+0.19	+1.26	+0.16	+0.23	+0.02	+0.06	+0.65	+0.21	+0.12	+0.02	+0.29
-	$Avg(\Delta)$	+0.76	+0.90	+0.35	+0.24	+0.29	+0.15	+0.30	+0.16	+0.35	+0.24	+0.37

Table 5: The WAUC(%) results illustrate the impact of changing the number of neurons in hidden layers on the model's performance. Specifically, "1x," "2x," "3x," "4x," and "5x" correspond to the number of neurons in three-layer FCNs, scaled by [256, 128, 64], [512, 256, 128], [1024, 512, 256], [1024, 1024, 512] and [2048, 1024, 512], respectively.

MLoRA	1x	2x	3x	4x	5x
WAUC	75.39	75.82	75.99	76.68	76.93

Taobao-10, Taobao-20, and Taobao-30 datasets, respectively. MLoRA exhibits improvements across datasets with different numbers of domains. We also note that the improvements are more pronounced when the number of domains is smaller, while they decrease slightly as the number of domains increases. The reason may be that when the number of domains increases, the base model is more influenced by different domain data, which results in more fluctuations in our MLoRA performance. As the base model is frozen in the finetuning phase, one possible solution is to finetune the base model and LoRA network simultaneously in the future.

**Evaluation of Parameter Size of Pretrained Model.** Due to the larger scale of Amazon-6 dataset, we hypothesize that the

current three-layer fully connected neural network with hidden layer neuron quantities of (256,128,64) might not be sufficient to learn the data distribution knowledge in large-scale data. Therefore, we employ MLP+MLoRA approach and explore the impacts of changing the number of neurons on the model's performance. As shown in Table 5, it can be observed that the model's performance continuously improves along with the number of neurons in hidden layers increasing. This demonstrates that larger-scale data requires more complex and parameter-rich networks to enable the model to learn sufficiently.

# 4.6 Industry Application

Our MLoRA approach is already deployed on the Alibaba. COM e-commerce website and an online A/B testing was conducted to further verify the effectiveness. We integrated 10 core recommendation domains and utilized data from nearly 90 days (13 billion samples) for pretraining, and data from nearly 21 days (3.2 billion samples) for finetuning to build the MLoRA approach. It is a challenging endeavor to deploy the MLoRA approach in a large-scale recommendation system as it serves millions of users daily. Considering this, we only deployed the best offline method as our baseline. The temperature coefficient  $\alpha$  of MLoRA is set to 16 by default,

Table 6: The CTR, CVR and PB gains in the online recommendation system. CVR refers to the order conversion rate for recommendation scenarios. PB refers to the number of paid buyers across the entire site.

	CTR	CVR	PB
Δ	+1.49%	+3.37%	+2.71%

which means that the r value is one-sixteenth of the output dimension size, and the r value has a minimum value of one. To prevent the bottom layer parameter size of MLoRA from being too large, we increase the temperature coefficient until the parameter size of  $\bf A$  and  $\bf B$  does not exceed 1024. Ultimately, additional parameter size is only increased by 1.76%.

We carefully conducted online A/B testing from February to March 2024. MLoRA contributes to 1.49% and 3.37% increases in CTR and in the order conversion rate for recommendation scenarios, as well as a 2.71% increase in the number of paid buyers across the entire site. This indicates a significant improvement in proving the effectiveness of MLoRA in a real-world production environment. MLoRA has been launched online serving millions of users every day. **Note that, MLoRA is flexible to newly added domains.** When there is a new domain, adding and finetuning a new LoRA adaptor is very convenient.

## 5 CONCLUSION

In this paper, we have proposed a model-agnostic framework (i.e., MLoRA) for multi-domain CTR prediction. MLoRA alleviates the challenges of data sparsity and insufficient learning by configuring LoRA adaptors for each domain, while also utilizing fewer model parameters. Our proposed MLoRA approach is a general framework, which can be applied to various CTR models conveniently. Experimental results on public datasets and online A/B testing results in the Alibaba.COM e-commerce website demonstrate the superiority and feasibility of MLoRA.

In the future, we will further refine the cooperation of the base model and LoRA networks to accommodate complex issues of multidomain CTR prediction better as discussed in the evaluation of size in Section 4.5. Meanwhile, we will further enhance the MLoRA approach to address the challenges associated with data sparsity and inadequate learning capabilities.

#### REFERENCES

- [1] Armen Aghajanyan, Luke Zettlemoyer, and Sonal Gupta. 2020. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. arXiv preprint arXiv:2012.13255 (2020).
- [2] Mathieu Blondel, Akinori Fujino, Naonori Ueda, and Masakazu Ishihata. 2016. Higher-order factorization machines. Advances in Neural Information Processing Systems 29 (2016).
- [3] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. 2016. Wide & deep learning for recommender systems. In Proceedings of the 1st Workshop on Deep Learning for Recommender Systems. 7–10.
- [4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018).
- [5] Mark Dredze, Alex Kulesza, and Koby Crammer. 2010. Multi-domain learning by confidence-weighted parameter combination. *Machine Learning* 79 (2010), 123–149.
- [6] Mark Dredze, Alex Kulesza, and Koby Crammer. 2010. Multi-domain learning by confidence-weighted parameter combination. *Machine Learning* 79 (2010), 123–149.
- [7] Zhengxiao Du, Xiaowei Wang, Hongxia Yang, Jingren Zhou, and Jie Tang. 2019. Sequential scenario-specific meta learner for online recommendation. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2895–2904.
- [8] Tom Fawcett. 2006. An introduction to ROC analysis. Pattern Recognition Letters 27, 8 (2006), 861–874.
- [9] Jerome H Friedman. 2001. Greedy function approximation: a gradient boosting machine. Annals of Statistics (2001), 1189–1232.
- [10] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: a factorization-machine based neural network for CTR prediction. arXiv preprint arXiv:1703.04247 (2017).
- [11] Ruining He and Julian McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In Proceedings of the 25th International Conference on World Wide Web. 507-517.
- [12] Xiangnan He and Tat-Seng Chua. 2017. Neural factorization machines for sparse predictive analytics. In Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval. 355–364.
- [13] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. arXiv preprint arXiv:2106.09685 (2021).
- [14] Tongwen Huang, Zhiqi Zhang, and Junlin Zhang. 2019. FiBiNET: combining feature importance and bilinear feature interaction for click-through rate prediction. In Proceedings of the 13th ACM Conference on Recommender Systems. 169–177.
- [15] Mahesh Joshi, Mark Dredze, William Cohen, and Carolyn Rose. 2012. Multi-domain learning: when do domains matter?. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. 1302–1312.
- [16] Yehuda Koren. 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 426–434.
- [17] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. Computer 42, 8 (2009), 30–37.
- [18] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. arXiv preprint arXiv:1910.13461 (2019).
- [19] Chunyuan Li, Heerad Farkhoor, Rosanne Liu, and Jason Yosinski. 2018. Measuring the intrinsic dimension of objective landscapes. arXiv preprint arXiv:1804.08838 (2018).
- [20] Dongfang Li, Baotian Hu, Qingcai Chen, Xiao Wang, Quanchang Qi, Liubin Wang, and Haishan Liu. 2021. Attentive capsule network for click-through rate and conversion rate prediction in online advertising. *Knowledge-Based Systems* 211 (2021). 106522.
- [21] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. 2017. Deeper, broader and artier domain generalization. In Proceedings of the IEEE International Conference on Computer Vision. 5542–5550.
- [22] Pengcheng Li, Runze Li, Qing Da, An-Xiang Zeng, and Lijun Zhang. 2020. Improving multi-scenario learning to rank in e-commerce by exploiting task relationships in the label space. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management. 2605–2612.
- [23] Pengcheng Li, Runze Li, Qing Da, An-Xiang Zeng, and Lijun Zhang. 2020. Improving multi-scenario learning to rank in e-commerce by exploiting task relationships in the label space. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management. 2605–2612.
- [24] Jianxun Lian, Xiaohuan Zhou, Fuzheng Zhang, Zhongxia Chen, Xing Xie, and Guangzhong Sun. 2018. xdeepfm: Combining explicit and implicit feature interactions for recommender systems. In Proceedings of the 24th ACM SIGKDD

- International Conference on Knowledge Discovery & Data Mining. 1754-1763.
- [25] Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin A Raffel. 2022. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. Advances in Neural Information Processing Systems 35 (2022), 1950–1965.
- [26] Jiahui Liu, Peter Dolan, and Elin Rønby Pedersen. 2010. Personalized news recommendation based on click behavior. In Proceedings of the 15th International Conference on Intelligent User Interfaces. 31–40.
- [27] Tie-Yan Liu et al. 2009. Learning to rank for information retrieval. Foundations and Trends® in Information Retrieval 3, 3 (2009), 225–331.
- [28] Linhao Luo, Yumeng Li, Buyu Gao, Shuai Tang, Sinan Wang, Jiancheng Li, Tanchao Zhu, Jiancai Liu, Zhao Li, and Shirui Pan. 2023. MAMDR: A model agnostic learning framework for multi-domain recommendation. In 2023 IEEE 39th International Conference on Data Engineering (ICDE). IEEE, 3079–3092.
- [29] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H Chi. 2018. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 1930–1939.
- [30] Kelong Mao, Jieming Zhu, Liangcai Su, Guohao Cai, Yuru Li, and Zhenhua Dong. 2023. FinalMLP: an enhanced two-stream MLP model for CTR prediction. In Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 37. 4552–4560.
- [31] Sewon Min, Mike Lewis, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2021. Metaicl: Learning to learn in context. arXiv preprint arXiv:2110.15943 (2021).
- [32] Maxim Naumov, Dheevatsa Mudigere, Hao-Jun Michael Shi, Jianyu Huang, Narayanan Sundaraman, Jongsoo Park, Xiaodong Wang, Udit Gupta, Carole-Jean Wu, Alisson G Azzolini, et al. 2019. Deep learning recommendation model for personalization and recommendation systems. arXiv preprint arXiv:1906.00091 (2019).
- [33] Ashutosh Nayak, Mayur Garg, and Rajasekhara Reddy Duvvuru Muni. 2023. News Popularity Beyond the Click-Through-Rate for Personalized Recommendations. In Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval. 1396–1405.
- [34] Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2019. Adversarial NLI: A new benchmark for natural language understanding. arXiv preprint arXiv:1910.14599 (2019).
- [35] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. Advances in Neural Information Processing Systems 35 (2022), 27730–27744.
- [36] Changhua Pei, Yi Zhang, Yongfeng Zhang, Fei Sun, Xiao Lin, Hanxiao Sun, Jian Wu, Peng Jiang, Junfeng Ge, Wenwu Ou, et al. 2019. Personalized re-ranking for recommendation. In Proceedings of the 13th ACM Conference on Recommender Systems. 3–11.
- [37] Qi Pi, Guorui Zhou, Yujing Zhang, Zhe Wang, Lejian Ren, Ying Fan, Xiaoqiang Zhu, and Kun Gai. 2020. Search-based user interest modeling with lifelong sequential behavior data for click-through rate prediction. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management. 2685–2692.
- [38] Jiarui Qin, Weinan Zhang, Rong Su, Zhirong Liu, Weiwen Liu, Guangpeng Zhao, Hao Li, Ruiming Tang, Xiuqiang He, and Yong Yu. 2023. Learning to retrieve user behaviors for click-through rate estimation. ACM Transactions on Information Systems 41, 4 (2023), 1–31.
- [39] Jiarui Qin, Weinan Zhang, Xin Wu, Jiarui Jin, Yuchen Fang, and Yong Yu. 2020. User behavior retrieval for click-through rate prediction. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. 2347–2356.
- [40] Yanru Qu, Han Cai, Kan Ren, Weinan Zhang, Yong Yu, Ying Wen, and Jun Wang. 2016. Product-based neural networks for user response prediction. In 2016 IEEE 16th International Conference on Data Mining (ICDM). IEEE, 1149–1154.
- [41] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog* 1, 8 (2019), 9.
- [42] Steffen Rendle. 2010. Factorization machines. In 2010 IEEE International Conference on Data Mining. IEEE, 995–1000.
- [43] Reuven Rubinstein. 1999. The cross-entropy method for combinatorial and continuous optimization. Methodology and Computing in Applied Probability 1 (1999), 127–190.
- [44] Hinrich Schütze, Christopher D Manning, and Prabhakar Raghavan. 2008. Introduction to information retrieval. Vol. 39. Cambridge University Press Cambridge.
- [45] Xiang-Rong Sheng, Liqin Zhao, Guorui Zhou, Xinyao Ding, Binding Dai, Qiang Luo, Siran Yang, Jingshan Lv, Chi Zhang, Hongbo Deng, et al. 2021. One model to serve all: Star topology adaptive recommender for multi-domain ctr prediction. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management. 4104–4113.
- [46] Weiping Song, Chence Shi, Zhiping Xiao, Zhijian Duan, Yewen Xu, Ming Zhang, and Jian Tang. 2019. Autoint: Automatic feature interaction learning via self-attentive neural networks. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management. 1161–1170.

- [47] Yukihiro Tagami, Shingo Ono, Koji Yamamoto, Koji Tsukamoto, and Akira Tajima. 2013. Ctr prediction for contextual advertising: Learning-to-rank approach. In Proceedings of the Seventh International Workshop on Data Mining for Online Advertising. 1–8.
- [48] Hongyan Tang, Junning Liu, Ming Zhao, and Xudong Gong. 2020. Progressive layered extraction (ple): A novel multi-task learning (mtl) model for personalized recommendations. In Proceedings of the 14th ACM Conference on Recommender Systems. 269–278.
- [49] Ruoxi Wang, Bin Fu, Gang Fu, and Mingliang Wang. 2017. Deep & cross network for ad click predictions. In *Proceedings of the ADKDD'17*. 1–7.
- [50] Xin Wang, Peng Yang, Shaopeng Chen, Lin Liu, Lian Zhao, Jiacheng Guo, Mingming Sun, and Ping Li. 2021. Efficient learning to learn a robust CTR model for web-scale online sponsored search advertising. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management. 4203–4213.
- [51] Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, et al. 2022. Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks. arXiv preprint arXiv:2204.07705 (2022).
- [52] Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. arXiv preprint arXiv:2109.01652 (2021).
- [53] Jun Xiao, Hao Ye, Xiangnan He, Hanwang Zhang, Fei Wu, and Tat-Seng Chua. 2017. Attentional factorization machines: Learning the weight of feature interactions via attention networks. arXiv preprint arXiv:1708.04617 (2017).
- [54] Zhibo Xiao, Luwei Yang, Wen Jiang, Yi Wei, Yi Hu, and Hao Wang. 2020. Deep multi-interest network for click-through rate prediction. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management.
- [55] Zhibo Xiao, Luwei Yang, Tao Zhang, Wen Jiang, Wei Ning, and Yujiu Yang. 2024. Deep Evolutional Instant Interest Network for CTR Prediction in Trigger-Induced

- Recommendation. In Proceedings of the 17th ACM International Conference on Web Search and Data Mining.
- [56] Haizhi Yang, Tengyun Wang, Xiaoli Tang, Qianyu Li, Yueyue Shi, Siyu Jiang, Han Yu, and Hengjie Song. 2021. Multi-task learning for bias-free joint ctr prediction and market price modeling in online advertising. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management. 2291–2300.
- [57] Weinan Zhang, Jiarui Qin, Wei Guo, Ruiming Tang, and Xiuqiang He. 2021. Deep learning for click-through rate estimation. arXiv preprint arXiv:2104.10584 (2021).
- [58] Guorui Zhou, Weijie Bian, Kailun Wu, Lejian Ren, Qi Pi, Yujing Zhang, Can Xiao, Xiang-Rong Sheng, Na Mou, Xinchen Luo, et al. 2020. CAN: revisiting feature coaction for click-through rate prediction. arXiv preprint arXiv:2011.05625 (2020).
- 59] 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 Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33. 5941–5948
- [60] 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 Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Dining. 1059–1068.
- [61] 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 Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 1059–1068.
- [62] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. 2022. Domain generalization: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence 45, 4 (2022), 4396–4415.
- [63] Yunhong Zhou, Dennis Wilkinson, Robert Schreiber, and Rong Pan. 2008. Large-scale parallel collaborative filtering for the netflix prize. In Algorithmic Aspects in Information and Management: 4th International Conference, AAIM 2008, Shanghai, China, June 23-25, 2008. Proceedings 4. Springer, 337-348.