DDPO: Direct Dual Propensity Optimization for Post-Click Conversion Rate Estimation

Hongzu Su
hongzus@std.uestc.edu.cn
University of Electronic Science and Technology of China
Chengdu, China

Lichao Meng
menglc1107@gmail.com
University of Electronic Science and Technology of China
Chengdu, China

Lei Zhu
leizhu0608@gmail.com
Tongji University
Shanghai, China

Ke Lu
kel@uestc.edu.cn
University of Electronic Science and Technology of China
Chengdu, China

Jingjing Li
jjl@uestc.edu.cn
University of Electronic Science and Technology of China
Chengdu, China

ABSTRACT

In online advertising, the sample selection bias problem is a major cause of inaccurate conversion rate estimates. Current mainstream solutions only perform causality-based optimization in the click space since the conversion labels in the non-click space are absent. However, optimization for unclicked samples is equally essential because the non-click space contains more samples and user characteristics than the click space. To exploit the unclicked samples, we propose a Direct Dual Propensity Optimization (DDPO) framework to optimize the model directly in impression space with both clicked and unclicked samples. In this framework, we specifically design a click propensity network and a conversion propensity network. The click propensity network is dedicated to ensuring that optimization in the click space is unbiased. The conversion propensity network is designed to generate pseudo-conversion labels for unclicked samples, thus overcoming the challenge of absent labels in non-click space. With these two propensity networks, we are able to perform causality-based optimization in both click space and non-click space. In addition, to strengthen the causal relationship, we design two causal transfer modules for the conversion rate prediction model with the attention mechanism. The proposed framework is evaluated on five real-world public datasets and one private Tencent advertising dataset. Experimental results verify that our method is able to improve the prediction performance significantly. For instance, our method outperforms the previous state-of-the-art method by 7.0% in terms of the Area Under the Curve on the Ali-CCP dataset. Code: https://github.com/TL-UESTC/DDPO.

CCS CONCEPTS

• **Information systems → Online advertising**; **Recommender systems**.

SIGIR '24, July 14–18, 2024, Washington, DC, USA

ACM ISBN 979-8-4007-0431-4/24/07
https://doi.org/10.1145/3626772.3657817

KEYWORDS

Post-click Conversion Rate Estimation, Conversion Propensity Estimation, Impression Space Optimization

ACM Reference Format:
Hongzu Su, Lichao Meng, Lei Zhu, Ke Lu, and Jingjing Li. 2024. DDPO: Direct Dual Propensity Optimization for Post-Click Conversion Rate Estimation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '24), July 14–18, 2024, Washington, DC, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3626772.3657817

1 INTRODUCTION

Online advertising services have become an integral part of the business for mainstream Internet entities, where accurately estimating the post-click conversion rate (CVR) is a pivotal business for revenue growth. The post-click conversion rate reflects the probability that a consumer will convert (e.g., register or pay) after a click event. A typical sequence of user behavior in online advertising is formulated as "$impression \rightarrow click \rightarrow conversion$" [26], and we illustrate it in Figure 1. From this figure, the click space $\mathcal{O}$ consists of samples selectively clicked by users, and therefore its data distribution is inevitably different from the impression space $\mathcal{D}$ that contains all the samples. This phenomenon is known as sample selection bias (SSB) [10, 13, 15, 21], which leads to models trained in the click space being biased for unclicked samples [25, 27].

To mitigate the SSB problem, most of the solutions utilize the inverse propensity score (IPS) estimator [21, 26] or doubly robust (DR) estimator [2, 4, 21] to eliminate the bias between click-space training and impression-space ground truth. However, these methods perform no direct optimization on unclicked samples due to the absence of conversion labels, which inevitably leads to biases and thus degrades prediction performance [25] (see Section 2.2). Recently, Zhu et al. [27] constructed a fully converted counterfactual space and designed a factual tower and a counterfactual tower with opposite optimization targets on a shared encoding module. However, two opposite towers will introduce gradient conflicts to the shared module and thus cause performance degradation. Xu et al. [25] introduced a knowledge distillation strategy to generate

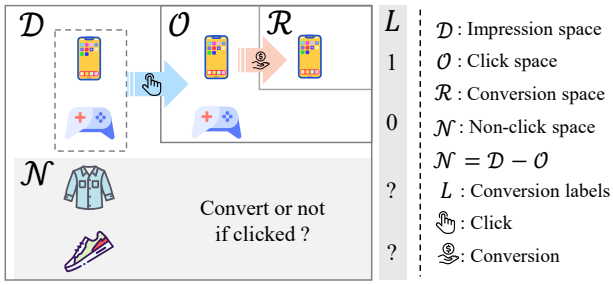


Figure 1: Illustration of user behaviors in online advertising.

pseudo-conversion labels for unclicked samples. However, knowledge distillation generally requires a generalization-capable teacher model [5, 6], which is not always available in real-world scenarios.

For better understanding, we outline three critical issues in reducing sample selection bias:

(1) **Absent Conversion Labels:** The absence of conversion labels for unclicked samples degrades CVR estimation performance and exacerbates the sample selection bias problem.
(2) **Impression Space Optimization:** Optimizing the model with both clicked and unclicked samples in impression space ensures unbiased CVR estimation, yet most previous methods do not utilize unclicked samples. Furthermore, Zhu et al. [27] have shown that failing to optimize on unclicked samples is prone to result in overestimated CVR.
(3) **Limited Causal Relationship:** The click features and CVR features contain rich causal information that improves prediction performance. However, previous methods fail to exploit feature-level causality since they formulate the CVR and click propensity networks as independent models.

To challenge these issues, we propose a novel solution to simultaneously optimize all clicked and unclicked samples in the impression space, named the Direct Dual Propensity Optimization (DDPO) framework. This framework consists of three networks: the click propensity prediction network, the conversion propensity prediction network, and the CVR prediction network. Technically, **to handle the challenge of Absent Conversion Labels**, we generate pseudo-conversion labels for unclicked samples by training the conversion propensity prediction network. It is optimized with actual conversion observations in the impression space. **To handle the challenge of Impression Space Optimization**, we first label the unclicked samples with corresponding conversion propensity scores and then perform causality-based optimization with both clicked and unclicked samples in the impression space. Furthermore, we propose a dynamic soft-labeling mechanism to enhance the generalization ability of the CVR model. **To handle the challenge of Limited Causal Relationship**, we design a click causal transfer module and a conversion causal transfer module for the CVR task with the attention mechanism. Specifically, we first compute attention masks for CVR features with click and conversion features and then fetch CVR features via attention masks.

To summarize, we list the main contributions as follows:

(1) We propose a novel framework for directly optimizing CVR in the impression space with both clicked and unclicked samples. In this framework, we introduce the concept of conversion propensity score and leverage it as the pseudo-conversion label for unclicked samples.
(2) We propose to strengthen the causal relationship between clicks and conversions by integrating feature-level information sharing into the CVR prediction task. Furthermore, we propose a dynamic soft-labeling mechanism to improve the generalization ability of the CVR model.
(3) We conduct extensive experiments on five real-world public datasets and one private Tencent dataset. Experimental results verify that our method is able to significantly outperform state-of-the-art methods by an average of 2.2% in terms of AUC.

2 PRELIMINARIES

2.1 Problem Formulation

In online advertising systems, user information, item information, and user-item interactions are all recorded in logs. To introduce the problem formulation, we first denote the set of n users by $\mathcal{U} = \{u_1, u_2, ..., u_n\}$, and the set of m items by $\mathcal{I} = \{i_1, i_2, ..., i_m\}$. The impression space is a collection of all user-item pairs, which is denoted as $\mathcal{D} = \mathcal{U} \times \mathcal{I}$. Then we collect the click status and the conversion status of each user-item pair in the impression space $\mathcal{D}$ from the recorded logs. The click matrix is denoted as $\mathbf{O} \in \{0,1\}^{n \times m}$, where $o_{u,i} \in \{0,1\}$ indicates whether user u clicked on item i. The conversion matrix is denoted as $\mathbf{R} \in \{0,1\}^{n \times m}$, where $r_{u,i} \in \{0,1\}$ indicates whether user u converted on item i. To train the CVR prediction model, we also collect user-item features and denote the feature vector of user u and item i by $x_{u,i}$.

In the CVR prediction task, the model takes the user-item feature $x_{u,i}$ as input and outputs a prediction $\hat{r}_{u,i}$. We denote the prediction matrix of all user-item pairs by $\hat{\mathbf{R}} \in \mathbb{R}^{n \times m}$, where $\hat{r}_{u,i} \in [0,1]$ represents the probability that user u will convert on item i. If the conversion matrix $\mathbf{R}$ is fully observed in both click space $\mathcal{O}$ and non-click space $\mathcal{N}$, the ideal loss function for the CVR prediction task can be expressed as:

$$\mathcal{L}_{ideal} = \mathbb{E}(\mathbf{R}, \hat{\mathbf{R}}) = \frac{1}{|\mathcal{D}|} \sum_{(u,i) \in \mathcal{D}} \delta\left(r_{u,i}, \hat{r}_{u,i}\right), \tag{1}$$

where $\delta(r_{u,i}, \hat{r}_{u,i}) = -r_{u,i} log(\hat{r}_{u,i}) - (1 - r_{u,i}) log(1 - \hat{r}_{u,i})$. However, $\mathcal{L}_{ideal}$ only represents the theoretical loss of CVR in impression space since conversion labels in non-click space cannot be observed in real-world scenarios. The bias of CVR loss between a real-world prediction model M and the ideal observation is formulated as:

$$\text{Bias}[\mathcal{L}_{\text{M}}] = |\mathcal{L}_{\text{M}} - \mathcal{L}_{ideal}|, \tag{2}$$

where $\mathcal{L}_{\text{M}}$ refers to the loss function of CVR prediction model M.

2.2 Causality-based Solutions

Causality-based methods perform CVR estimation with the help of the inverse propensity score (IPS) estimator [21, 26] or the doubly robust (DR) estimator [2, 4, 21]. Among them, the IPS-based methods involve a click propensity network to predict the click propensity score and optimize the CVR model with the following loss function:

$$\mathcal{L}_{\text{IPS}} = \frac{1}{|\mathcal{D}|} \sum_{(u,i) \in \mathcal{D}} \frac{o_{u,i} \delta(r_{u,i}, \hat{r}_{u,i})}{\hat{o}_{u,i}}, \tag{3}$$

where $\hat{o}_{u,i}$ indicates the click propensity that user u will click on item i. According to this loss function, the IPS-based methods are optimized with clicked samples and do not perform any optimization with unclicked samples. The DR-based methods add an imputation network to the IPS-based approaches to predict the loss of CVR. A typical loss function for the DR-based methods is expressed as:

$$\mathcal{L}_{\mathrm{DR}} = \frac{1}{|\mathcal{D}|} \sum_{(u,i)\in\mathcal{D}} \left(\hat{e}_{u,i} + \frac{o_{u,i}[\delta(r_{u,i}, \hat{r}_{u,i}) - \hat{e}_{u,i}]}{\hat{o}_{u,i}} \right), \tag{4}$$

where $\hat{e}_{u,i}$ refers to the imputed error predicted by the imputation network. Although the imputation network is optimized in the impression space, it is not trained with unclicked samples, and thus it also struggles to make accurate predictions.

In summary, both the IPS-based and DR-based methods fail to perform model optimization with unclicked samples due to the absence of conversion labels for the unclicked samples.

3 PROPOSED METHOD

3.1 The DDPO Framework

In our work, our aim is to mitigate the sample selection bias in post-click conversion rate estimation. Toward this goal, we design a novel framework to directly optimize the CVR prediction model with both clicked and unclicked samples in the impression space, and illustrate this framework in Figure 2. The proposed framework consists of four tasks: the click propensity prediction task, the conversion propensity prediction task, the conversion rate (CVR) prediction task and the click-through conversion rate (CTCVR) prediction task. In this framework, we specifically design a click propensity network and a conversion propensity network for the CVR prediction network. These three networks share the same network structure and are all optimized in the impression space $\mathcal{D}$. During feature encoding, two causal transfer modules separately integrate the click propensity and conversion propensity into corresponding CVR features. At the model optimization stage, we first generate pseudo-conversion labels for unclicked samples with the conversion propensity network. Then, we optimize the CVR network with inverse click propensity scores in both click space and non-click space.

3.2 Non-click Pseudo-label Generation

In real-world advertising systems, conversion labels are not available for unclicked samples, yet the unclicked samples are beneficial in mitigating the sample selection bias problem. In order to optimize the CVR model with unclicked samples, it is necessary to assign appropriate pseudo-conversion labels to unclicked samples. An intuitive practice is to set the pseudo-conversion labels of unclicked samples to 0. Unfortunately, the studies of Zhu et al. [27] and Xu et al. [25] have demonstrated that setting the conversion labels of unclicked samples to 0 degrades the prediction performance.

In order to assign pseudo-conversion labels to unclicked samples, we propose to design a conversion propensity prediction network θ_c. The conversion propensity prediction network takes user-item features as input and predicts the conversion propensity as follows:

$$r^c_{u,i} = sigmoid\left(\theta_c(x_{u,i})\right), \tag{5}$$

where $x_{u,i}$ denotes the feature vector of user u and item i, $r^c_{u,i}$ refers to the conversion propensity that user u will convert on item i. We optimize the conversion propensity network with real conversion observation $r_{u,i} \in \mathbf{R}$ in the impression space by minimizing the following loss function:

$$\begin{aligned} \mathcal{L}_c &= \frac{1}{|\mathcal{D}|} \sum_{(u,i)\in O} \delta(r_{u,i}, r^c_{u,i}) \\ &= \frac{1}{|\mathcal{D}|} \sum_{(u,i)\in\mathcal{D}} o_{u,i}\delta(r_{u,i}, r^c_{u,i}), \end{aligned} \tag{6}$$

where δ refers to the log loss, and $\delta(r_{u,i}, r^c_{u,i}) = -r_{u,i}log(r^c_{u,i}) - (1 - r_{u,i})log(1 - r^c_{u,i})$, $o_{u,i}$ indicates whether user u clicked on item i. Notice that the loss function is calculated with only clicked samples since the conversion propensity prediction network is designed to generate pseudo-conversion labels for unclicked samples.

We provide a discussion of the rationale for optimizing the conversion propensity prediction network with only the clicked samples in impression space. According to Zhu et al. [27], there are false negative samples in the non-click space, i.e., there are unclicked samples in real-world scenarios where the conversion label should be 1. To avoid optimizing potential false negative samples, we only utilize click samples with real conversion observations to train the model. During model inference, false-negative samples are predicted to carry high conversion propensity since they share similar user-item features as the converted samples. Similarly, true negative samples will be predicted to carry a low conversion propensity. Thus, the conversion propensity prediction network in our work is able to provide proper conversion labels for unclicked samples. The experimental results in Section 4.3 also demonstrate that our approach is rational and practical.

With the well-trained conversion propensity prediction network, we generate pseudo-conversion labels for all samples in the impression space and record them in matrix $\mathbf{R}_c \in \mathbb{R}^{n\times m}$. Each entry $r^c_{u,i} \in [0, 1]$ in $\mathbf{R}_c$ denotes the conversion propensity that user u will convert on item i.

3.3 Impression Space Optimization

Directly optimizing the CVR prediction model in impression space requires reasonable pseudo-labels for unclicked samples. As mentioned above, we specifically design a conversion propensity prediction network to generate pseudo-conversion labels and record them in the matrix $\mathbf{R}_c$. With the pseudo-conversion matrix $\mathbf{R}_c$, we are able to optimize the CVR model with all clicked and unclicked samples in a supervised learning manner. Following the widely studied setting [2, 15], we design a click propensity prediction network θ_o to optimize the causality-based model. The click propensity prediction network takes user-item feature $x_{u,i}$ as input and predicts the corresponding click propensity as follows:

$$\hat{o}_{u,i} = sigmoid\left(\theta_o(x_{u,i})\right), \tag{7}$$

where $\hat{o}_{u,i}$ refers to the click propensity that user u will click on item i. With the click propensity score $\hat{o}_{u,i}$, we are able to perform CVR optimization on both clicked and unclicked samples by minimizing

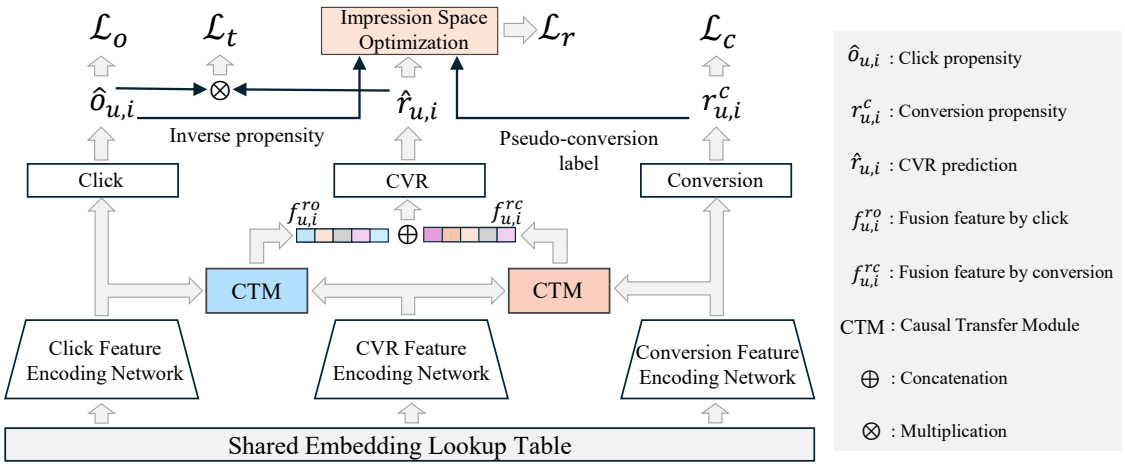


Figure 2: Illustration of the Direct Dual Propensity Optimization framework. The proposed framework consists of three networks: the click propensity prediction network, the conversion propensity prediction network, and the CVR prediction network. In this framework, the click propensity network is dedicated to ensuring that optimization in the click space is unbiased. The conversion propensity network is designed to generate pseudo-conversion labels for unclicked samples.

the following loss function:

$$\mathcal{L}_r^{naive} = \frac{1}{|\mathcal{D}|}\left(\sum_{(u,i)\in O} \frac{\delta(r_{u,i}, \hat{r}_{u,i})}{\hat{o}_{u,i}} + \sum_{(u,i)\in N} \frac{\delta(r^c_{u,i}, \hat{r}_{u,i})}{1-\hat{o}_{u,i}} \right), \tag{8}$$

where $\hat{r}_{u,i} = sigmoid\left(\theta_r\left(x_{u,i}\right)\right)$ refers to the conversion probability predicted by the CVR model θ_r.

Theoretically, the model optimized by minimizing Eq. (8) is capable of achieving excellent performance in both validation and inference. However, data distributions for validation and inference inevitably vary due to rapid changes in data over time in online advertising systems [1, 16, 17]. To alleviate this problem, we propose a dynamic soft-labeling mechanism to enhance the generalization ability of the model. Our proposed dynamic soft-labeling mechanism is represented as:

$$r^*_{u,i} = \begin{cases} r_{u,i}, & if\ r_{u,i} = 1 \\ r^c_{u,i}, & if\ r_{u,i} = 0 \end{cases}, \tag{9}$$

where $r^*_{u,i}$ refers to the soft label of sample $x_{u,i}$. $r_{u,i}$ and $r^c_{u,i}$ denote the conversion observation and the generated conversion propensity score, respectively. Based on the dynamic soft-labeling mechanism, we replace the loss function of the CVR model by:

$$\begin{aligned} \mathcal{L}_r &= \frac{1}{|\mathcal{D}|}\left(\sum_{(u,i)\in O} \frac{\delta(r^*_{u,i}, \hat{r}_{u,i})}{\hat{o}_{u,i}} + \sum_{(u,i)\in N} \frac{\delta(r^*_{u,i}, \hat{r}_{u,i})}{1-\hat{o}_{u,i}} \right) \\ &= \frac{1}{|\mathcal{D}|} \sum_{(u,i)\in \mathcal{D}} \left(\frac{o_{u,i}\delta(r^*_{u,i}, \hat{r}_{u,i})}{\hat{o}_{u,i}} + \frac{(1-o_{u,i})\delta(r^*_{u,i}, \hat{r}_{u,i})}{1-\hat{o}_{u,i}} \right). \end{aligned} \tag{10}$$

According to [27], there are potential false negative samples in the non-click space, and it is impossible to tell whether the model is unbiased or not in the presence of false negative samples. Thus, we provide a discussion of whether our method is unbiased when there are no false negative samples in the non-click space.

Theorem 1. *The CVR estimate of DDPO is unbiased in impression space when both click propensity and conversion propensity are accurately predicted, i.e.,*

$$\begin{aligned} &Bias[\mathcal{L}_r] = |\mathcal{L}_r - \mathcal{L}_{ideal}| = 0, \\ &s.t. \quad \hat{o}_{u,i} = o_{u,i}\ ,\ r^c_{u,i} = r_{u,i}. \end{aligned} \tag{11}$$

Note that according to Eq. (9), $r^*_{u,i} = r_{u,i}$ holds when $r^c_{u,i} = r_{u,i}$.

Proof.

$$\begin{aligned} Bias[\mathcal{L}_r] &= |\mathcal{L}_r - \mathcal{L}_{ideal}| \\ &= \left| \frac{1}{|\mathcal{D}|}\left(\sum_{(u,i)\in O} \frac{\delta(r^*_{u,i}, \hat{r}_{u,i})}{\hat{o}_{u,i}} + \sum_{(u,i)\in N} \frac{\delta(r^*_{u,i}, \hat{r}_{u,i})}{1-\hat{o}_{u,i}} \right) - \frac{1}{|\mathcal{D}|} \sum_{(u,i)\in \mathcal{D}} \delta(r_{u,i}, \hat{r}_{u,i}) \right| \\ &= \left| \frac{1}{|\mathcal{D}|}\left(\sum_{(u,i)\in O} \frac{\delta(r_{u,i}, \hat{r}_{u,i})}{\hat{o}_{u,i}} + \sum_{(u,i)\in N} \frac{\delta(r_{u,i}, \hat{r}_{u,i})}{1-\hat{o}_{u,i}} \right) - \frac{1}{|\mathcal{D}|} \sum_{(u,i)\in \mathcal{D}} \delta(r_{u,i}, \hat{r}_{u,i}) \right| \\ &= \left| \frac{1}{|\mathcal{D}|}\left(\sum_{(u,i)\in O} \frac{(1-\hat{o}_{u,i})\delta(r_{u,i}, \hat{r}_{u,i})}{\hat{o}_{u,i}} + \sum_{(u,i)\in N} \frac{\hat{o}_{u,i}\delta(r_{u,i}, \hat{r}_{u,i})}{1-\hat{o}_{u,i}} \right) \right| \\ &\overset{(1)}{=} \left| \frac{1}{|\mathcal{D}|}\left(\sum_{(u,i)\in O} \frac{0}{\hat{o}_{u,i}} + \sum_{(u,i)\in N} \frac{0}{1-\hat{o}_{u,i}} \right) \right| = 0, \end{aligned}$$

where (1) holds because $\hat{o}_{u,i} = o_{u,i} = 1$ in the click space O and $\hat{o}_{u,i} = o_{u,i} = 0$ in the non-click space N. □

3.4 Causal Relationship Enhancement

As mentioned above, previous methods fail to exploit the feature-level causal relationship between clicks and conversions, and thus learn a limited causal relationship. To mitigate this problem, we design two causal transfer modules to transfer the click propensity

Algorithm 1: Direct Dual Propensity Optimization

Input: The user-item feature $x_{u,i}$, click label $o_{u,i}$, conversion label $r_{u,i}$ and learning rate α.
Output: The optimal parameter θ for the multi-task model.

Initialization: *Randomly initialize the parameter* θ.
while $n <$ *MaxIter* **do**
  Sample a minibatch from the dataset;
  Estimate click propensity $\hat{o}_{u,i}$ and compute $\mathcal{L}_o$ according to Eq. (16);
  Estimate conversion propensity $r^c_{u,i}$ and compute $\mathcal{L}_c$ according to Eq. (6);
  Calculate the fusion features according to Eq. (14);
  Assign dynamic soft labels according to Eq. (9);
  Estimate CVR $\hat{r}_{u,i}$ and compute $\mathcal{L}_r$ according to Eq. (10);
  Estimate CTCVR $\hat{t}_{u,i}$ and compute $\mathcal{L}_t$ according to Eq. (17);
  Compute the overall loss function $\mathcal{L}$;
  Update the parameter θ through Adam:
    $\theta \leftarrow \text{Adam}\left(\nabla_\theta \mathcal{L}, \theta, \alpha\right)$.
end
Output optimal parameter θ.

and conversion propensity information to the CVR prediction task. The causal transfer module takes the original features as input and computes the fused features with the attention mechanism [20, 24]. Denote the input of the causal transfer module T as f^{in} and the output fusion feature as f^{out}. The process of computing fusion features is formulated as:

$$\begin{aligned} f^{out} &= T(f^{in}) \\ &= softmax(w_f)\theta_v(f^{in}) \\ &= \frac{\exp(w_f)}{\sum_{d_f}\exp(w_f)}\theta_v(f^{in}), \end{aligned} \tag{12}$$

where f^{in} and f^{out} are with the same feature dimension d_f. w_f is the weight which is expressed as:

$$w_f = \frac{< \theta_q(f^{in}), \theta_k(f^{in}) >}{\sqrt{d_f}}, \tag{13}$$

where $< \cdot, \cdot >$ denotes the dot product. $\theta_q(\cdot)$, $\theta_k(\cdot)$ and $\theta_v(\cdot)$ are three fully connected layers with the same input and output dimensions (i.e., d_f). In our work, we denote the click causal transfer module by T_o and the causal transfer module by T_c. As illustrated in Figure 2, T_o and T_c compute the fused CVR features as follows:

$$\begin{aligned} f^{ro}_{u,i} &= T_o\left(f^o_{u,i}, f^r_{u,i}\right), \\ f^{rc}_{u,i} &= T_c\left(f^c_{u,i}, f^r_{u,i}\right), \\ f^r_{u,i} &= \theta_t\left(f^{ro}_{u,i}, f^{rc}_{u,i}\right), \end{aligned} \tag{14}$$

where $f^o_{u,i}$, $f^c_{u,i}$ and $f^r_{u,i}$ refer to task features encoded by click propensity network, conversion propensity network, and CVR prediction network, respectively. These task features are all encoded from the same user-item feature $x_{u,i}$. $\theta_t(\cdot)$ is a fully connected layer. With the specifically designed causal transfer modules, the CVR network is able to exploit causal information at the feature level to strengthen the causal relationship between clicks and conversions.

Table 1: Statistics of datasets. M and K are short for millions and thousands. Conv is short for conversion.

Datasets	# Train	# Click	# Conv	# Test	# Click	# Conv
Ali-CCP	42.3M	1.6M	9K	43M	1.7M	9.4K
AE-US	20M	0.29M	7K	7.5M	0.16M	3.9K
AE-NL	12.2M	0.25M	8.9K	5.6M	0.14M	4.9K
AE-FR	18.2M	0.34M	9K	8.8M	0.2M	5.3K
AE-ES	22.3M	0.57M	12.9K	9.3M	0.27M	6.1K
Industrial	11.4M	2.8M	0.15M	3.4M	0.8M	46K

3.5 Overall Optimization Strategy

The proposed framework is constructed following the multi-task learning paradigm in our work. As illustrated in Figure 2, We optimize the conversion propensity prediction network with $\mathcal{L}_c$ and leverage it to generate pseudo-conversion labels. With the generated pseudo-labels, we are able to optimize the CVR prediction network with $\mathcal{L}_r$. Following [19, 27], we leverage the self-normalized inverse click propensity to reduce the variance of the CVR estimator. Thus, $\frac{1}{\hat{o}_{u,i}}$ and $\frac{1}{1-\hat{o}_{u,i}}$ in Eq. (10) are replaced by

$$\frac{\frac{1}{\hat{o}_{u,i}}}{\sum_{\mathcal{O}}\frac{1}{\hat{o}_{u,i}}} \text{ and } \frac{\frac{1}{1-\hat{o}_{u,i}}}{\sum_{\mathcal{N}}\frac{1}{1-\hat{o}_{u,i}}}. \tag{15}$$

In our work, the click propensity network is designed for unbiased estimation in the click space. It is optimized by

$$\mathcal{L}_o = \frac{1}{|\mathcal{D}|}\sum_{(u,i)\in\mathcal{D}} \delta\left(o_{u,i}, \hat{o}_{u,i}\right). \tag{16}$$

We also design a click-through conversion rate (CTCVR) prediction task to preserve the causal relationship between clicks and conversions, which is optimized with the following loss function:

$$\mathcal{L}_t = \frac{1}{|\mathcal{D}|}\sum_{(u,i)\in\mathcal{D}} \delta\left(r_{u,i}, \hat{t}_{u,i}\right), \tag{17}$$

where $\hat{t}_{u,i} = \hat{o}_{u,i} * \hat{r}_{u,i}$ refers to the predicted CTCVR. To optimize the complete framework, we formulate the overall optimization objective of our approach as follows:

$$\mathcal{L} = \lambda_o\mathcal{L}_o + \lambda_c\mathcal{L}_c + \lambda_r\mathcal{L}_r + \lambda_t\mathcal{L}_t, \tag{18}$$

where λ_o, λ_c, λ_r and λ_t are hyper-parameters that control the contribution of four tasks. For better understanding, we summarize the details of the overall training strategy in Algorithm 1.

4 EXPERIMENTS

In this section, we conduct extensive experiments on five large-scale public datasets and one real-world industrial dataset to study the following research questions:

RQ1: Does the DDPO improve CVR estimation performance?
RQ2: To what extent does the proposed method reduce bias?
RQ3: Is it necessary to optimize the conversion propensity network and unclicked samples?
RQ4: What is the contribution of the conversion propensity network, the causal transfer module and the dynamic soft-labeling mechanism to the performance improvement?
RQ5: What is the effect of different hyper-parameters in Eq. (18) on the model performance?

Table 2: The performance comparison of different models. All results are reported in terms of AUC. MMoE+DDPO is a combination approach formed by applying our method to the base model MMoE. The best and the second results are marked with Bold and Underline. The improvement is calculated by comparing to the best baseline on each dataset.

Methods	Ali-CCP		AE-US		AE-NL		AE-FR		AE-ES		Industrial	
	CVR	CTCVR	CVR	CTCVR	CVR	CTCVR	CVR	CTCVR	CVR	CTCVR	CVR	CTCVR
ESMM [15]	0.5879	0.5721	0.7854	0.8116	0.7650	0.8292	0.8063	0.8300	0.8091	0.8641	0.8139	0.8386
Multi-DR [26]	0.6191	0.5992	0.8065	0.8225	0.7715	0.8320	0.8107	0.8450	0.8091	0.8567	0.8132	0.8390
MRDR [4]	0.6152	0.6047	0.8132	0.8311	0.7767	0.8287	0.8115	0.8466	0.8133	0.8615	0.8133	0.8402
DR-JL [22]	0.6153	0.5962	0.8114	0.8312	0.7865	0.8296	0.8085	0.8496	0.8165	0.8640	0.8143	0.8367
ESCM^2-IPW [15]	0.6287	0.6001	0.8155	0.8238	0.7876	0.8343	0.8128	0.8296	0.8184	0.8630	0.8181	0.8398
ESCM^2-DR [15]	0.6268	0.6116	0.8148	0.8530	0.7843	0.8353	0.8109	0.8527	0.8192	0.8671	0.8163	0.8397
DCMT [27]	0.6318	0.5971	0.8156	0.8361	0.7840	0.8375	0.8130	0.8397	0.8202	0.8632	0.8199	0.8426
MMoE+DDPO	0.6731	**0.6541**	**0.8298**	**0.8747**	0.7896	0.8571	0.8162	**0.8819**	0.8296	**0.8875**	0.8282	0.8444
- Improvement	6.5%	6.9%	1.7%	2.5%	0.2%	2.3%	0.4%	3.4%	1.1%	2.3%	1.0%	0.2%
DDPO (Ours)	**0.6761**	0.6487	0.8286	0.8737	**0.7954**	**0.8627**	**0.8192**	0.8731	**0.8297**	0.8864	**0.8357**	**0.8525**
- Improvement	7.0%	6.0%	1.6%	2.4%	0.9%	3.0%	0.7%	2.4%	1.1%	2.2%	1.9%	1.2%

4.1 Datasets

To evaluate our method, we conduct extensive experiments on five large-scale public datasets and one private industrial dataset. The five public datasets are collected in the advertising system of a large-scale online shopping website, and the private dataset is collected in the affiliate advertising system of a giant Internet entity.

Public datasets. We validate our method on Ali-CCP [15] and AliExpress [12]. The Ali-CCP contains 400 thousand users and 4.3 million items, as well as over 80 million user-item interactions. The AliExpress consists of several sub-datasets with data collected from different countries. Following [27], we select sub-datasets collected from the following countries: America (AE-US), Netherlands (AE-NL), French (AE-FR) and Spain (AE-ES). The AE-US contains 500 thousand users, 1.3 million items, and over 27.5 million user-item interactions. The AE-NL contains 370 thousand users, 810 thousand items, and over 17 million user-item interactions. The AE-FR contains 570 thousand users, 1.2 million items, and over 26 million user-item interactions. The AE-ES contains 600 thousand users, 1.4 million items, and over 31 million user-item interactions. The statistics of the datasets are detailed in the Table 1.

Private dataset. We also conduct validation experiments on the affiliate advertising platform of Tencent[1]. The data is randomly sampled from the logs of our platform. Each sample is characterized by user-specific data (e.g., user ID, gender, age), item-specific data (e.g., item description, item category), and user-item interaction records (e.g., historical click records, historical conversion records). We collect 50 fields of discrete features and embed them into 400-D features for the CVR model. The click labels and conversion labels are collected by querying the user-item interaction logs, and the labels and sample features are linked by user IDs and item IDs. According to our business settings, the dataset is sampled from five consecutive days of logs in the system. The data from the first four days is utilized for training, and the data from the last day is utilized for testing. We detail the statistics of our dataset in Table 1.

4.2 Experimental Protocols

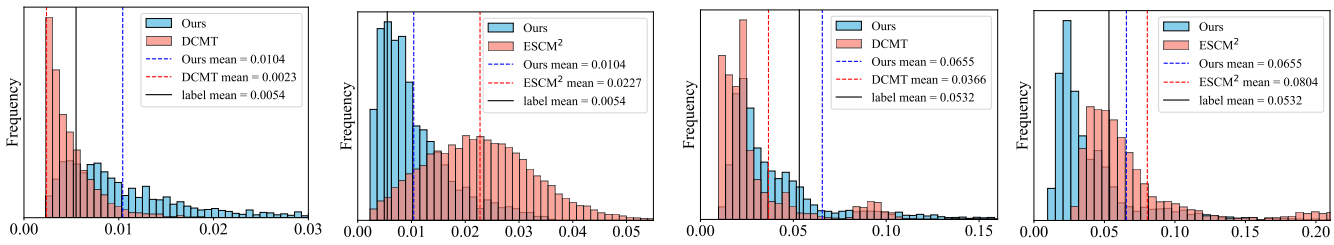
Evaluation Metric. In this paper, we evaluate the CVR estimation performance with the metric of Area Under the ROC Curve (AUC) [3]. The AUC is able to reflect the prediction accuracy of a recommendation model and is widely used in previous CVR prediction tasks [2, 15, 26]. The high quantitative results of AUC indicate the excellent predictive ability of the recommendation model. To fully demonstrate the capabilities of our method, we report the predictions of both conversion rate (CVR) and click-through conversion rate (CTCVR) in terms of AUC.

Implementation Details. In our work, we implement an MMoE model according to [14] and [21] as the base model for the compared methods. We then implement various causality-based CVR estimators as introduced in the original paper. According to Section 3.4, the causal transfer modules in our framework are implemented with multiple fully connected layers with the same input and output dimensions. The feature embedding network is implemented with multi-layer perceptrons and the prediction network is implemented with a single fully connected layer. In our work, the embedding size is set to 5 for the Ali-CCP dataset and 8 for the other datasets. We optimize all the models with the batch size set to 2048. We use Adam [7] with $\beta_1 = 0.9$, $\beta_2 = 0.999$ to optimize all models. In our work, we select all the hyper-parameters through cross validation. Hyper-parameters λ_o, λ_c, λ_r and λ_t are chosen from $\{0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1.0, 5.0\}$. Our method is implemented with Pytorch and trained on NVIDIA 3090 GPUs.

Compared Methods. In our work, we validate our approach by comparing it with the following CVR estimation methods:

(1) ESMM [15] is the first method to model CVR over the entire space, yet later studies have shown it to be biased.
(2) Multi-DR [26] is a method that optimizes the CVR prediction model with a doubly robust estimator.
(3) MRDR [4] is a method to simultaneously reduce the bias and variance of the doubly robust estimator.
(4) DR-JL [22] is a joint learning method to reduce error deviations of the imputation models.
(5) ESCM^2-IPW[21] is a method that involves the number of click samples into the inverse propensity score estimator.
(6) ESCM^2-DR[21] is a method that introduces an auxiliary imputation loss to enhance the doubly robust estimator.
(7) DCMT [27] is a method to separately optimize two opposite CVR towers in the click space and the non-click space. However, two opposite CVR towers will introduce gradient conflicts to the shared module.

[1] https://e.qq.com/dev/index.html



(a) Comparison with DCMT on Ali-CCP (b) Comparison with ESCM^2 on Ali-CCP (c) Comparison with DCMT on Industrial (d) Comparison with ESCM^2 on Industrial

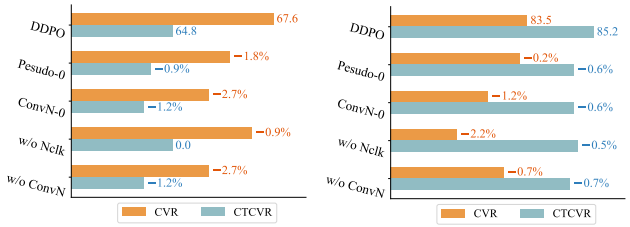
Figure 3: Comparison of CVR estimats on Ali-CPP and the industrial dataset. Note that the CVR estimation is performed in the impression space $\mathcal{D}$. The label mean refers to the average of conversion observations in the conversion matrix R. In order to make the comparison results clearer, only the predictions around the label mean are shown in this figure. Best viewed in color.

4.3 Experimental Results

RQ1: The ability to improve estimation performance. We report the results of our method and several CVR estimation methods in Table 2. According to Table 2, our method is able to outperform the previous methods by a wide margin in the CVR prediction task. From the results, our method is able to achieve the improvement of 7.0%, 1.6%, 1.4%, 0.7%, 1.1% and 1.9% when compared with the DCMT on the Ali-CCP, AE-US, AE-NL, AE-FR, AE-ES and the industrial dataset. Compared to methods that do not utilize unclicked samples, our method is able to outperform ESCM^2-DR by 7.8%, 1.2%, 1.4%, 1.0%, 1.2% and 2.4% on Ali-CCP, AE-US, AE-NL, AE-FR, AE-ES, and the industrial dataset. For the CTCVR prediction task, our method is also capable of achieving significant performance improvements. For instance, our method is able to outperform DCMT by 8.6%, 4.5%, 3.0%, 4.0%, 2.7% and 1.2% on the Ali-CCP, AE-US, AE-NL, AE-FR, AE-ES and the industrial dataset.

To fully validate our approach, we apply the proposed framework on the MMoE and report the results in Table 2. From the CVR prediction results, our method is able to achieve the improvement of 6.5%, 1.7%, 0.7%, 0.4%, 1.1% and 1.0% when compared with DCMT on the Ali-CCP, AE-US, AE-NL, AE-FR, AE-ES and the industrial dataset. Compared to ESCM^2-DR, our method is able to outperform it by 7.3%, 1.4%, 0.6%, 0.6%, 1.2% and 1.4% on the Ali-CCP, AE-US, AE-NL, AE-FR, AE-ES and the industrial dataset. For the CTCVR prediction task, our method is able to achieve the improvement of 9.5%, 4.6%, 2.3%, 5.0%, 2.8% and 0.2% when compared with the DCMT on the Ali-CCP, AE-US, AE-NL, AE-FR, AE-ES and the industrial dataset. In addition, we can observe that the combination of our method and MMoE is able to achieve higher CTCVR prediction performance than the single DDPO on Ali-CCP, AE-US, AE-FR, and AE-ES. This observation verifies that applying our approach to mainstream multi-task models is able to improve the performance.

RQ2: The ability to reduce estimation bias. We compare the prediction results of our method with previous state-of-the-art methods ESCM^2-DR [21] (a DR-based method) and DCMT [27] (a direct optimization method). The ESCM^2 is a typical causality-based method that optimizes the CVR model with only the clicked samples. The DCMT is a method that optimizes the CVR model with both clicked and unclicked samples in the impression space. We first perform CVR prediction with all three methods on the same data to validate the ability to reduce estimation bias and record their CVR estimates. We then plot the estimates of different methods in the same coordinate system and report the results in Figure 3. Since



(a) Results on Ali-CCP (b) Results on Industrial

Figure 4: Comparison of different strategies to utilize the unclicked samples. Note that the results are reported as 100 times the original AUC for clarity of presentation.

there is a large imbalance between the converted and unconverted samples, the distribution of prediction results is also unbalanced. To make the comparison results clear, we only report the predictions around the mean value in Figure 3. The label mean refers to the average of conversion observations in the conversion matrix **R**. All three methods are tested in the impression space $\mathcal{D}$.

According to Figure 3(b) and 3(d), the results of our method are closer to the label mean than ESCM^2, indicating that our method is better at reducing prediction bias. The results in Figure 3(a) show that the mean value of DCMT is closer to the label mean than our method on Ali-CCP, yet the quantitative results of DCMT in Table 2 are lower than our method. This phenomenon is because DCMT treats the labels of unclicked samples as 0 when training the factual CVR model, leading to an underestimation of CVR values. This assumption is supported by Figure 3(c), since the mean value of DCMT is also lower than the label mean on the industrial dataset. In addition, the mean value of our method is closer to the label mean than that of DCMT on the industrial dataset. This observation verifies that our method is able to exceed DCMT in the CVR prediction task. Another exciting observation is that the mean values of both our method and DCMT are closer to the label mean than ESCM^2. This observation demonstrates that optimizing the CVR model with unclicked samples is able to reduce the prediction bias and thus mitigate the sample selection bias.

RQ3: The effect of non-click space optimization. In our work, we aim to optimize the CVR prediction model with both clicked and unclicked samples in the impression space. According to Table 2 and Figure 3, our method is able to achieve more accurate CVR estimates than previous methods. To investigate the effect of optimizing the CVR model in the non-click space, we conduct

Table 3: Results of the ablation study. w/o is short for without. ConvN is short for the conversion propensity network. CTM and DSL are short for the causal transfer modules and the dynamic soft labels, respectively.

Methods	Ali-CCP		Industrial	
	CVR	CTCVR	CVR	CTCVR
DDPO w/o ConvN	0.6588	0.6407	0.8298	0.8460
DDPO w/o CTM	0.6741	0.6403	0.8278	0.8482
DDPO w/o DSL	0.6655	0.6480	0.8342	0.8466
DDPO (full model)	**0.6761**	**0.6487**	**0.8357**	**0.8525**

experiments with five optimization strategies and report the results in Figure 4. The five optimization strategies are listed as follows:

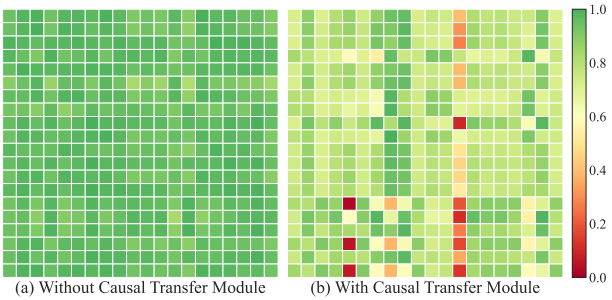
(1) DDPO: The method presented in this paper.
(2) Pseudo-0: Assume that each unclicked sample is labeled with a conversion label of 0.
(3) ConvN-0: Assume that the conversion propensity of each unclicked sample is 0.
(4) w/o Nclk: DDPO without the non-click space optimization.
(5) w/o ConvN: DDPO without the conversion propensity prediction network.

According to Figure 4, our method is able to outperform all other optimization strategies. From the prediction results on Ali-CCP, the CVR prediction is 1.8% lower than DDPO when the conversion labels of unclicked samples are assumed to be 0. This observation validates that setting the conversion labels of unclicked samples to 0 is inappropriate, which also supports that false negative samples exist in the non-click space. Similarly, assuming that the conversion propensity of unclicked samples is 0 leads to a significant performance degradation. This observation supports the idea that the conversion propensity of unclicked samples may not be 0, which is why we train the conversion propensity prediction network with only click samples in Eq. (6).

According to the results of "w/o Nclk", the performance degrades on both the Ali-CCP and the industrial dataset. We can observe that the performance degradation on Ali-CCP is less than that on the industrial dataset. The main reason for this phenomenon is that the dynamic soft-labeling mechanism in our approach allows for better model generalization and training on large-scale dataset Ali-CCP is able to fully activate the dynamic soft-labeling mechanism. The results of the ablation study in Table 3 also verify that the dynamic soft-labeling mechanism is capable of improving model performance. According to the results of "w/o ConvN", both the CVR prediction performance and the CTCVR prediction performance show significant decrease. This observation verifies that the proposed conversion propensity network is beneficial for performance improvement. Based on previous studies [25, 27] and the results in Figure 3, Figure 4, we are able to conclude that optimizing model in the non-click space is able to improve the performance.

4.4 Model Analysis

RQ4: The contribution of different components. To validate the contribution of each component, we conduct the ablation study on the public dataset Ali-CCP and our private dataset and report the experimental results in Table 3. In the ablation study, we focus on the effects of the conversion propensity network, the causal transfer modules, and the dynamic soft labels in our framework. As illustrated in Table 3, "DDPO w/o ConvN" refers to the model without the conversion propensity prediction network. Note that the conversion causal transfer module and soft labels are absent when no conversion propensity network exists in the model. "DDPO w/o CTM" refers to the model without both the click causal transfer module and conversion causal transfer module. "DDPO w/o DSL" refers to the model without the dynamic soft-labeling mechanism.



(a) Without Causal Transfer Module (b) With Causal Transfer Module

Figure 5: Comparison of similarity of CVR features. The vertical axis represents clicked samples and the horizontal axis represents unclicked samples. A low similarity indicates that the clicked and unclicked samples are easily distinguishable.

According to Table 3, the conversion propensity network contributes the most to the performance improvement. We can observe that the conversion propensity network improves the CVR performance more significantly than the CTCVR performance on the large-scale dataset Ali-CCP. This observation verifies that generating pseudo-conversion labels for unclicked samples with a conversion propensity network is able to improve the CVR prediction performance. From Table 3, the dynamic soft-labeling mechanism is able to achieve a 1.59% performance improvement on the Ali-CCP and a 0.18% performance improvement on our dataset. The main reason for this phenomenon is that our dataset is sampled over five consecutive days, in which the differences between the training and test sets are insignificant.

According to Table 3, the causal transfer modules are able to improve CVR prediction performance on both Ali-CCP and the industrial dataset. In order to intuitively demonstrate the effect of causal transfer modules, we extract features from the penultimate layer of the CVR prediction network and calculated the cosine similarity between clicked and unclicked samples. We report the comparison results in Figure 5. The results show that the similarity between clicked and unclicked CVR features is significantly lower in the approach with causal transfer modules. This observation verifies that the causal transfer module is able to transfer clicked and unclicked information to CVR features.

RQ5: The effect of different hyper-parameters in Eq. (18). Effect of hyper-parameter λ_o. The hyper-parameter λ_o controls the contribution of click propensity loss function $\mathcal{L}_o$ which is utilized to optimize the click propensity network. We conduct experiments on Ali-CPP to study the effect of λ_o and report the experimental results in Figure 6(a). According to Figure 6(a), the CVR results first show a short decrease and then slowly grow up when λ_o increases. Our method is prone to achieve better CVR performance with a greater λ_o. Form Figure 6(a), the CTCVR results continuously increase as λ_o gets larger. Similar to the results of CVR prediction,

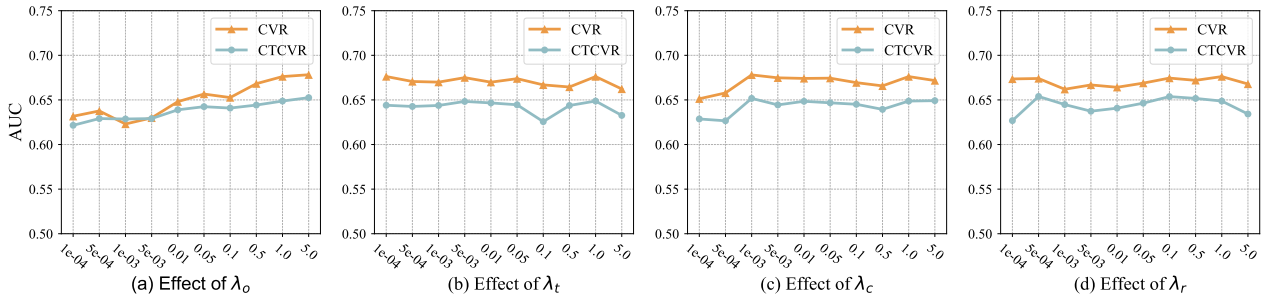


Figure 6: Parameter sensitivity analysis on Ali-CCP.

our method is prone to achieve better CTCVR performance with a greater λ_o. The results in Figure 6(a) indicate that optimization of the click propensity network is beneficial to improve the model performance. In our work, the hyper-parameter λ_o is set to 1.

Effect of hyper-parameter λ_t. The hyper-parameter λ_t controls the contribution of CTCVR loss function $\mathcal{L}_t$. We conduct experiments on Ali-CPP and report the experimental results in Figure 6(b). From the results, our method is relatively not sensitive to hyper-parameter λ_t. As the hyper-parameter λ_t grows up, the results of CVR prediction fluctuate at the best value. The results of CTCVR prediction first fluctuate and meet a rapid decrease when λ_t increases. We can also observe that the performance of our method is prone to decrease when λ_t is greater than 1. In our work, the hyper-parameter λ_t is set to 1.

Effect of hyper-parameter λ_c. The hyper-parameter λ_c controls the contribution of conversion propensity loss function $\mathcal{L}_c$ which is utilized to optimize the conversion propensity network. We conduct experiments on Ali-CPP and report the experimental results in Figure 6(c). According to the results, the CVR prediction first grows up and then shows a slow decrease as the λ_c increases. The results of CVR prediction turn to growth at $\lambda_c = 0.5$ and then begin to decline when λ_c is greater than 1. From the results, our method is prone to achieve better CTCVR prediction performance with greater λ_c. To achieve good performance in both CVR and CTCVR prediction tasks, the hyper-parameter λ_c is set to 1.

Effect of hyper-parameter λ_r. The hyper-parameter λ_r controls the contribution of CVR loss function $\mathcal{L}_r$ which is utilized to optimize the CVR prediction network. We conduct experiments on Ali-CPP and report the experimental results in Figure 6(d). According to Figure 6(d), the results of CVR first show a short decrease and then slowly grow up when λ_r increases. The results of CTCVR prediction first decrease and then turn to growth at $\lambda_r = 0.005$. The CTCVR prediction result meets its peak at $\lambda_r = 0.1$ and then begin to decline. In our work, the hyper-parameter λ_r is set to 1.

5 RELATED WORK

Eliminating the sample selection bias in recommender systems is a widely studied topic in both academia and industry [8, 23, 26]. In literature [15], researchers present an entire space method to optimize the CVR model in the impression space for the first time. However, the subsequent researchers found that this solution still suffers from bias. To achieve the unbiased estimation, Zhang et al. [26] propose a causality-based method with the inverse click propensity score and doubly robust estimator. Meanwhile, researchers propose various improvements to make the doubly robust estimator more accurate. DR-JL [22] is a joint learning method for rating prediction and error imputation. MRDR [4] and DR-BIAS [2] are methods to simultaneously optimize the bias and variance of doubly robust estimators. StableDR [11] presents a solution with a weaker reliance on extrapolation to strengthen the stability of doubly robust estimators. TDR-CL [9] is designed to simultaneously reduce the bias and variance of the doubly robust estimator when the error imputation model is inaccurate. ESCM^2[21] introduces an auxiliary imputation loss to enhance the doubly robust estimator. Li et al. [10] propose a model-agnostic balancing method to mitigate the harmful effects of unobserved confounding. CDR [18] rejects harmful imputations by scrutinizing the mean and variance of the imputations.

Recently, in order to improve the CVR prediction accuracy in recommender systems, researchers have proposed several solutions to utilize unclicked samples. DCMT [27] is a method that learns a factual tower in the click space and learns a counterfactual tower in the non-click space. In addition, researchers of this work have pointed out that there are potential false negative samples in the non-click space. UKD [25] presents a knowledge distillation strategy to generate pseudo-conversion labels for unclicked samples. This method simultaneously optimizes a teacher model and a student model with uncertainty regularization.

6 CONCLUSION

This work proposes a direct dual propensity optimization framework to mitigate the sample selection bias problem in the post-click conversion rate prediction task. The proposed framework is designed to simultaneously optimize the conversion rate estimation model with both clicked and unclicked samples. To utilize the unlabeled unclicked samples, we specifically design a conversion propensity prediction network to generate pseudo-conversion labels for the unclicked samples. We conduct extensive experiments on five public datasets and one private industrial dataset to validate our method. The experimental results verify that our method is able to significantly improve the prediction performance. Furthermore, we experimentally verify that the conversion labels of unclicked samples should not be considered as 0. The experimental results also demonstrate that optimizing CVR models in the non-click space is able to mitigate the sample selection bias.

ACKNOWLEDGMENTS

This work was supported in part by the National Natural Science Foundation of China under Grant 62176042, and in part by Sichuan Science and Technology Program under Grant 2023NSFSC0483, and in part by Tencent Marketing Solution Rhino-Bird Focused Research Program.

REFERENCES

[1] Pedro G Campos, Fernando Díez, and Iván Cantador. 2014. Time-aware recommender systems: a comprehensive survey and analysis of existing evaluation protocols. *User Modeling and User-Adapted Interaction* 24 (2014), 67–119.

[2] Quanyu Dai, Haoxuan Li, Peng Wu, Zhenhua Dong, Xiao-Hua Zhou, Rui Zhang, Rui Zhang, and Jie Sun. 2022. A generalized doubly robust learning framework for debiasing post-click conversion rate prediction. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 252–262.

[3] Tom Fawcett. 2006. An introduction to ROC analysis. *Pattern recognition letters* 27, 8 (2006), 861–874.

[4] Siyuan Guo, Lixin Zou, Yiding Liu, Wenwen Ye, Suqi Cheng, Shuaiqiang Wang, Hechang Chen, Dawei Yin, and Yi Chang. 2021. Enhanced doubly robust learning for debiasing post-click conversion rate estimation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 275–284.

[5] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* (2015).

[6] Tao Huang, Shan You, Fei Wang, Chen Qian, and Chang Xu. 2022. Knowledge distillation from a stronger teacher. *Advances in Neural Information Processing Systems* 35 (2022), 33716–33727.

[7] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.). http://arxiv.org/abs/1412.6980

[8] Haoxuan Li, Quanyu Dai, Yuru Li, Yan Lyu, Zhenhua Dong, Xiao-Hua Zhou, and Peng Wu. 2023. Multiple robust learning for recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 4417–4425.

[9] Haoxuan Li, Yan Lyu, Chunyuan Zheng, and Peng Wu. 2023. TDR-CL: Targeted Doubly Robust Collaborative Learning for Debiased Recommendations. In *The Eleventh International Conference on Learning Representations*. https://openreview.net/forum?id=EIgLnNx_lC

[10] Haoxuan Li, Yanghao Xiao, Chunyuan Zheng, and Peng Wu. 2023. Balancing unobserved confounding with a few unbiased ratings in debiased recommendations. In *Proceedings of the ACM Web Conference 2023*. 1305–1313.

[11] Haoxuan Li, Chunyuan Zheng, and Peng Wu. 2023. StableDR: Stabilized Doubly Robust Learning for Recommendation on Data Missing Not at Random. In *The Eleventh International Conference on Learning Representations*. https://openreview.net/forum?id=3VO1y5N7K1H

[12] 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.

[13] Haochen Liu, Da Tang, Ji Yang, Xiangyu Zhao, Hui Liu, Jiliang Tang, and Youlong Cheng. 2022. Rating distribution calibration for selection bias mitigation in recommendations. In *Proceedings of the ACM Web Conference 2022*. 2048–2057.

[14] 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.

[15] Xiao Ma, Liqin Zhao, Guan Huang, Zhi Wang, Zelin Hu, Xiaoqiang Zhu, and Kun Gai. 2018. Entire space multi-task model: An effective approach for estimating post-click conversion rate. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 1137–1140.

[16] Dimitrios Rafailidis and Alexandros Nanopoulos. 2015. Modeling users preference dynamics and side information in recommender systems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 46, 6 (2015), 782–792.

[17] Diego Sánchez-Moreno, Yong Zheng, and María N Moreno-García. 2020. Time-aware music recommender systems: Modeling the evolution of implicit user preferences and user listening habits in a collaborative filtering approach. *Applied Sciences* 10, 15 (2020), 5324.

[18] Zijie Song, Jiawei Chen, Sheng Zhou, Qihao Shi, Yan Feng, Chun Chen, and Can Wang. 2023. CDR: Conservative doubly robust learning for debiased recommendation. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 2321–2330.

[19] Adith Swaminathan and Thorsten Joachims. 2015. The self-normalized estimator for counterfactual learning. *advances in neural information processing systems* 28 (2015).

[20] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[21] Hao Wang, Tai-Wei Chang, Tianqiao Liu, Jianmin Huang, Zhichao Chen, Chao Yu, Ruopeng Li, and Wei Chu. 2022. Escm2: Entire space counterfactual multi-task model for post-click conversion rate estimation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 363–372.

[22] Xiaojie Wang, Rui Zhang, Yu Sun, and Jianzhong Qi. 2019. Doubly robust joint learning for recommendation on data missing not at random. In *International Conference on Machine Learning*. PMLR, 6638–6647.

[23] Zifeng Wang, Xi Chen, Rui Wen, Shao-Lun Huang, Ercan Kuruoglu, and Yefeng Zheng. 2020. Information theoretic counterfactual learning from missing-not-at-random feedback. *Advances in Neural Information Processing Systems* 33 (2020), 1854–1864.

[24] Dongbo Xi, Zhen Chen, Peng Yan, Yinger Zhang, Yongchun Zhu, Fuzhen Zhuang, and Yu Chen. 2021. Modeling the sequential dependence among audience multi-step conversions with multi-task learning in targeted display advertising. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 3745–3755.

[25] Zixuan Xu, Penghui Wei, Weimin Zhang, Shaoguo Liu, Liang Wang, and Bo Zheng. 2022. Ukd: Debiasing conversion rate estimation via uncertainty-regularized knowledge distillation. In *Proceedings of the ACM Web Conference 2022*. 2078–2087.

[26] Wenhao Zhang, Wentian Bao, Xiao-Yang Liu, Keping Yang, Quan Lin, Hong Wen, and Ramin Ramezani. 2020. Large-scale causal approaches to debiasing post-click conversion rate estimation with multi-task learning. In *Proceedings of The Web Conference 2020*. 2775–2781.

[27] Feng Zhu, Mingjie Zhong, Xinxing Yang, Longfei Li, Lu Yu, Tiehua Zhang, Jun Zhou, Chaochao Chen, Fei Wu, Guanfeng Liu, and Yan Wang. 2023. DCMT: A Direct Entire-Space Causal Multi-Task Framework for Post-Click Conversion Estimation. *2023 IEEE 39th International Conference on Data Engineering (ICDE)* (2023), 3113–3125.