



# Utilizing Non-click Samples via Semi-supervised Learning for Conversion Rate Prediction

Jiahui Huang  
University of Science and Technology  
of China  
Hefei, Anhui, China  
hjh233@mail.ustc.edu.cn

Lan Zhang\*  
University of Science and Technology  
of China  
Hefei, Anhui, China  
zhanglan@ustc.edu.cn

Junhao Wang  
University of Science and Technology  
of China  
Hefei, Anhui, China  
junhaow@mail.ustc.edu.cn

Shanyang Jiang  
University of Science and Technology  
of China  
Hefei, Anhui, China  
yang12@mail.ustc.edu.cn

Dongbo Huang  
Tencent  
Shanghai, China  
andrewhuang@tencent.com

Cheng Ding  
Tencent  
Shanghai, China  
kinding@tencent.com

Lan Xu  
Tencent  
Shanghai, China  
lanxu@tencent.com

## ABSTRACT

Conversion rate (CVR) prediction is essential in recommender systems, facilitating precise matching between recommended items and users' preferences. However, the sample selection bias (SSB) and data sparsity (DS) issues pose challenges to accurate prediction. Existing works have proposed the click-through and conversion rate (CTCVR) prediction task which models samples from exposure to "click and conversion" in entire space and incorporates multi-task learning. This approach has shown efficacy in mitigating these challenges. Nevertheless, it intensifies the false negative sample (FNS) problem. To be more specific, the CTCVR task implicitly treats all the CVR labels of non-click samples as negative, overlooking the possibility that some samples might convert if clicked. This oversight can negatively impact CVR model performance, as empirical analysis has confirmed. To this end, we advocate for discarding the CTCVR task and proposing a Non-click samples Improved Semi-supervised (NISE) method for conversion rate prediction, where the non-click samples are treated as *unlabeled*. Our approach aims to predict their probabilities of conversion if clicked, utilizing these predictions as pseudo-labels for further model training. This strategy can help alleviate the FNS problem, and direct modeling of the CVR task across the entire space also mitigates the SSB and DS challenges. Additionally, we conduct multi-task learning by introducing an auxiliary click-through rate prediction task, thereby enhancing

embedding layer representations. Our approach is applicable to various multi-task architectures. Comprehensive experiments are conducted on both public and production datasets, demonstrating the superiority of our proposed method in mitigating the FNS challenge and improving the CVR estimation. The implementation code is available at <https://github.com/Hjh233/NISE>.

## CCS CONCEPTS

• **Information systems** → **Recommender systems**; • **Computing methodologies** → *Semi-supervised learning settings*.

## KEYWORDS

Recommender Systems, Conversion Rate Prediction, False Negative Samples, Semi-supervised Learning

### ACM Reference Format:

Jiahui Huang, Lan Zhang, Junhao Wang, Shanyang Jiang, Dongbo Huang, Cheng Ding, and Lan Xu. 2024. Utilizing Non-click Samples via Semi-supervised Learning for Conversion Rate Prediction. In *18th ACM Conference on Recommender Systems (RecSys '24)*, October 14–18, 2024, Bari, Italy. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/3640457.3688151>

## 1 INTRODUCTION

Recommender systems are designed to offer customized contents that align with user preferences by effectively navigating extensive databases [10, 22, 32, 42]. These systems generally operate in two stages. First, a candidate generation model swiftly retrieves hundreds of potential items from the database using initial user data and predictive signals. Then, a ranking model refines these selections by evaluating and sorting the items based on engagement metrics such as click-through rates (CTR) [1, 25, 29] and post-click conversion rates (CVR) [11, 23, 33, 39, 40, 43]. Accurate predictions of CTR and CVR are thus essential to improve recommendation effectiveness, user experiences and financial outcomes for platforms [43]. In this work, we primarily focus on the task of CVR prediction.

\*Lan Zhang is the corresponding author.

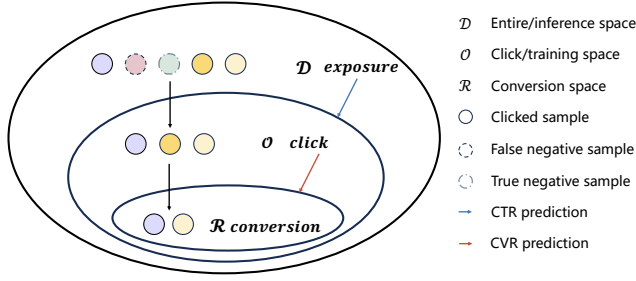
Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components 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](mailto:permissions@acm.org).

RecSys '24, October 14–18, 2024, Bari, Italy

© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 979-8-4007-0505-2/24/10

<https://doi.org/10.1145/3640457.3688151>



**Figure 1: Illustration of key challenges in CVR prediction: sample selection bias, data sparsity and false negative sample. The training space  $O$  comprises only clicked samples, while the inference space  $D$  encompasses all exposed samples.**

Users' behaviors typically follow the "*exposure*  $\rightarrow$  *click*  $\rightarrow$  *conversion*" sequence [23, 33, 40, 43], making the CVR prediction task inherently *counterfactual*. This characterization introduces three critical issues. i) *Sample selection bias (SSB)*. Conventional CVR models trained in click space  $O$  encounter challenges when inferring in entire space  $D$  due to the missing not at random (MNAR) [24, 30] nature of exposed but non-click samples. The discrepancy in distribution between  $O$  and  $D$  results in a biased CVR model. ii) *Data sparsity (DS)*. Clicked samples are notably sparse in comparison to exposed ones. For instance, in the Ali-CCP dataset<sup>1</sup>, out of the 84 million exposed samples, only 3.4 million are clicked, making up just 4% of the total samples in  $D$ . Unlike CTR models trained on the entire space  $D$ , contending with limited training samples in  $O$ , CVR models are hard to be fitted [23]. iii) *False negative sample (FNS) problem*. Due to the counterfactual nature of CVR task, non-click samples may be treated as negative since conversions are unlikely without clicking. Yet, the absence of clicks does not necessarily imply a dislike for these items [40, 43]. Instead, it could be attributed to factors such as exposure or position bias [8], where users may be unaware of the presence of exposed items. Certain samples still have the potential to convert if clicked and these samples as addressed as false negative samples. As illustrated in Figure 1, addressing the above challenges is crucial for accurate CVR prediction.

To tackle the SSB and DS issues, Ma et al. [23] propose the Entire Space Multi-task Model (ESMM) by modeling CVR task across the entire space. They utilize auxiliary CTR and click-through & conversion rate (CTCVR) tasks to enhance estimation accuracy through feature representation transfer learning. To optimize the biased CVR estimator of ESMM, causal approaches such as inverse propensity weighting (IPW) [35] and doubly robust (DR) estimator [36] are introduced. Specifically, Multi-IPW [40] and ESCM<sup>2</sup>-IPS [33] assign a propensity score to each sample in  $O$  to weight the CVR error terms. Multi-DR [40] and ESCM<sup>2</sup>-DR [33] add additional imputation models to predict errors and ensure unbiasedness when either imputation errors or learned propensities is accurate. To further address the SSB problem, Zhu et al. [43] propose a general framework involving a new counterfactual space  $N^*$  where samples have opposite labels to their corresponding factual ones to *directly* debias in  $O \cup N^*$ .

<sup>1</sup><https://tianchi.aliyun.com/dataset/408>

Despite significant advances in this field, all previously mentioned methods leverage the auxiliary CTCVR task to mitigate the issues of SSB and DS. However, due to the counterfactual nature of the CVR task, the CTCVR task shares the same label space with the CVR task. We identify that the benefits of CTCVR task comes at the cost of treating all samples in  $N$  as negative in the CVR task, thereby exacerbating the FNS problem. To validate these concerns, we have conducted empirical analysis on a benchmark dataset, which corroborates the harmful impact of this approach.

To this end, we utilize a Non-click samples Improved Semi-supervised (NISE) method for conversion rate prediction, and discard the CTCVR task in the classical multi-task learning framework. This method models all samples in the entire space  $D$ , which inherits the merits from previous works to tackle the SSB and DS challenges. More importantly, instead of outright discarding the vast number of non-click samples or categorizing them as negative, we maintain them unlabeled and first predict the conversion probabilities for samples in non-click space  $N$  if they were clicked. Then, these probabilities are treated as pseudo-labels to facilitate the learning of CVR model in a semi-supervised manner. Furthermore, we incorporate an auxiliary CTR task and apply multi-task learning to enhance the representations of the embedding layer, taking advantage of the parameter transfer [23]. During training, we adopt a dynamic task weight assignment mechanism to ensure a balanced and stable training process.

The contributions of this paper are summarized as follows:

- We conduct empirical analysis on the widely utilized CTCVR task. We identify that it could potentially aggravate the false negative sample problem because all exposed but non-clicked samples are implicitly treated as negative, thereby impairing the model's CVR prediction performance.
- Instead of utilizing the conventional CTCVR task, we treat the massive exposed yet non-click samples as *unlabeled* and innovatively devise a semi-supervised learning framework to predict their conversion rate. This entire space modeling strategy not only addresses the SSB and DS challenges but also mitigates the FNS problem.
- To leverage the power of parameter transfer, we also introduce an auxiliary CTR task and conduct multi-task learning. The proposed approach can be seamlessly applied to various multi-task learning architectures, such as shared bottom, multi-experts and mixed-experts models.
- Extensive experiments conducted on three real-world datasets validate the superiority of our approach over five state-of-the-art (SOTA) baselines in predicting CVR. Specifically, we observed a 1.11% and 2.00% relative gain in AUC on two public datasets, and a 0.70% AUC gain on a production dataset.

## 2 PRELIMINARIES

In this section, we begin by defining the focused problem in our paper. Then, we review representative methods in existing literature, laying the foundation for discussions in subsequent sections.

### 2.1 Problem Statement

This paper focuses on the conversion rate (CVR) prediction problem. We first present several important notations below. We denote

$\mathcal{U} = \{u_1, u_2, \dots, u_m\}$  as the set of  $m$  users and  $\mathcal{I} = \{i_1, i_2, \dots, i_n\}$  as the set of  $n$  items.  $\mathcal{D} = \mathcal{U} \times \mathcal{I}$  represents the entire space composed of all exposed user-item pairs.  $x_{u,i} \in \mathbb{R}^d$  is the feature vector of user  $u$  and item  $i$  and their interactions, with  $d$  denoting the feature dimension.  $\mathcal{O} = \{(u, i) | o_{u,i} = 1, (u, i) \in \mathcal{D}\}$  and  $\mathcal{R} = \{(u, i) | r_{u,i} = 1, (u, i) \in \mathcal{D}\}$  represent click space and conversion space respectively, where  $o_{u,i} \in \{0, 1\}$  indicates whether user  $u$  clicks on item  $i$  and  $r_{u,i} \in \{0, 1\}$  indicates whether item  $i$  is converted by user  $u$ .  $\mathcal{N} = \{(u, i) | o_{u,i} = 0, (u, i) \in \mathcal{D}\}$ , on the other hand, represents the non-click space.  $\mathbf{O} \in \mathbb{R}^{m \times n}$  and  $\mathbf{R} \in \mathbb{R}^{m \times n}$  are click matrix and conversion matrix of users on items respectively.

With the above definitions, we formulate the target problem.

**PROBLEM 1 (CVR PREDICTION).** *Given user set  $\mathcal{U}$ , item set  $\mathcal{I}$ , exposure space  $\mathcal{D}$ , feature vector set  $\{x_{u,i} \in \mathbb{R}^d | (u, i) \in \mathcal{D}\}$ , mapping function  $f(\cdot) : \mathbb{R}^d \rightarrow \mathbb{R}$  and fully observed conversion matrix  $\mathbf{R}$ , we seek to minimize the following ideal CVR loss*

$$\mathcal{L}_{ideal} = \frac{1}{|\mathcal{D}|} \sum_{(u,i) \in \mathcal{D}} e(r_{u,i}, \hat{r}_{u,i}) \quad (1)$$

where  $\hat{r}_{u,i} = f(x_{u,i})$  represents predicted converted probability,  $e(\cdot, \cdot)$  is the cross-entropy loss and  $e(r_{u,i}, \hat{r}_{u,i}) = -r_{u,i} \log \hat{r}_{u,i} - (1-r_{u,i}) \log(1-\hat{r}_{u,i})$ .

Nonetheless, it is unrealistic to obtain a fully observed  $\mathbf{R}$  since the conversion labels in non-click space  $\mathcal{N}$  remain unknown, and the naive way to train the CVR model is to optimize the following loss based on the observed conversion labels in click space  $\mathcal{O}$ :

$$\mathcal{L}_{naive} = \frac{1}{|\mathcal{O}|} \sum_{(u,i) \in \mathcal{O}} e(r_{u,i}, \hat{r}_{u,i}) = \frac{1}{|\mathcal{O}|} \sum_{(u,i) \in \mathcal{D}} o_{u,i} e(r_{u,i}, \hat{r}_{u,i}) \quad (2)$$

Due to the missing not at random nature [24, 30],  $\mathcal{L}_{naive}$  is not an unbiased estimator of  $\mathcal{L}_{ideal}$  [33, 40].

## 2.2 Existing CVR Prediction Methods

In this section, we provide a brief overview of representative and state-of-the-art approaches aimed at mitigating the SSB and DS problems. All methods utilize multi-task learning to enhance model training. We first introduce the general multi-task learning framework and then list debiasing methods in CVR prediction task.

The auxiliary CTR and CTCVR tasks are first introduced in [23]. The idea behind this is that both CTR and CTCVR tasks are trained with all the exposed samples in the entire space, thus SSB problem can be alleviated. Furthermore, by sharing parameters of the lookup table between CTR and CVR models, the DS issue can be mitigated. ESMM [23] adopts multi-task learning to enhance CVR modeling, with the multi-task loss function formulated as:

$$\begin{aligned} \mathcal{L}_{ESMM} &= \mathcal{L}_{CTR} + \mathcal{L}_{CTCVR} \\ &= \frac{1}{|\mathcal{D}|} \sum_{(u,i) \in \mathcal{D}} e(o_{u,i}, \hat{o}_{u,i}) \\ &\quad + \frac{1}{|\mathcal{D}|} \sum_{(u,i) \in \mathcal{D}} e(o_{u,i} \& r_{u,i}, \hat{o}_{u,i} \times \hat{r}_{u,i}) \end{aligned} \quad (3)$$

where  $\hat{o}_{u,i}$  represents predicted clicked probability and  $\hat{r}_{u,i}$  is treated as an intermediate variable.

Subsequent works [33, 40, 43] adopt similar multi-task framework and augment it with an additional tailored CVR loss to further address the SSB problem.

**2.2.1 Inverse Propensity Weighting (IPW) for CVR Prediction.** Inverse propensity weighting [35] assigns weights to click samples to achieve an unbiased estimation of the ideal CVR loss. Typically, the probability  $p(o_{u,i} = 1)$  is regarded as the true propensity score, and the output of CTR model  $\hat{o}_{u,i}$  serves as an estimate [33, 40].

The loss function is formulated as:

$$\mathcal{L}_{IPW} = \frac{1}{|\mathcal{D}|} \sum_{(u,i) \in \mathcal{D}} \frac{o_{u,i} e(r_{u,i}, \hat{r}_{u,i})}{\hat{o}_{u,i}} \quad (4)$$

**2.2.2 Doubly Robust (DR) for CVR Prediction.** Doubly robust methods [36] combine error imputation based (EIB) models with inverse propensity weighting. If either the propensity score or the error imputation model is accurate, the DR estimator is unbiased [33, 40].

The loss function is formulated as:

$$\mathcal{L}_{DR} = \frac{1}{|\mathcal{D}|} \sum_{(u,i) \in \mathcal{D}} [\hat{e}_{u,i} + \frac{o_{u,i}(e(r_{u,i}, \hat{r}_{u,i}) - \hat{e}_{u,i})}{\hat{o}_{u,i}}] \quad (5)$$

**2.2.3 Counterfactual Mechanism for CVR Prediction.** Zhu et al. [43] propose a general debiasing framework DCMT, which introduces the concept of a counterfactual space for CVR task. In the training process, a counterfactual CVR tower is introduced to predict the counterfactual CVR scores and a counterfactual regularizer is incorporated as a soft constraint.

Specifically, for a factual sample  $\langle r_{u,i} = 0, o_{u,i} = 0, x_{u,i} \rangle$  in non-click space  $\mathcal{N}$ , its corresponding counterfactual sample is  $\langle r_{u,i}^* = 1, o_{u,i} = do(o_{u,i}) = 1, x_{u,i} \rangle$ , where “do” indicates the assumption that the sample is clicked. The total loss function is formulated as:

$$\begin{aligned} \mathcal{L}_{DCMT} &= \frac{1}{|\mathcal{D}|} \left( \sum_{(u,i) \in \mathcal{O}} \frac{e(r_{u,i}, \hat{r}_{u,i})}{\hat{o}_{u,i}} + \sum_{(u,i) \in \mathcal{N}^*} \frac{e(r_{u,i}^*, \hat{r}_{u,i}^*)}{1 - \hat{o}_{u,i}} \right) \\ &\quad + \frac{\lambda}{|\mathcal{D}|} \sum_{(u,i) \in \mathcal{D}} |1 - (\hat{r}_{u,i} + \hat{r}_{u,i}^*)| \end{aligned} \quad (6)$$

## 3 DISCUSSION ON NON-CLICK SAMPLES

In this section, we analyze the widely used CTCVR task and demonstrate its limitations with empirical evidence. Subsequently, we contend that non-click samples do not necessarily indicate disinterest, and appropriate utilization of these samples can be advantageous for CVR prediction.

### 3.1 Diving into the Auxiliary CTCVR Task

To address the data sparsity and sample selection bias problem, an auxiliary CTCVR task [23] is widely employed in the field of CVR prediction [33, 39, 40, 43]. This task models the transition of each item from exposure to “click and conversion”, thus the training domain for CTCVR encompasses the entire space  $\mathcal{D}$ . We define  $t_{u,i}$  and  $\hat{t}_{u,i}$  as the actual label and predicted probability of the CTCVR task, respectively. Specifically,  $t_{u,i} = 1$  signifies that the exposed item  $i$  is clicked and converted by user  $u$ , whereas  $t_{u,i} = 0$  indicates that the item is either not clicked or clicked but not converted. The

**Table 1: Performance Comparison of CVR Models Treating Non-Click Space ( $\mathcal{N}$ ) Samples as Negative vs. Unlabeled in Ali-CCP Dataset.**

Labels in $\mathcal{N}$	AUC	Relative gain	KS	Relative gain
Negative	0.6357	–	0.1928	–
Unlabeled	0.6392	0.55%	0.1977	2.54%

output probability of CTCVR task can be decomposed as follows,

$$p(t_{u,i} = 1 | x_{u,i}) = p(o_{u,i} = 1 | x_{u,i}) \times p(r_{u,i} = 1 | x_{u,i}, o_{u,i} = 1) \quad (7)$$

which treats CVR prediction as an intermediate task.

For samples that are exposed and clicked, denoted by the click label  $o_{u,i} = 1$ , the CTCVR label  $t_{u,i} = o_{u,i}$  and  $r_{u,i}$  aligns with the CVR label. For exposed but not clicked samples, with  $o_{u,i} = 0$ , the CTCVR label is  $t_{u,i} = o_{u,i}$  and  $r_{u,i} = 0$ . Because non-clicked samples are assigned a CTCVR label of 0, optimizing the CTCVR task means that the predicted CTCVR value for non-clicked samples should trend towards zero. Additionally, since the predicted CTCVR value is the product of the output probabilities of the CTR and CVR tasks, this optimization also pushes the predicted CVR value towards zero. Therefore, optimizing the CTCVR task effectively equates to assigning a CVR label of 0 to these non-clicked samples. Previously, the CVR model was trained exclusively in click space  $\mathcal{O}$  since it was designed to model the "click  $\rightarrow$  conversion" sequence, ignoring non-clicked samples. The introduction of the CTCVR task expands the training space from just  $\mathcal{O}$  to entire space  $\mathcal{D}$ , yet we argue it treats the CVR labels of non-click samples as negative. This could potentially influence the predictive performance of the CVR model, which we will discuss in the subsequent subsection.

### 3.2 Non-click $\neq$ Disinterest

Due to the sequential nature of the "exposure  $\rightarrow$  click  $\rightarrow$  conversion" process, conversion labels remain unknown for non-click samples in  $\mathcal{N}$ . One straightforward idea is to exclude non-click samples from the training dataset. However, it has been shown to exhibit poor generalization ability since it significantly reduces training space [23]. Another idea is to treat these samples as negative, as adopted in many previous studies [4, 5, 17, 18]. However, the absence of clicks does not necessarily imply disinterest. Instead, it could be attributed to factors such as exposure or position bias [8], where users might even be unaware of the presence of exposed items. Personal user habits can also affect interaction patterns. For example, in an online Tencent recommender system, selected ads are recommended to users during the morning rush hour. However, some users rarely click on these ads because they prefer to focus on the content of their current browsing pages, such as news on current affairs. In such cases, non-click behaviors often result from a desire to avoid distractions rather than a lack of interest. If these non-click samples were to be clicked, there is still a possibility that they could eventually convert [40, 43]. Such samples are referred to as false negative samples.

To further support our argument, we conducted empirical analysis on the Ali-CCP dataset to show the harmfulness of treating all non-click samples directly as negative. The details of the dataset will be presented in Section 5.1.1. Two different methodologies are compared. In the first method, all samples in  $\mathcal{N}$  were treated as

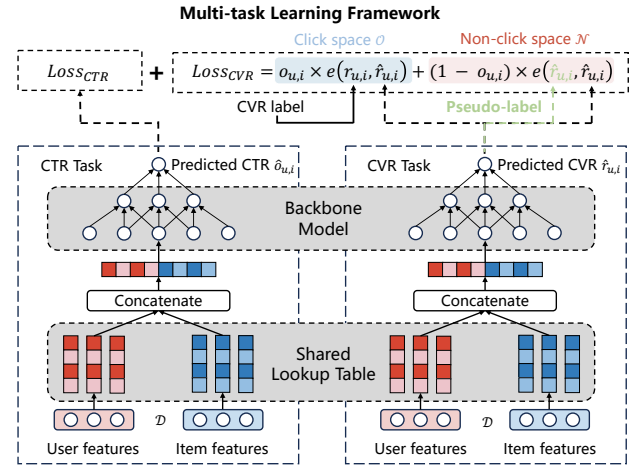
negative, whereas in the second, these instances were classified as unlabeled and modeled using semi-supervised learning techniques, which will be illustrated in Section 4.1. As shown in Table 1, the results demonstrate that treating all samples as unlabeled leads to a performance improvement of 0.55%, which is significant given the massive traffic in industrial-level recommender systems.

Moreover, we argue that while CTR task is directly related to the content (ad creatives), CVR task relates more to the conversion funnel aspects (landing page, download/installation process/order smoothness), as well as the quality and competitive pricing of the product itself. Recognizing that products with high quality and competitive pricing might not attract clicks due to less appealing creatives but are likely to convert once clicked, we discard the widely adopted CTCVR task and decouple CTR and CVR tasks, treating them as relatively independent. This strategic separation also serves to mitigate the FNS problem by dissociating the strong correlation between CTR and CVR tasks.

Grounded in the analysis above, we propose a novel semi-supervised approach to more effectively leverage the non-click samples for CVR prediction, as will be elaborated in Section 4.

## 4 METHODOLOGY

In this section, we detail the designs of our Non-click samples Improved Semi-supervised (NISE) method for conversion rate prediction. This approach allows us to model the CVR task directly across the entire space, effectively addressing the SSB and DS challenges while also mitigating the FNS problem. Additionally, we incorporate the CTR task to enhance multi-task learning. Moreover, a dynamic weighting strategy is proposed to ensure a balanced training process. The framework of NISE is illustrated in Figure 2.

**Figure 2: System Overview**

### 4.1 Semi-supervised Non-click Samples Exploitation

In CVR prediction task, we propose to keep the unknown CVR labels *unlabeled* to prevent from exacerbating the FNS problem. Inspired

by work [38], we first estimate the probabilities that non-click samples in  $\mathcal{N}$  would convert if clicked. By treating these probabilities as pseudo-labels, we model the CVR prediction task across the entire space  $\mathcal{D}$ , associated with the following loss function:

$$\mathcal{L}_{CVR} = \frac{1}{|\mathcal{D}|} \left( \sum_{(u,i) \in \mathcal{O}} e(r_{u,i}, \hat{r}_{u,i}) + \sum_{(u,i) \in \mathcal{N}} e(w_{u,i}, \hat{r}_{u,i}) \right) \quad (8)$$

where  $w_{u,i}$  is the probability that the non-click item  $i$  will be converted by user  $u$ .

In the first term, we compute the standard cross-entropy loss in click space using the ground truth labels. While in the second term, we utilize the predicted probabilities  $w_{u,i}$  as pseudo-labels to calculate the cross-entropy loss. If the predictions of  $w_{u,i}$  are accurate, this formulation serves as an unbiased estimate of the ideal loss. However, obtaining an accurate estimation of  $w_{u,i}$  remains challenging. Lacking access to additional supervised signals [39] or the means to collect a counterfactual dataset for training an additional label correction model [38], we opt to directly employ the core CVR prediction model to estimate these probabilities. Consequently, the loss in  $\mathcal{N}$  is calculated as  $\sum_{(u,i) \in \mathcal{N}} e(\hat{r}_{u,i}, \hat{r}_{u,i})$  and the CVR loss across entire space simplifies to the following:

$$\mathcal{L}_{CVR} = \frac{1}{|\mathcal{D}|} \left( \sum_{(u,i) \in \mathcal{O}} e(r_{u,i}, \hat{r}_{u,i}) + \sum_{(u,i) \in \mathcal{N}} e(\hat{r}_{u,i}, \hat{r}_{u,i}) \right) \quad (9)$$

The intuition behind this loss term is that employing the model's output as pseudo-labels causes the cross-entropy loss to degrade into the entropy of  $\hat{r}_{u,i}$ . Since entropy measures the uncertainty of random variables, during backpropagation, the model is encouraged to predict  $\hat{r}_{u,i}$  towards the boundaries, i.e. towards 0 or 1. This approach alleviates the FNS problem to some extent by identifying high-probability converting samples in  $\mathcal{N}$ , rather than categorically treating them as negative.

## 4.2 Multi-task Learning Framework

Multi-task learning (MTL) has proven effective in simultaneously learning multiple correlated tasks by sharing network parameter information across different tasks [2]. Given the strong correlation between CTR and CVR tasks, we construct a multi-task learning framework that includes an additional CTR prediction task, motivated by ESMM [23]. Specifically, the CVR model shares the same embedding lookup table with the CTR model, which maps large-scale sparse input data to low-dimensional representation vectors. This shared use of bottom-layer modules benefits from transfer learning, enabling the learning of common feature representations across these two sequential tasks [23, 26]. The MTL training loss is expressed as follows:

$$\begin{aligned} \mathcal{L}_{MTL} &= \mathcal{L}_{CTR} + \lambda \times \mathcal{L}_{CVR} \\ &= \frac{1}{|\mathcal{D}|} \sum_{(u,i) \in \mathcal{D}} e(o_{u,i}, \hat{o}_{u,i}) + \\ &\quad \frac{\lambda}{|\mathcal{D}|} \left( \sum_{(u,i) \in \mathcal{O}} e(r_{u,i}, \hat{r}_{u,i}) + \sum_{(u,i) \in \mathcal{N}} e(\hat{r}_{u,i}, \hat{r}_{u,i}) \right) \end{aligned} \quad (10)$$

where  $\lambda$  is a hyper-parameter that controls the balance between these two tasks, and further details are explained in Section 4.3.

---

### Algorithm 1: Utilizing Non-click Samples for CVR Prediction with Multi-task Learning.

---

**Input:** training samples  $\{(u, i) \in \mathcal{D}\}$ , click matrix  $\mathbf{O} \in \mathbb{R}^{m \times n}$ , conversion matrix  $\mathbf{R} \in \mathbb{R}^{m \times n}$ , training epoch  $e$ , learning rate  $\eta$ , hyper-parameters  $\alpha, \beta$

**Output:** CVR prediction model  $\theta_{CVR}$

```

1 Initialize CVR prediction model  $\theta_{CVR}$ , CTR prediction
  model  $\theta_{CTR}$ 
2 for  $k = 1$  to  $e$  do
3   for training samples  $\mathcal{B}$  in one batch do
4     Calculate  $loss_{CVR} =$ 
       $\frac{1}{|\mathcal{B}|} \left( \sum_{(u,i) \in \mathcal{B}, o_{u,i}=1} e(r_{u,i}, \hat{r}_{u,i}) + \right.$ 
       $\left. \sum_{(u,i) \in \mathcal{B}, o_{u,i}=0} e(\hat{r}_{u,i}, \hat{r}_{u,i}) \right)$ ;
5     Calculate  $loss_{CTR} = \frac{1}{|\mathcal{B}|} \sum_{(u,i) \in \mathcal{B}} e(o_{u,i}, \hat{o}_{u,i})$ ;
6     Calculate  $weight = \alpha \times \min\{\frac{loss_{CTR}}{loss_{CVR}}, \beta\}$ ;
7     Calculate MTL loss
       $loss_{MTL} = loss_{CTR} + weight \times loss_{CVR}$ ;
8      $\theta_{CVR} \leftarrow \theta_{CVR} - \eta \nabla_{\theta_{CVR}} loss_{MTL}$ ;
9      $\theta_{CTR} \leftarrow \theta_{CTR} - \eta \nabla_{\theta_{CTR}} loss_{MTL}$ ;
10 return trained CVR model  $\theta_{CVR}$ 
```

---

## 4.3 Dynamic Weighting Strategy

In practice, we observe that the training loss for CVR is consistently one to two orders of magnitude smaller than for CTR in the Ali-CCP dataset. This is likely due to the substantial difference in the size of positive samples: positive samples for the CVR task are only 0.5% of those in CTR task. Such a discrepancy in loss values can cause severe training imbalances, potentially allowing the CTR task to dominate training and hinder effective parameter updates for the CVR task, as highlighted in previous studies [9, 16].

While a universal hyper-parameter to balance the losses of CTR and CVR tasks offers a basic solution, it may be too coarse given the variability in positive sample sizes and loss values across different batches. To address this, we propose a more nuanced weighting strategy that dynamically assigns weights to each task based on their forward loss per batch, aiming for a balanced training process.

Specifically, we introduce two hyper-parameters,  $\alpha$  and  $\beta$ . For each mini-batch, we calculate the CVR and CTR losses during forward propagation. The weight is then calculated using the formula:

$$weight = \alpha \times \min\left\{\frac{loss_{CTR}}{loss_{CVR}}, \beta\right\} \quad (11)$$

The hyper-parameter  $\alpha$  is used to adjust the relative magnitudes of CVR and CTR loss, while  $\beta$  is introduced to prevent the weights from reaching excessively high values in cases where the CVR loss is extremely small, which could lead to instability during model training. Detailed training strategy is outlined in algorithm 1.

## 5 EVALUATION

We conduct extensive experiments to validate the effectiveness of our proposed method and address the following research questions:

**RQ1:** How does the proposed approach perform compared with representative or state-of-the-art baselines in CVR prediction.

**RQ2:** Does the proposed method still work when integrated into various multi-task learning frameworks?

**Table 2: Comparisons of the evaluation results of the proposed model and baselines on public datasets.**

Backbone	Method	Ali-CCP			Kuaipure		
		AUC $\uparrow$	LogLoss $\downarrow$	KS $\uparrow$	AUC $\uparrow$	LogLoss $\downarrow$	KS $\uparrow$
MLP	ESMM	0.6287 $\pm$ 0.0024	0.0113 $\pm$ 0.0012	0.1874 $\pm$ 0.0052	0.8431 $\pm$ 0.0029	0.0719 $\pm$ 0.0022	0.5558 $\pm$ 0.0025
	MMoE	0.6216 $\pm$ 0.0025	0.0108 $\pm$ 0.0014	0.1758 $\pm$ 0.0043	0.8246 $\pm$ 0.0034	0.0782 $\pm$ 0.0029	0.5230 $\pm$ 0.0061
	ESCM <sup>2</sup> -IPS	<u>0.6411<math>\pm</math>0.0023</u>	<u>0.0092<math>\pm</math>0.0018</u>	<u>0.2034<math>\pm</math>0.0032</u>	<u>0.8449<math>\pm</math>0.0018</u>	0.0744 $\pm$ 0.0011	0.5513 $\pm$ 0.0017
	ESCM <sup>2</sup> -DR	0.6182 $\pm$ 0.0120	0.0119 $\pm$ 0.0020	0.1700 $\pm$ 0.0161	0.7986 $\pm$ 0.0182	0.1144 $\pm$ 0.0288	0.4661 $\pm$ 0.0541
	DCMT	0.6407 $\pm$ 0.0024	0.0101 $\pm$ 0.0011	0.2000 $\pm$ 0.0056	0.8443 $\pm$ 0.0018	0.0739 $\pm$ 0.0033	0.5524 $\pm$ 0.0070
	NISE	<b>0.6498<math>\pm</math>0.0038</b>	<b>0.0024<math>\pm</math>0.0001</b>	<b>0.2137<math>\pm</math>0.0068</b>	<b>0.8622<math>\pm</math>0.0021</b>	<b>0.0641<math>\pm</math>0.0005</b>	<b>0.5785<math>\pm</math>0.0023</b>
DeepFM	ESMM	0.6306 $\pm$ 0.0031	<u>0.0094<math>\pm</math>0.0009</u>	0.1874 $\pm$ 0.0060	<u>0.8496<math>\pm</math>0.0031</u>	0.0695 $\pm$ 0.0047	<u>0.5618<math>\pm</math>0.0068</u>
	MMoE	0.6192 $\pm$ 0.0025	0.0104 $\pm$ 0.0019	0.1728 $\pm$ 0.0032	0.8263 $\pm$ 0.0058	0.0747 $\pm$ 0.0038	0.5217 $\pm$ 0.0111
	ESCM <sup>2</sup> -IPS	0.6394 $\pm$ 0.0023	0.0104 $\pm$ 0.0003	0.1986 $\pm$ 0.0025	0.8480 $\pm$ 0.0038	0.0764 $\pm$ 0.0061	0.5543 $\pm$ 0.0108
	ESCM <sup>2</sup> -DR	0.6003 $\pm$ 0.0059	0.0131 $\pm$ 0.0018	0.1452 $\pm$ 0.0106	0.8037 $\pm$ 0.1007	0.1233 $\pm$ 0.0914	0.4863 $\pm$ 0.1662
	DCMT	<u>0.6420<math>\pm</math>0.0005</u>	0.0090 $\pm$ 0.0011	<u>0.2094<math>\pm</math>0.0028</u>	0.8494 $\pm$ 0.0024	<u>0.0692<math>\pm</math>0.0072</u>	0.5602 $\pm$ 0.0051
	NISE	<b>0.6487<math>\pm</math>0.0016</b>	<b>0.0023<math>\pm</math>0.0001</b>	<b>0.2153<math>\pm</math>0.0065</b>	<b>0.8653<math>\pm</math>0.0013</b>	<b>0.0639<math>\pm</math>0.0017</b>	<b>0.5827<math>\pm</math>0.0030</b>
DCNV2	ESMM	0.6280 $\pm$ 0.0035	<u>0.0096<math>\pm</math>0.0006</u>	0.1855 $\pm$ 0.0037	0.8423 $\pm$ 0.0053	0.0751 $\pm$ 0.0039	0.5547 $\pm$ 0.0110
	MMoE	0.6250 $\pm$ 0.0031	0.0113 $\pm$ 0.0017	0.1787 $\pm$ 0.0027	0.8308 $\pm$ 0.0040	0.0756 $\pm$ 0.0046	0.5310 $\pm$ 0.0086
	ESCM <sup>2</sup> -IPS	0.6425 $\pm$ 0.0014	0.0106 $\pm$ 0.0010	<u>0.2089<math>\pm</math>0.0031</u>	0.8438 $\pm$ 0.0028	<u>0.0724<math>\pm</math>0.0019</u>	0.5493 $\pm$ 0.0054
	ESCM <sup>2</sup> -DR	0.6208 $\pm$ 0.0055	0.0120 $\pm$ 0.0024	0.1758 $\pm$ 0.0078	0.5889 $\pm$ 0.0575	0.6490 $\pm$ 0.2225	0.1396 $\pm$ 0.0785
	DCMT	<u>0.6431<math>\pm</math>0.0034</u>	0.0110 $\pm$ 0.0011	0.2071 $\pm$ 0.0058	<u>0.8463<math>\pm</math>0.0031</u>	0.0750 $\pm$ 0.0036	<u>0.5553<math>\pm</math>0.0057</u>
	NISE	<b>0.6520<math>\pm</math>0.0044</b>	<b>0.0024<math>\pm</math>0.0001</b>	<b>0.2188<math>\pm</math>0.0077</b>	<b>0.8641<math>\pm</math>0.0016</b>	<b>0.0629<math>\pm</math>0.0011</b>	<b>0.5781<math>\pm</math>0.0039</b>

The best results are highlighted in bold and the best baselines are underlined.  $\uparrow$  means the higher the metric the better while  $\downarrow$  indicates the lower the metric the better.

**RQ3:** How does the dynamic weighting strategy perform compared to alternative ones?

**RQ4:** How does each component contribute to the framework.

**RQ5:** How do hyper-parameters influence the performance of NISE.

**RQ6:** How does negative sampling influence the model performance of SOTA baselines and our method.

## 5.1 Experimental Setup

**5.1.1 Datasets.** To evaluate the prediction performance of our proposed NISE method and other baselines, we conduct comprehensive experiments on both public and production dataset collected from industrial platforms.

- **Public dataset:** The Ali-CCP (Alibaba Click and Conversion Prediction) dataset [23] is a benchmark dataset for conversion rate prediction, collected from traffic logs in Taobao platform. KuaiRand-Pure is an unbiased sequential recommendation dataset [13] collected from the recommendation logs of the video-sharing mobile app, Kuaishou.
- **Production dataset:** The production dataset is collected from traffic logs in Tencent platform.

The statistics of three datasets are shown in Table 3.

**Table 3: Statistics of experimental datasets**

Dataset	#Feature	#Exposure	#Click	#Conversion
Ali-CCP	33	84M	3.4M	18k
KuaiPure	41	1.4M	0.7M	22k
Production	20	2.3M	130k	51k

**Table 4: Comparisons of the evaluation results of the proposed model and baselines on production datasets.**

Method	Production		
	AUC $\uparrow$	LogLoss $\downarrow$	KS $\uparrow$
ESMM	<u>0.8120<math>\pm</math>0.0010</u>	<u>0.0867<math>\pm</math>0.0001</u>	0.4971 $\pm$ 0.0011
MMoE	0.7923 $\pm$ 0.0156	0.0905 $\pm$ 0.0027	0.4720 $\pm$ 0.0181
ESCM <sup>2</sup> -IPS	0.8107 $\pm$ 0.0012	0.0869 $\pm$ 0.0001	0.4951 $\pm$ 0.0023
ESCM <sup>2</sup> -DR	0.7314 $\pm$ 0.0381	0.4713 $\pm$ 0.0695	0.3687 $\pm$ 0.0547
DCMT	0.8106 $\pm$ 0.0010	0.0868 $\pm$ 0.0001	<u>0.4972<math>\pm</math>0.0019</u>
NISE	<b>0.8172<math>\pm</math>0.0039</b>	<b>0.0861<math>\pm</math>0.0001</b>	<b>0.5037<math>\pm</math>0.0014</b>

The best results are highlighted in bold and the best baselines are underlined.  $\uparrow$  means the higher the metric the better while  $\downarrow$  indicates the lower the metric the better.

**5.1.2 Evaluation Metrics.** Following previous works, we primarily use the AUC score (Area Under the ROC Curve) and Logloss to evaluate the CVR prediction performance. It has been widely recognized that even a 0.1% increase in the AUC score is significant. Additionally, we calculate and report the KS (Kolmogorov-Smirnov) score score to assess the model’s ability to discriminate between positive and negative samples.

**5.1.3 Backbone Models.** We choose three classical models as backbone models for classification<sup>2</sup>:

- **MLP:** the fully connected neural networks.

<sup>2</sup>If not directly specified, MLP is utilized as the backbone model.



- **DeepFM** [14]: a model that combines the factorization machines and deep neural networks to emphasize both low and high order feature interactions.
- **DCNV2** [34]: a model that utilizes cross layers to learn explicit feature interactions and combines deep neural networks to learn implicit interactions.

**5.1.4 Comparison Baselines.** Following [43], we compare our method with five baseline models in three groups described as follows: (i) Parallel MTL Baselines: **ESMM** [23]; (ii) Multi-gate MTL Baselines: **MMoE** [22] (iii) Causal Baselines: **ESCM<sup>2</sup>-IPW** [33], **ESCM<sup>2</sup>-DR** [33] and **DCMT** [43].

**5.1.5 Implementation Details.** We implement all methods using a public recommendation library<sup>3</sup>. The hyper-parameters are uniformly set to ensure a fair comparison across all experiments. For both the Ali-CCP and Kuaipure datasets, the embedding dimension for each categorical feature is set to 16. In terms of architecture, for all three backbones, the MLP tower dimensions are set at [160, 80] for the Ali-CCP dataset and [512, 256, 128, 64] for the Kuaipure dataset. For the MMoE model, we configure 8 experts. The PLE model incorporates 4 shared experts and 4 task-specific experts. Following [38], we use the stacked structure and one cross-layer for DCNV2 model. Adam [19] is used as the default optimizer and the learning rate is set to be 1e-3 with weight decay being 1e-5. The batch size is set to be 2048 for all methods. Early stopping is applied to prevent overfitting. We repeat each experiment five times on a single NVIDIA 3090 GPU and report the average results.

## 5.2 Overall Performance (RQ1)

In Table 2, we present the model performance of NISE and other strong baselines across two benchmark datasets. The results demonstrate that NISE consistently outperforms SOTA methods in terms of AUC, LogLoss, and KS metrics with various backbone models. Specifically, in the Ali-CCP dataset where only 0.02% of samples are converted, NISE exhibits an average relative AUC score improvement of 1.11% compared to SOTA methods. Similarly, in the Kuaipure dataset with 1.57% converted samples, NISE achieves an average relative AUC score improvement of 2.00% compared to SOTA methods, indicating a significant performance improvement. We attribute the performance boost to the utilization of massive non-click samples from a semi-supervised perspective. This approach aids in identifying high-probability converting samples within the non-click space, effectively mitigating the FNS problem. This utilization enhances the model’s ability to recognize potential positives that previous methods might overlook, thereby increasing accuracy and robustness. Among the strong baselines, ESCM<sup>2</sup>-IPS and DCMT stand out, both leveraging the inverse propensity weighting strategy. We posit that the efficacy of this strategy partly stems from its emphasis on assigning greater weight to the loss of CVR, thereby achieving a more balanced training process as mentioned in Section 4.2. However, it’s worth noting that ESCM<sup>2</sup>-DR exhibits instability during training, a phenomenon also pointed out in prior research [43]. The evaluation results in production dataset are shown in Table 4 with similar observations with Table 2. NISE

surpasses the top-performing baseline method by 0.70% in terms of relative improvement in AUC.

## 5.3 Generalization Ability Analysis (RQ2)

In this section, we assess the NISE framework across various prevalent multi-task learning (MTL) architectures to evaluate its generalization capabilities. These architectures include the shared bottom model (ESMM [23]), the multi-experts model (MMoE [22]), and the mixed-experts model (PLE [32]).

As illustrated in Table 5, integrating NISE into these widely adopted MTL frameworks leads to significant performance gains. Specifically, we record AUC improvements of 3.65% for the Ali-CCP dataset and 2.91% for the Kuaipure dataset. These consistent enhancements in CVR model predictions significantly demonstrate the effectiveness of the NISE approach when integrated into various MTL architectures, highlighting its robustness and adaptability in enhancing predictive accuracy across different settings.

## 5.4 Weighting Strategy Comparison (RQ3)

Various weighting strategies are proposed in multi-task learning scenarios, such as dynamic task prioritization (DTP) loss in [16] and dynamic weight average (DWA) strategy in [20]. To validate the effectiveness of our proposed dynamic weighting strategy, we conduct experiments on Ali-CCP dataset using three backbones. As shown in Table 6, NISE outperforms both DTP and DWA by a large margin. DTP employs the AUC per batch to gauge the difficulty of a task. However, the weights allocated to each task are nearly equal since the AUCs for CTR and CVR tasks are similar, leading to CTR dominance. DWA assigns weights based on changes in loss between two epochs. However, compared to the ratio of the losses, the ratio of the changes in losses is closer to 1. Additionally, both tasks are assigned equal weights of 1 in the initial two epochs, maintaining CTR’s dominance early in training.

## 5.5 Ablation Study (RQ4)

In this section, we aim to identify the crucial components of our proposed method and implement three variants of NISE on Ali-CCP dataset:

- **NISE-1:** To validate the effectiveness of multi-task training, we train a single CVR model across the entire space.
- **NISE-2:** To investigate the effectiveness of the introduction of non-click samples, we model the CVR task in click space  $\mathcal{O}$  using the naive loss in Equation 2.
- **NISE-3:** To explore the effectiveness of weighting mechanism, we omit the assignment of weights per batch, and instead directly using the addition of CTR and CVR loss as the final multi-task loss.

As illustrated in Figure 3, the performance of NISE-1 experiences a notable decline, underscoring the importance of multi-task learning. Including the CTR task in training is beneficial for learning better representations in the embedding layer, which occupies a significant proportion of the model parameters. The performances of NISE-2 and NISE-3 exhibit similar trends, with a relative AUC drop of 1.5%. The absence of consideration for non-click samples results in a much smaller training space for the CVR task. Direct

<sup>3</sup><https://github.com/datawhalechina/torch-rechub>

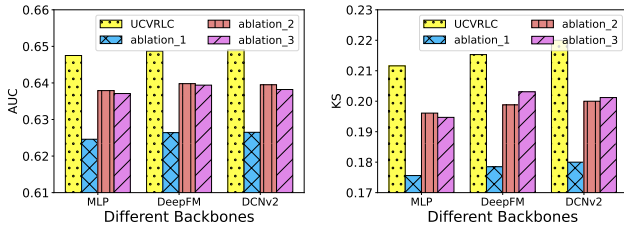
**Table 5: Overall performance improvement by applying NISE to various multi-task learning architectures.**

MTL architecture	Method	Ali-CCP			Kuaipure		
		AUC $\uparrow$	LogLoss $\downarrow$	KS $\uparrow$	AUC $\uparrow$	LogLoss $\downarrow$	KS $\uparrow$
Shared bottom	ESMM	0.6287 $\pm$ 0.0024	0.0113 $\pm$ 0.0012	0.1874 $\pm$ 0.0052	0.8431 $\pm$ 0.0029	0.0719 $\pm$ 0.0022	0.5558 $\pm$ 0.0025
	NISE	0.6498 $\pm$ 0.0038	0.0024 $\pm$ 0.0001	0.2137 $\pm$ 0.0068	0.8622 $\pm$ 0.0021	0.0641 $\pm$ 0.0005	0.5785 $\pm$ 0.0023
	Gain	3.36%	78.76%	14.03%	2.27%	10.84%	4.08%
Multi-experts	MMoE	0.6216 $\pm$ 0.0025	0.0108 $\pm$ 0.0014	0.1758 $\pm$ 0.0043	0.8246 $\pm$ 0.0034	0.0782 $\pm$ 0.0029	0.5230 $\pm$ 0.0061
	NISE	0.6468 $\pm$ 0.0026	0.0023 $\pm$ 0.0001	0.2137 $\pm$ 0.0048	0.8535 $\pm$ 0.0059	0.0702 $\pm$ 0.0031	0.5722 $\pm$ 0.0040
	Gain	4.05%	78.70%	21.56%	3.50%	10.23%	9.41%
Mixed-experts	PLE	0.6207 $\pm$ 0.0016	0.0112 $\pm$ 0.0011	0.1764 $\pm$ 0.0029	0.8413 $\pm$ 0.0034	0.0731 $\pm$ 0.0014	0.5461 $\pm$ 0.0068
	NISE	0.6449 $\pm$ 0.0004	0.0023 $\pm$ 0.0001	0.2088 $\pm$ 0.0035	0.8661 $\pm$ 0.0040	0.0649 $\pm$ 0.0018	0.5826 $\pm$ 0.0024
	Gain	3.90%	79.46%	18.37%	2.95%	11.2%	6.68%

$\uparrow$  means the higher the metric the better while  $\downarrow$  indicates the lower the metric the better.

**Table 6: Comparisons of different weighting strategies on Ali-CCP dataset.**

Backbone	Method	Ali-CCP		
		AUC $\uparrow$	LogLoss $\downarrow$	KS $\uparrow$
MLP	DTP	0.6244 $\pm$ 0.0032	0.0022 $\pm$ 0.0001	0.1762 $\pm$ 0.0050
	DWA	0.6340 $\pm$ 0.0035	0.0022 $\pm$ 0.0001	0.1890 $\pm$ 0.0056
	NISE	0.6498 $\pm$ 0.0038	0.0024 $\pm$ 0.0001	0.2137 $\pm$ 0.0068
DeepFM	DTP	0.6269 $\pm$ 0.0025	0.0022 $\pm$ 0.0001	0.1811 $\pm$ 0.0043
	DWA	0.6325 $\pm$ 0.0028	0.0021 $\pm$ 0.0001	0.1865 $\pm$ 0.0043
	NISE	0.6487 $\pm$ 0.0016	0.0023 $\pm$ 0.0001	0.2153 $\pm$ 0.0065
DCNV2	DTP	0.6233 $\pm$ 0.0024	0.0022 $\pm$ 0.0001	0.1746 $\pm$ 0.0037
	DWA	0.6337 $\pm$ 0.0023	0.0022 $\pm$ 0.0001	0.1902 $\pm$ 0.0036
	NISE	0.6520 $\pm$ 0.0044	0.0024 $\pm$ 0.0001	0.2188 $\pm$ 0.0077



(a) AUC with different backbones.

(b) KS with different backbones.

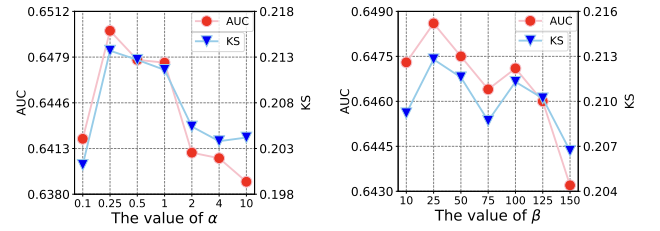
**Figure 3: Ablation study on Ali-CCP.**

discarding numerous negative and potential positive samples leads to significant information loss. Furthermore, given the observation that CVR loss is considerably smaller than that of CTR, the absence of the dynamic weighting strategy leads to severe training imbalance, resulting in poor performance of the CVR model.

## 5.6 Hyper-parameter Study (RQ5)

In this section, we explore how the hyper-parameters  $\alpha$  and  $\beta$  affect our model's prediction performance on Ali-CCP.

As is shown in Figure 4(a), when fixing the value of  $\beta$ , the AUC score undergoes significant changes with different choices of  $\alpha$ .



(a) AUC and KS with different value of  $\alpha$  and (b) AUC and KS with different value of  $\beta$ : a fixed  $\beta = 50$

**Figure 4: Hyper-parameter analysis on Ali-CCP.**

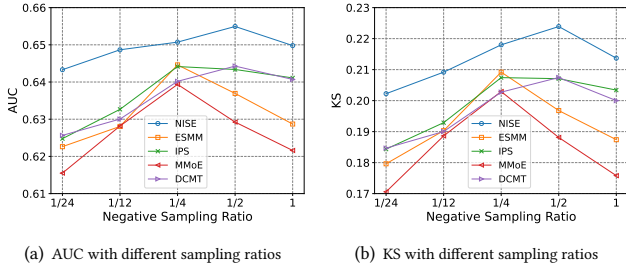
Too small value of  $\alpha$  will make CTR task dominate in the multi-task learning process, while too large value of  $\alpha$  also deteriorates CVR model performance. We presume it could be attributed to the fact that the labels for the CTR task are accurate across the entire space, whereas for the CVR task, most labels remain unknown. If too much attention is allocated to the CVR task, the potential benefit of parameter transfer introduced by the CTR task diminishes, leading to poorer learned representations. When fixing  $\alpha$ , similar observations are shown in Figure 4(b) with various value of  $\beta$ : a low  $\beta$  leads to training imbalances, while a high  $\beta$  overly prioritizes the CVR task, diminishing the CTR task's contributions.

## 5.7 Negative sampling vs Non-sampling (RQ6)

While leveraging all non-click samples and conducting multi-task learning can significantly enhance CVR model performance, it also results in substantially higher computational costs. Therefore, we evaluate the impact of negative sampling versus non-sampling on model performance using the Ali-CCP dataset. We experiment by randomly sampling  $\frac{1}{24}$ ,  $\frac{1}{12}$ ,  $\frac{1}{4}$ , and  $\frac{1}{2}$  of the non-click samples in  $\mathcal{N}$ , and compare the results with a non-sampling strategy.

Surprisingly, almost all methods show an increase followed by a decrease in performance as the proportion of sampled non-click samples increases. This pattern suggests that negative sampling can, up to a certain point, simultaneously improve the model's predictive ability and reduce training costs compared to the non-sampling strategy. We hypothesize that as the number of selected





**Figure 5: Comparison of negative sampling and non-sampling strategies on Ali-CCP.**

non-click samples initially increases, the data sparsity issue within the model is substantially alleviated, benefiting the CVR model from the enhanced dataset. However, as the proportion of sampled non-click samples continues to rise, SOTA methods may encounter more severe class imbalance issues, as an increasing number of non-click samples are treated as negative. Furthermore, our proposed method may face challenges due to the growing imbalance between labeled and unlabeled data, leading to increased uncertainty and a consequent degradation in model performance.

## 6 RELATED WORK

### 6.1 Multi-Task Learning for CVR Prediction

Due to the inherent similarity between click-through rate (CTR) and conversion rate (CVR) predictions and the sequential pattern of users' behaviors, CTR models are widely employed in CVR task. Ma et al. [23] first proposed an entire space multi-task model (ESMM) in which they incorporated post-view click-through&conversion rate (CTCVR) as an auxiliary task with CTR task to alleviate sample selection bias and data sparsity problems in CVR prediction. Wen et al. [39] then proposed post-click behavior decomposition to leverage extra purchase-related actions as supervised signals. However, the estimates of ESMM for CVR prediction were found to be higher than the ground truth [33, 40]. Additionally, ESMM overlooked the causal effect of *click*  $\rightarrow$  *conversion* and exhibited the potential independence priority (PIP) problem. To address the above problems and obtain an unbiased CVR estimation, causal inference methods [7, 31, 36] have been adopted. This involved combining inverse propensity weighting (IPW) and doubly robust (DR) methods with a multi-task learning framework [33, 40], regulating CVR predictions through a counterfactual risk minimizer. Dai et al. [11] further introduced a generalized doubly robust learning framework and proposed DR-BIAS and DR-MSE to balance the bias and variance term of DR estimator, achieving better generalization performance. Despite these improvements, existing causal inference methods primarily conducted debiasing in click space. Zhu et al. [43] addressed this limitation by introducing the concept of counterfactual samples in the non-click space. They proposed a counterfactual mechanism to directly debias in the entire space, predicting factual CVR and counterfactual CVR in respective spaces.

### 6.2 False Negatives in Recommender Systems

The prevalence of false negative samples presents a significant challenge in recommender systems, particularly in scenarios where

models are trained using implicit user feedback, such as clicks or watches. In these cases, observed interactions are labeled as positive, while unobserved ones are typically labeled as negative. Existing training approaches can be broadly categorized into two types: the first type involves sampling negative instances from unobserved interactions [12, 27, 28, 41], and the second type treats all missing data as negative [5, 6, 17]. Both approaches, however, grapple with the challenge of mislabeling potential positive samples as negative, exacerbated by exposure or position bias [8, 15, 37], which can compromise the accuracy and robustness of the model.

In the realm of sequential recommendation, Liu et al. [21] introduce UFNRec, which identifies false negative samples based on higher prediction scores in successive rounds and subsequently reverses the labels of these samples for model training. In the context of CVR prediction, where samples may take a considerable time to convert, delayed feedback [3] can lead to incorrect negative labels as well. To address this, Wang et al. [38] developed a label correction method that predicts the probability of an unobserved sample being a false negative, thereby aiming to achieve an unbiased estimate of the oracle loss. Moreover, self-selection bias can also contribute to false negative samples in CVR tasks. Previous studies [40, 43] have shown that factors such as exposure position can influence whether items are clicked. Non-click items that might otherwise convert if clicked are often overlooked. Despite the significant impact of these biases, the issue of false negative samples in CVR prediction has not yet been thoroughly examined.

## 7 CONCLUSION

In this paper, we introduce NISE, a novel approach for conversion rate prediction. While previous methods focus on addressing the SSB and DS problems, we further tackle the FNS challenge, which emerges from potential positive samples in the expansive non-click space. We argue that previously proposed CTCVR task treat the CVR labels for non-click samples as negative, compromising the model's robustness. NISE differentiates itself by treating all non-click samples as unlabeled and using the generated conversion probabilities as pseudo-labels for semi-supervised learning. The addition of a CTR task and a dynamic loss assigning strategy further enhance representation learning and ensure a balanced training process. Experimental results on both public and production datasets validate the effectiveness of our approach.

Despite its advancements, NISE has limitations that pave the way for future research. While negative sampling is promising for enhancing performance and reducing computational costs as observed in Section 5.7, its integration into multi-task learning frameworks for CVR prediction remains a challenge. Furthermore, the reliance on possibly biased prediction probabilities as soft labels introduces potential inaccuracies, highlighting the need for advanced semi-supervised learning techniques and additional supervised signals to improve training effectiveness.

## ACKNOWLEDGMENTS

Lan Zhang is the corresponding author. This research was supported by the China National Natural Science Foundation with No. 61932016 "the Fundamental Research Funds for the Central Universities" WK2150110024, and Tencent Marketing Solution Rhino-Bird Focused Research Program.

## REFERENCES

- [1] Deepak Agarwal, Rahul Agrawal, Rajiv Khanna, and Nagaraj Kota. 2010. Estimating rates of rare events with multiple hierarchies through scalable log-linear models. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. 213–222.
- [2] Rich Caruana. 1997. Multitask learning. *Machine learning* 28 (1997), 41–75.
- [3] Olivier Chapelle. 2014. Modeling delayed feedback in display advertising. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1097–1105.
- [4] Chong Chen, Weizhi Ma, Min Zhang, Chenyang Wang, Yiqun Liu, and Shaoping Ma. 2023. Revisiting negative sampling vs. non-sampling in implicit recommendation. *ACM Transactions on Information Systems* 41, 1 (2023), 1–25.
- [5] Chong Chen, Min Zhang, Chenyang Wang, Weizhi Ma, Minming Li, Yiqun Liu, and Shaoping Ma. 2019. An efficient adaptive transfer neural network for social-aware recommendation. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 225–234.
- [6] Chong Chen, Min Zhang, Yongfeng Zhang, Yiqun Liu, and Shaoping Ma. 2020. Efficient neural matrix factorization without sampling for recommendation. *ACM Transactions on Information Systems (TOIS)* 38, 2 (2020), 1–28.
- [7] Jiawei Chen, Hande Dong, Yang Qiu, Xiangnan He, Xin Xin, Liang Chen, Guli Lin, and Keping Yang. 2021. AutoDebias: Learning to debias for recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 21–30.
- [8] Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. 2023. Bias and debias in recommender system: A survey and future directions. *ACM Transactions on Information Systems* 41, 3 (2023), 1–39.
- [9] Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. 2018. GradNorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *International conference on machine learning*. PMLR, 794–803.
- [10] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM conference on recommender systems*. 191–198.
- [11] 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.
- [12] Jingtao Ding, Yuhuan Quan, Quanming Yao, Yong Li, and Depeng Jin. 2020. Simplify and robustify negative sampling for implicit collaborative filtering. *Advances in Neural Information Processing Systems* 33 (2020), 1094–1105.
- [13] Chongming Gao, Shijun Li, Yuan Zhang, Jiawei Chen, Biao Li, Wenqiang Lei, Peng Jiang, and Xiangnan He. 2022. KuaiRand: An Unbiased Sequential Recommendation Dataset with Randomly Exposed Videos. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 3953–3957.
- [14] 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).
- [15] Huifeng Guo, Jinkai Yu, Qing Liu, Ruiming Tang, and Yuzhou Zhang. 2019. PAL: a position-bias aware learning framework for CTR prediction in live recommender systems. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 452–456.
- [16] Michelle Guo, Albert Haque, De-An Huang, Serena Yeung, and Li Fei-Fei. 2018. Dynamic task prioritization for multitask learning. In *Proceedings of the European conference on computer vision (ECCV)*. 270–287.
- [17] Xiangnan He, Hanwang Zhang, Min-Yen Kan, and Tat-Seng Chua. 2016. Fast matrix factorization for online recommendation with implicit feedback. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*. 549–558.
- [18] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative filtering for implicit feedback datasets. In *2008 Eighth IEEE international conference on data mining*. Ieee, 263–272.
- [19] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).
- [20] Shikun Liu, Edward Johns, and Andrew J Davison. 2019. End-to-end multi-task learning with attention. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 1871–1880.
- [21] Xiaoyang Liu, Chong Liu, Pinzheng Wang, Rongqin Zheng, Lixin Zhang, Leyu Lin, Zhiyun Chen, and Liangliang Fu. 2023. UFNRec: Utilizing False Negative Samples for Sequential Recommendation. In *Proceedings of the 2023 SIAM International Conference on Data Mining (SDM)*. SIAM, 46–54.
- [22] 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.
- [23] 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.
- [24] Benjamin Marlin, Richard S Zemel, Sam Roweis, and Malcolm Slaney. 2012. Collaborative filtering and the missing at random assumption. *arXiv preprint arXiv:1206.5267* (2012).
- [25] H Brendan McMahan, Gary Holt, David Sculley, Michael Young, Dietmar Ebner, Julian Grady, Lan Nie, Todd Phillips, Eugene Davydov, Daniel Golovin, et al. 2013. Ad click prediction: a view from the trenches. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1222–1230.
- [26] Yabo Ni, Dan Ou, Shichen Liu, Xiang Li, Wenwu Ou, Anxiang Zeng, and Luo Si. 2018. Perceive your users in depth: Learning universal user representations from multiple e-commerce tasks. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 596–605.
- [27] Steffen Rendle and Christoph Freudenthaler. 2014. Improving pairwise learning for item recommendation from implicit feedback. In *Proceedings of the 7th ACM international conference on Web search and data mining*. 273–282.
- [28] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2012. BPR: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618* (2012).
- [29] Matthew Richardson, Ewa Dominowska, and Robert Ragno. 2007. Predicting clicks: estimating the click-through rate for new ads. In *Proceedings of the 16th international conference on World Wide Web*. 521–530.
- [30] Yuta Saito, Suguru Yaginuma, Yuta Nishino, Hayato Sakata, and Kazuhide Nakata. 2020. Unbiased recommender learning from missing-not-at-random implicit feedback. In *Proceedings of the 13th International Conference on Web Search and Data Mining*. 501–509.
- [31] Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. 2016. Recommendations as treatments: Debiasing learning and evaluation. In *international conference on machine learning*. PMLR, 1670–1679.
- [32] 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.
- [33] 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.
- [34] Ruoxi Wang, Rakesh Shivanna, Derek Cheng, Sagar Jain, Dong Lin, Lichan Hong, and Ed Chi. 2021. Dcn v2: Improved deep & cross network and practical lessons for web-scale learning to rank systems. In *Proceedings of the web conference 2021*. 1785–1797.
- [35] Xuanhui Wang, Michael Bendersky, Donald Metzler, and Marc Najork. 2016. Learning to rank with selection bias in personal search. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*. 115–124.
- [36] 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.
- [37] Xiaojie Wang, Rui Zhang, Yu Sun, and Jianzhong Qi. 2021. Combating selection biases in recommender systems with a few unbiased ratings. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. 427–435.
- [38] Yifan Wang, Peijie Sun, Min Zhang, Qinglin Jia, Jingjie Li, and Shaoping Ma. 2023. Unbiased Delayed Feedback Label Correction for Conversion Rate Prediction. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2456–2466.
- [39] Hong Wen, Jing Zhang, Yuan Wang, Fuyi Lv, Wentian Bao, Quan Lin, and Keping Yang. 2020. Entire space multi-task modeling via post-click behavior decomposition for conversion rate prediction. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*. 2377–2386.
- [40] 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.
- [41] Weinan Zhang, Tianqi Chen, Jun Wang, and Yong Yu. 2013. Optimizing top-n collaborative filtering via dynamic negative item sampling. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*. 785–788.
- [42] 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.
- [43] Feng Zhu, Mingjie Zhong, Xinxing Yang, Longfei Li, Lu Yu, Tiejia Zhang, Jun Zhou, Chaochao Chen, Fei Wu, Guanpeng Liu, et al. 2023. DCMT: A Direct Entire-Space Causal Multi-Task Framework for Post-Click Conversion Estimation. *arXiv preprint arXiv:2302.06141* (2023).