CTnoCVR: A Novelty Auxiliary Task Making the Lower-CTR-Higher-CVR Upper

Dandan Zhang*
Zhejiang Lab
Hangzhou, China
danae.zdd@zhejianglab.com

Haotian Wu*
Beijing Jiaotong University
Beijing, China
19711032@bjtu.edu.cn

Guanqi Zeng†
Cainiao Network
Hangzhou, China
eduboost.ai@gmail.com

Yao Yang†
Zhejiang Lab
Hangzhou, China
yangyao@zhejianglab.com

Weijiang Qiu
China Electric Power Research Institute
Beijing, China
qiuwj@epri.sgcc.com.cn

Yujie Chen
Cainiao Network
Hangzhou, China
aisling.cyj@alibaba-inc.com

Haoyuan Hu
Cainiao Network
Hangzhou, China
haoyuan.huhy@cainiao.com

ABSTRACT

In recent years, multi-task learning models based on deep learning in recommender systems have attracted increasing attention from researchers in industry and academia. Accurately estimating post-click conversion rate (CVR) is often considered as the primary task of multi-task learning in recommender systems. However, some advertisers may try to get higher click-through rates (CTR) by over-decorating their ads, which may result in excessive exposure to samples with lower CVR. For example, some only eye-catching clickbait have higher CTR, but actually, CVR is very low. As a result, the overall performance of the recommender system will be hurt. In this paper, we introduce a novelty auxiliary task called CTnoCVR, which aims to predict the probability of events with click but no-conversion, in various state-of-the-art multi-task models of recommender systems to promote samples with high CVR but low CTR. Plentiful Experiments on a large-scale dataset gathered from traffic logs of Taobao's recommender system demonstrate that the introduction of CTnoCVR task significantly improves the prediction effect of CVR under various multi-task frameworks. In addition, we conduct the online test and evaluate the effectiveness of our proposed method to make those samples with high CVR and low CTR rank higher.

*Both authors contributed equally to this research.
†Corresponding authors.

SIGIR '22, July 11–15, 2022, Madrid, Spain.

ACM ISBN 978-1-4503-8732-3/22/07…$15.00
https://doi.org/10.1145/3477495.3531843

CCS CONCEPTS

• **Computing methodologies** → **Neural networks**; • **Applied computing** → *Electronic commerce*.

KEYWORDS

Recommender Systems; Conversion Rate; CTnoCVR Auxiliary Task

ACM Reference Format:
Dandan Zhang, Haotian Wu, Guanqi Zeng, Yao Yang, Weijiang Qiu, Yujie Chen, and Haoyuan Hu. 2022. CTnoCVR: A Novelty Auxiliary Task Making the Lower-CTR-Higher-CVR Upper. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22), July 11–15, 2022, Madrid, Spain.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3477495.3531843

1 INTRODUCTION

As a result of the proliferation of the Internet, information overload is becoming increasingly ubiquitous, which leads to obstacles for customers to access desired information[10]. Recommender systems (RS) were proposed to filter and prioritize the information tailored to users' expectations, which effectively alleviates the information overload issue. Additionally, recommender systems can help platforms and marketers increase the effectiveness of information distribution. Estimating the post-click conversion rate (CVR) accurately is one of the crucial issues in recommender systems[7].

Taking a recommender for e-commerce as an explanatory example of CVR estimation, the platform first recommends several items to the users, who then click on some of the items and further purchase their preferred items. This inherent and sequential user action pattern of impression->click->conversion can be employed to model the CVR. In this paper, we focus on modeling post-click conversion rate, following the previous definition in[4], namely $pCVR = p(conversion \mid click, impression)$.

In recent years, deep learning-based approaches[3–5, 9] have been serving as a promising research direction in such recommender systems. To eliminate sample selection bias and data sparsity problems, Ma et al.[4] proposed an entire space deep learning model, which treats $pCVR$ as an intermediate variable derived by estimating $pCTCVR$ and $pCTR$, where $pCTCVR = p(conversion, click \mid impression)$. In addition, multi-task learning with deep learning can satisfy different business requirements simultaneously by jointly optimizing multiple objectives, so multi-task learning models in recommender systems are attracting more and more attention[6, 11]. Despite the notable success of multi-task learning in recommender systems, multi-task learning models do not always outperform the corresponding single-task models on all tasks, because of the data distribution differences and inherent conflicts among tasks. Ma et al.[3] proposed a novel multi-task learning approach, Multi-gate Mixture-of-Experts (MMoE), which can explicitly learn to model task relationships from data. MMOE demonstrates excellent performance in multiple scenarios across different datasets. Tang et al.[8] proposed Customized Gated Control (CGC) model to eliminate the challenge of the seesaw phenomenon. In contrast to MMOE, CGC separates special task-specific experts and shared experts to capture common and task-specific knowledge respectively, which mitigates harmful parameter interference between them. The experimental results show that CGC can handle inter-task rivalrousness better than MMOE.

However, in real industrial scenarios, advertisers often attempt to gain higher click-through rates (CTR) by over-decorating ads, resulting in overexposure of low CVR samples. For example, some clickbait or brand exposure samples that are only eye-catching have a high CTR but actually have a very low CVR, which hurts the overall performance of the recommender system. This problem has been largely neglected in previous studies[1, 2, 12, 13]. The Existing multi-task learning frameworks in recommender systems default to high CVR will be caused by high CTR. However, when there is no CVR interaction signal in the sample space, CTR will play a dominant role in the ranking. In this paper, we propose a novelty auxiliary task to penalize the high CTR but low CVR samples under various multi-task learning frameworks. This task aims to predict the probability of action with click but no-conversion, which is noted as $pCTnoCVR = p(noconversion, click|x)$, given a sample x. The advantages of this task are mainly in two aspects. The first is that it is not linearly related to the estimation of CTR, and is therefore less susceptible to high CTR, which potentially facilitates the training of the CVR task. Secondly, the label of the CTnoCVR task can be inferred by the label of CTR and CVR along with the entire space, which to some extent complements the information of the CVR task.

We deploy the proposed CTnoCVR task in three state-of-the-art models, ESMM, MMOE, and CGC, and evaluate these models on a large-scale publicly available dataset gathered from traffic logs of Taobao[1] recommender system. The offline experimental results demonstrate that the introduction of CTnoCVR significantly improves the performance of these three multi-task learning models. In addition, CTnoCVR effectively promotes the samples with high CVR but low CTR. We will open source our code on Github[2].

[1]https://en.wikipedia.org/wiki/Taobao, Taobao largest Chinese online shopping website.
[2]https://github.com/scalaboy/MutiTaskLearn.

The remainder of this paper is organized as follows: In Section 2, we present our proposed method. In Section 3, we introduce the settings of the experiments, and the experimental results are reported and discussed. Finally, we conclude our work briefly and provide future research directions in Section 4.

2 THE PROPOSED METHOD

2.1 Problem Definition

Assuming $S = \{x_i, y_i, z_i\}|_{i=1}^{N}$ is the observed dataset, where N is the total number of samples in the observed dataset. x_i represents the ith sample with multiple fields(e.g. user field, item field, combined field). y_i, z_i denote the labels of click and conversion to the ith sample respectively, which are binary.

Table 1: Valid labels in observations

Label Values	Is Validation	Comments
y = 0 & z = 0	Yes	no click and no conversion
y = 0 & z = 1	No	-
y = 1 & z = 0	Yes	click with no conversion
y = 1 & z = 0	Yes	click with conversion

Herein, two probability distributions can be defined:

- post-view click-through rate (CTR): $pCTR = p(y = 1|x)$
- post-click conversion rate (CVR): $pCVR = p(z = 1|y = 1, x)$

The above two distributions correspond to two main tasks of CTR and CVR prediction, which are common in recommender systems. The main focus of this paper is to estimate the probability of post-click conversion, namely $pCVR$.

2.2 CTnoCVR: Auxiliary Task Construction

To promote samples with high CVR but low CTR, we propose a punitive auxiliary task, namely post-view click-through & no-conversion (CTnoCVR), which aims to predict the probability of event with click but no-conversion. We denote c as the label of the CTnoCVR task, then we have $pCTnoCVR = p(c = 1|x)$.The binary label c is inferred based on the labels of the CVR and CTR task, and it is defined as:

- $c = 1$ if $y = 1$ & $z = 0$
- $c = 0$ if $y = 0$ or $y = 1$ & $z = 1$

Mathematically, the conditional probability of $pCTnoCVR$ can be decomposed as follows:

$$
\begin{aligned}
pCTnoCVR &= p(y = 1, z = 0 \mid x) \\
&= p(y = 1 \mid x)p(z = 0 \mid y = 1, x) \\
&= p(y = 1 \mid x)[1 - p(z = 1 \mid y = 1, x)] \\
&= p(y = 1 \mid x) - p(y = 1 \mid x)p(z = 1 \mid y = 1, x)
\end{aligned} \tag{1}
$$

$$
pCTnoCVR = pCTR - pCTR * pCVR = pCTR(1 - pCVR) \tag{2}
$$

This task is not linearly related to the click-through rate (CTR) prediction task, which means this task less susceptible to excessively high CTR and is potentially beneficial to the CVR task training. Much like the CTCVR task[4], CTnoCVR is not merely trained with clicked-only samples. Instead, it can be trained over the entire sample space to avoid data sparsity.

2.3 Take CGC Model as an Example

Considering that we achieve the best AUC of CVR prediction task under the CGC framework, we take CGC model as an example to explain how CTnoCVR task works and benefits to CVR prediction.

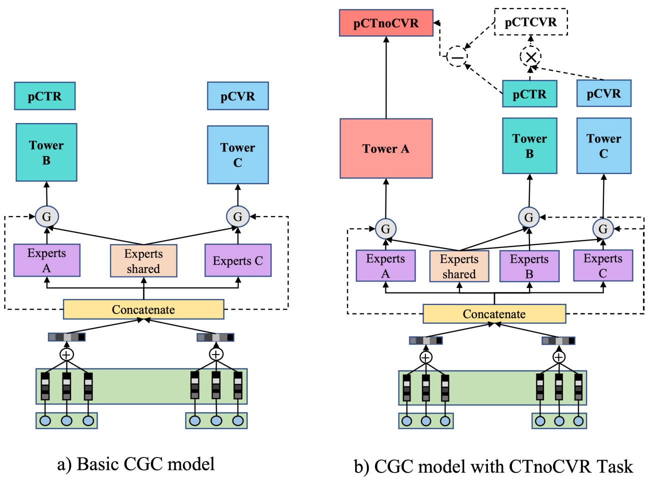

Figure 1: Comparison of Basic CGC Model and CGC Model with CTnoCVR Task.

To make good use of entire data and balance advertising intention, two auxiliary tasks, namely $pCTR$ and $pCTnoCVR$ are employed, to accurately predict $pCVR$. Our model generally contains three basic building blocks, which is depicted in Fig. 1. To facilitate the demonstration of the relationship between CTnoCVR on CTR and CVR, we have kept and dashed CTCVR task in Fig. 1, but the CTCVR task is not involved in the actual modeling in this paper.

The first one is the feature transformation block. The user field f_{user} and item field f_{item} are embedded separately, the representation of the field can be calculated by summing up the embedding vectors. Then, all the field vectors are concatenated as input of the next building block of CGC, denoted as x. Embedding parameters of CTR, CVR and CTnoCVR network are shared since this kind of parameter transfer learning can alleviate the data sparsity of CVR remarkably as shown in[4].

The second one is CGC block. CGC separates task-specific experts and shared experts[8], which is different from MMOE. The shared experts and gating mechanism make it possible for training of CVR task can employ the information in training the CTR and CTnoCVR tasks. G denotes a gate which is based on a single-layer forward feed network with Softmax as the activation function. The output of gating network for task k is $f^k(x)$, which can be calculated as follows:

$$f^k(x) = w^k(x)S^k(x) = \text{Softmax}\left(W_g^k x\right) S^k(x) \tag{3}$$

W_g^k is a trainable parameter matrix with dimension of $(m_k + m_s) \times d$, where m_k and m_s represent the number of specific experts and shared experts respectively and d denotes the dimension of input representation. S_k is the selected matrix for task k, which contains task k’s specific experts and shared experts. Specifically,

$$S_k = \left[E^T_{(k,1)}, E^T_{(k,1)}, \cdots, E^T_{(k,m_k)}, E^T_{(s,1)}, E^T_{(s,2)}, \cdots, E^T_{(s,m_s)}\right] \tag{4}$$

The last block is a stacked neural network for each task tower. In this paper, each tower, denoted as h, is a multiple-layer perception(MLP), which takes the output of gating network as input for one task. The output of task k tower is:

$$y^k = h^k\left(f^k(x)\right) \tag{5}$$

where y^k is the prediction (sigmoid activation) of task k.

Overall, all tasks trained together under the multi-task framework can leverage shared representations and learn task similarities and differences between tasks simultaneously. And all task-specific functionalities can be learned for further prediction.

In this paper, three tasks are co-train together. The CTR and CTnoCVR tasks are based on entire space to avoid data sparsity problem, while the CVR task is trained with only clicked sample data by giving zero weight to samples without click. CTnoCVR is treated as an intermediate task between CVR and CTR to ensure that CVR tasks can be trained on the entire space.

The loss function of the method proposed in this paper is defined as Eq (6). It consists of three loss terms from all the three tasks, which is a linear combination of different tasks.

$$\begin{aligned} l\left(\theta_{ctr}, \theta_{cvr}, \theta_{ctnocvr}\right) = &\sum_{i=1}^{N} l\left(y_i, f\left(x_i, \theta_{ctr}\right)\right) \\ &+ \lambda_1 * \sum_{i=1}^{N} l\left(y_i, f\left(x_i, \theta_{cvr}\right)\right) \\ &+ \lambda_2 * \sum_{i=1}^{N} l\left(c_i, f\left(x_i, \theta_{ctnocvr}\right)\right) \end{aligned} \tag{6}$$

where $\theta_{ctr}, \theta_{cvr}, \theta_{ctnocvr}$ are the parameters for sub-networks and $l(\cdot)$ is the cross-entropy loss function. λ_1 and λ_2 are the loss weights of the CVR and CTnoCVR tasks respectively.

3 EXPERIMENT

3.1 Dataset and Evaluation Metrics

3.1.1 The Ali-CCP. Alibaba Click and Conversion Prediction dataset[3] is adopted for evaluation. This is a dataset gathered from real-world traffic logs of the recommender system in Taobao. Users of Taobao.com might click interested items and make a further purchase among them. The statistic of the dataset is described in Table 2:

Table 2: Statistics of Experimental Dataset

Dataset	#user	#item	#impression	#click	#conversion
Ali-CCP	0.4M	4.3M	84M	3.4M	18k

[3]https://tianchi.aliyun.com/dataset/dataDetail?dataId=408

3.1.2 Metrics. Following previous study[4], we use Area under the ROC curve (AUC) scores as the evaluation metric to reflecting the ranking ability and report the averaged results with standard deviations over 10 runs.

3.2 Parameter Settings

To build strong baselines and make a fair comparison, the parameters for different models are optimized the same way. Specifically, to find the optimal parameter, we sweep over the following ranges respectively: the embedding size of all features in $\{12, 18, 24\}$, the dimension in the MLP layers in $\{300, 200, 80\}$, number of experts and units in $\{16, 8, 4\}$, the loss weights of the CVR task and CTnoCVR task in $\{0.2, 0.5, 0.8, 1, 2, 4, 5\}$. In addition, a parameter called score weight is introduced to enable the CVR task to better obtain information from the CTnoCVR task. We sweep the score weight over $\{0.0001, 0.0005, 0.001, 0.005, 0.01, 0.02, 0.03\}$.

3.3 The Necessity of CTnoCVR Task Introduction

In this section, we test the effectiveness of introducing the CTnoCVR task in two aspects. In the offline experiments, we introduced the CTnoCVR task in three state-of-the-art multitask learning models, ESMM, MMOE, and CGC, respectively, and compared the AUC of the CVR task before and after the introduction of this task, and the experimental results are shown in Table 3. In addition, we conduct online tests and analyze the data for the ranking of different types of samples before and after the introduction of the CTnoCVR task shown in Table 4, and the analysis results are shown in Fig. 2.

Table 3: Necessity of CTnoCVR Task Introduction in terms of CVR AUC.

Model		CVR AUC	Improvement	
			Absolute	Relative
ESMM	Base	67.11 ± 0.57	-	-
	+CTnoCVR	68.02 ± 0.49	+0.91	+1.36%
MMOE	Base	69.13 ± 0.21	-	-
	+CTnoCVR	69.61 ± 0.13	+0.48	+0.69%
CGC	Base	69.03 ± 0.04	-	-
	+CTnoCVR	69.80 ± 0.09	+0.77	+1.12%

The experimental results presented in Table 3 illustrate the effectiveness of the introduction of the CTnoCVR task. The introduction of the CTnoCVR task brings different degrees of improvement in different multi-expert frameworks. We observe that the most significant improvement is achieved by introducing CTnoCVR tasks in the ESMM framework. CVR AUC in the MMOE and CGC frameworks are generally higher than that in the ESMM, which is mainly due to the inter-task complexity and the seesaw phenomenon, and the MMOE and CGC frameworks reduce the impact of this phenomenon by controlling the shared information through the gating mechanism. In addition, the CGC framework generally outperforms MMOE due to the fact that CGC distinguishes task-specific experts from shared experts, which to a certain extent mitigates harmful parameter interference between them.

Table 4: CTR and CVR for Top 10 SKUs before and after the introduction of the CTnoCVR task. (The online tests are based on MMOE Framework.)

Before				After			
Sku id	CTR (%)	CVR (%)	Exposures	Sku id	CTR (%)	CVR (%)	Exposures
1	6.01	0.67	541.1k	1	7.45	0.86	1233k
2	1.06	0.13	524.8k	2	1.15	0.15	509.2k
3	1.70	0.13	312.1k	3	2.11	0.19	300.6k
4	1.99	0.72	220.7k	4	2.02	0.7	254.9k
5	**3.11**	**0.07**	**181.8k**	**7**	**2.32**	**0.3**	**200.7k**
6	**4.44**	**0.02**	**166.0k**	**5**	**4.22**	**0.12**	**103.5k**
7	**1.86**	**0.21**	**158.0k**	8	7.30	0.72	57.7k
8	5.86	0.36	146.5k	11	2.90	0.00	41.5k
9	11.26	0.03	130.6k	10	40.53	0.00	14.6k
10	17.05	0.00	86.8k	9	41.03	0.17	6.38k

As observed from Table 4, the introduction of the CTnoCVR task penalizes those samples (e.g. sku 5 and sku 6) with high CTR and low CVR, while increasing the exposure of those samples (e.g. sku 7) with low CTR and high CVR, which is consistent with our initial motivation for designing the CTnoCVR task. CTnoCVR is not linearly correlated with CTR and is less affected by an abnormally large CTR, so the models introducing CTnoCVR seldom tend to push up those samples with falsely high click-through rates. In addition to this, we observe that the model that introduces the CTnoCVR task tends to rank those samples with lower *pCTnoCVR* upper.

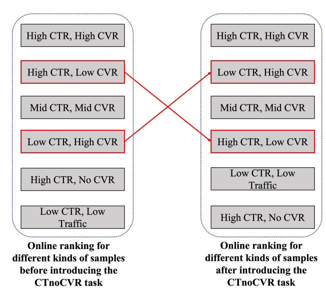

Figure 2: Necessity of CTnoCVR task introduction in terms of online ranking.

3.4 Ablation Study on Key Hyperparameters

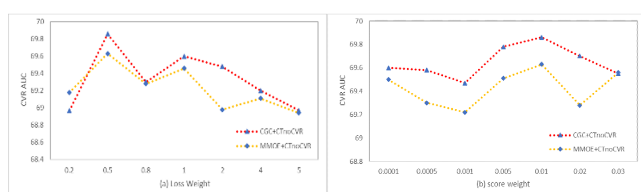

Figure 3: The results of different loss weight and score weight settings in MMOE and CGC model with CTnoCVR task.

In order to further evaluate the performance of models with CTnoCVR task, we take two critical parameters, namely loss weight for the CVR task and score weight for the CTnoCVR task. As shown in Figure 3(a), a loss weight for CVR of 0.5 leads to the best performance, which is probably due to data sparsity of the CVR task. The CTnoCVR task reserves more information but also contains noises and leads to higher model complexity. Increasing the score weight slightly for the CTnoCVR task may enhance the CVR task capacity to better obtain information, but also may fall into local optimum. We try different settings, as can be seen in Figure 3(b), it finally saturates at 0.01 while increasing it leads worse performance.

4 CONCLUSIONS AND FUTURE WORK

In this paper, a novel auxiliary task called CTnoCVR is proposed and introduced to various multi-task learning frameworks to promote samples with high CVR but low CTR. Experimental results on a large-scale dataset from Taobao show that the introduction of CTnoCVR tasks significantly and robustly improves the performance of different multi-expert frameworks. Such an improvement is reflected in two aspects. From the offline experiments, the CVR AUC of the multi-tasking model with the introduction of this task is generally higher than that of the model without introducing it. From the online test results, CTnoCVR makes the ranking of those samples with low CTR and high CVR upper. In future work, it would be valuable to explore the extension of the CTnoCVR task to more multi-task learning frameworks and the design of loss function.

ACKNOWLEDGMENTS

This work was supported by Key Research Project of ZheJiang Lab (No. 2022NF0AC01).

REFERENCES

[1] Xiaokai Chen, Xiaoguang Gu, and Libo Fu. 2021. Boosting share routing for multi-task learning. In *Companion Proceedings of the Web Conference 2021*. 372–379.

[2] Ke Ding, Xin Dong, Yong He, Lei Cheng, Chilin Fu, Zhaoxin Huan, Hai Li, Tan Yan, Liang Zhang, Xiaolu Zhang, et al. 2021. MSSM: a multiple-level sparse sharing model for efficient multi-task learning. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2237–2241.

[3] 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.

[4] 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.

[5] Zhen Qin, Yicheng Cheng, Zhe Zhao, Zhe Chen, Donald Metzler, and Jingzheng Qin. 2020. Multitask mixture of sequential experts for user activity streams. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 3083–3091.

[6] Sebastian Ruder. 2017. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098* (2017).

[7] Yumin Su, Liang Zhang, Quanyu Dai, Bo Zhang, Jinyao Yan, Dan Wang, Yongjun Bao, Sulong Xu, Yang He, and Weipeng Yan. 2021. An attention-based model for conversion rate prediction with delayed feedback via post-click calibration. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*. 3522–3528.

[8] 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 *Fourteenth ACM Conference on Recommender Systems*. 269–278.

[9] Hong Wen, Jing Zhang, Yuan Wang, Fuyu 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.

[10] Le Wu, Xiangnan He, Xiang Wang, Kun Zhang, and Meng Wang. 2022. A Survey on Accuracy-oriented Neural Recommendation: From Collaborative Filtering to Information-rich Recommendation. *IEEE Transactions on Knowledge and Data Engineering* (2022).

[11] 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.

[12] 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.

[13] Yongchun Zhu, Yudan Liu, Ruobing Xie, Fuzhen Zhuang, Xiaobo Hao, Kaikai Ge, Xu Zhang, Leyu Lin, and Juan Cao. 2021. Learning to Expand Audience via Meta Hybrid Experts and Critics for Recommendation and Advertising. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 4005–4013.