A Multi-Task Learning Approach for Delayed Feedback Modeling

Zhigang Huangfu
Ant Group
China
zhigang.hfzg@antfin.com

Gong-Duo Zhang
Ant Group
China
gongduo.zgd@alibaba-inc.com

Zhengwei Wu
Ant Group
China
zejun.wzw@antfin.com

Qintong Wu
Ant Group
China
qintong.wqt@antgroup.com

Zhiqiang Zhang
Ant Group
China
lingyao.zzq@antfin.com

Lihong Gu
Ant Group
China
lihong.glh@antgroup.com

Jun Zhou*
Ant Group
China
jun.zhoujun@antfin.com

Jinjie Gu
Ant Group
China
jinjie.gujj@antgroup.com

ABSTRACT

Conversion rate (CVR) prediction is one of the most essential tasks for digital display advertising. In industrial recommender systems, online learning is particularly favored for its capability to capture the dynamic change of data distribution, which often leads to significantly improvement of conversion rates. However, the gap between a click behavior and the corresponding conversion ranges from a few minutes to days; therefore, fresh data may not have accurate label information when they are ingested by the training algorithm, which is called the delayed feedback problem of CVR prediction. To solve this problem, previous works label the delayed positive samples as negative and correct them at their conversion time, then they optimize the expectation of actual conversion distribution via important sampling under the observed distribution. However, these methods approximate the actual feature distribution as the observed feature distribution, which may introduce additional bias to the delayed feedback modeling. In this paper, we prove the observed conversion rate is the product of the actual conversion rate and the observed non-delayed positive rate. Then we propose **M**ulti-**T**ask **D**elayed **F**eedback **M**odel (MTDFM), which consists of two sub-networks: actual CVR network and NDPR (non-delayed positive rate) network. We train the actual CVR network by simultaneously optimizing the observed conversion rate and non-delayed positive rate. The proposed method does not require the observed feature distribution to remain the same as the actual distribution. Finally, experimental results on both public and industrial datasets demonstrate that the proposed method outperforms the previous state-of-the-art methods consistently.

*Corresponding author.

WWW '22 Companion, April 25–29, 2022, Virtual Event, Lyon, France

ACM ISBN 978-1-4503-9130-6/22/04…$15.00
https://doi.org/10.1145/3487553.3524217

CCS CONCEPTS

• **Information systems → Collaborative filtering**.

KEYWORDS

delayed Feedback, recommender system, conversion rate prediction

ACM Reference Format:
Zhigang Huangfu, Gong-Duo Zhang, Zhengwei Wu, Qintong Wu, Zhiqiang Zhang, Lihong Gu, Jun Zhou, and Jinjie Gu. 2022. A Multi-Task Learning Approach for Delayed Feedback Modeling. In *Companion Proceedings of the Web Conference 2022 (WWW '22 Companion), April 25–29, 2022, Virtual Event, Lyon, France.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3487553.3524217

1 INTRODUCTION

Conversion rate (CVR) prediction is one of the most essential tasks for recommender system, which predicts the probability of whether a user will order an placement after clicking an item. In digital display advertising, a robust CVR prediction algorithm is important for a competitive advertising system. For example, to achieve a win-win situation for both the platform and advertisers, CVR prediction is used to adjust bid price per click. Recently, forecasting the CVR is widely investigated in both academia and industry.

In display advertising, the online learning is often used to capture the dynamic change of data distribution due to special events, new campaigns and other factors. However, conversions usually do not happen immediately after user clicks, it is observed ranging from a few minutes to days. The **delayed feedback** problem poses a challenge to the online learning model: we need new data to update our CVR model for model-freshness but these data have little user feedback.

Recently, several streaming training approaches are conducted to address delayed feedback problem by redesigning the data pipeline and loss function. Ktena et al. [4] treated each arriving instance as negative firstly and corrected upon its conversion at a later time, then they proposed the Fake Negative Weighted (FNW) loss function to optimize the expectation of true conversion distribution via importance sampling [6]. Yang et al. [7] studied the trade-off between waiting more accurate labels and exploiting fresher training data in the context of streaming CVR prediction. In their work, the training pipeline is designed by a short time window, when conversions occur in this window, these instances are labeled as positive and other instances are labeled as negative until they are converted. Then they proposed Elapsed-Time Sampling Delayed Feedback Model (ES-DFM) which is also a importance sampling based method to learn the actual CVR model under the observed distribution. Although these works achieve exciting performance, these methods have some issues. First, these important sampling based methods approximate the actual feature distribution as the observed feature distribution, we prove this assumption does not hold in our section 2.2. Second, these methods can only be used for the data pipeline they designed. In order to distinguish, we called the data pipeline used by FNW as real-time pipeline and data pipeline used by ES-DFM as elapsed-time pipeline.

In this work, we address these issues and propose a multi-task learning approach for delayed feedback modeling (MTDFM), which does not require the observed feature distribution to remain the same as the actual distribution and can be used in both real-time pipeline and elapsed-time pipeline. Instead of training the actual CVR model directly, the proposed method treats the actual CVR rate p_{cvr} as an intermediate variable which multiplied by the observed delayed positive rate p_{dp} equals to the observed CVR rate p_{ocvr}. Concretely, our model consists of two sub-networks: actual CVR network and NDPR (non-delayed positive rate) network. By making good use the relationship between the observed distribution and the actual distribution, We train the actual CVR model with the help of two auxiliary tasks of p_{dp} and p_{ocvr}. Our main contribution can be summarized as following:

- We propose a multi-task learning approach for delayed feedback modeling, which does not require the observed feature distribution to remain the same as the actual distribution. Moreover, we give the convergence of our method.
- We give the unified form for the relationship of actual and observed CVR distribution under the elapsed-time and the real-time pipeline. Therefore, our method can be used in both pipelines.
- We conduct experiments on public and industrial datasets. Our method outperforms the previous state-of-the-art results.

2 METHOD

2.1 Background

We focus on the CVR prediction task which can be formulated as probabilistic prediction of binary classification over a dataset $\mathcal{D} = \{x_i, y_i\}_{i=1}^N$, where x is the feature which consists of user field and item field and $y \in \{0, 1\}$ indicates the conversion label.

In practice, the ground truth label is unavailable since conversion action may delay a long period of time. In order to address the delayed feedback problem in the CVR prediction, the common approach is to wait for the real conversion in a certain time interval [7, 8]. Fake Negative Weighted (FNW) approach [4] can be viewed as a special case that waiting time window size is zero. Although the waiting time approach could partially correct the samples, the samples outside the waiting time window would be incorrectly labeled as negative. In conclusion, as the Figure 1 shows, there exists three types of samples:

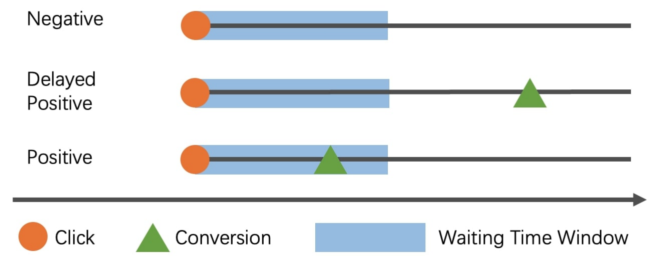

Figure 1: An illustration of different types of samples on training pipeline.

- **Negatives**. Negatives are samples that conversion does not occur inside the waiting time window.
- **Delayed Positives**. Delayed Positives have conversion action outside the waiting time window and are ingested into the training pipeline as soon as the engagement takes place.
- **Positives**. Positives are samples that conversion action takes place inside the waiting time window.

Then we denote p as the true data distribution and q as the biased observed data distribution which consists of the above three types of samples. Moreover, $p(y = 1|x)$ is the true conversion probability, $q(y = 1|x)$ is the probability of observing a conversion in the biased distribution and $q(dp = 1|x)$ denotes the probability of observing a delayed positive action in the biased distribution, where $dp \in \{0, 1\}$ is the label of delayed positive.

2.2 Relationship between True and Observed Conversion Distributions

With the ingestion of delayed positive samples, we know that $q(dp = 0) = \frac{1}{1+p(dp=1)}$ and $q(dp = 1) = \frac{p(dp=1)}{1+p(dp=1)}$. We can also get $q(x|dp = 0) = p(x)$ since the ingestion doesn't affect samples inside the waiting time window and $q(x|dp = 1) = p(x|dp = 1)$ and $q(x|y = 1) = p(x|y = 1)$ since the feature distribution of duplicated samples and conversion samples are the same in both true and observed data distribution. At last, we can get $q(y = 1) = \frac{p(y=1)}{1+p(dp=1)}$, since added delayed positive samples and true positive samples will be eventually labeled as positive in the observed data.

Based on probability equations described above and the law of total probability, the feature probability in the observed data can be computed as:

$$
\begin{aligned}
q(x) &= q(dp = 0)q(x|dp = 0) + q(dp = 1)q(x|dp = 1) \\
&= \frac{p(x)}{1 + p(dp = 1)} + \frac{p(x|dp = 1)p(dp = 1)}{1 + p(dp = 1)} \\
&= \frac{p(x) + p(x, dp = 1)}{1 + p(dp = 1)}.
\end{aligned} \tag{1}
$$

Based on probability equations described above and the formula for conditional probability, the joint observed distribution can be computed as:

$$q(x, y=1) = q(x|y=1)q(y=1) = \frac{p(x, y=1)}{1+p(dp=1)}. \tag{2}$$

By replacing Eq. (1) and Eq.(2) in the conditional probability formula that $q(y=1|x) = \frac{q(x,y=1)}{q(x)}$, we can obtain:

$$\begin{aligned} q(y=1|x) &= \frac{p(x,y=1)}{1+p(dp=1)} \cdot \frac{1+p(dp=1)}{p(x)+p(x,dp=1)} \\ &= \frac{p(x,y=1)}{p(x)+p(x,y=1,dp=1)} \\ &= \frac{p(y=1|x)}{1+p(y=1|x)p(dp=1|y=1,x)} \\ &= \frac{p(y=1|x)}{1+p(y=1|x)q(dp=1|y=1,x)}, \end{aligned} \tag{3}$$

where the last equation in Eq. (3) holds because the delayed positive distribution is unbiased in both true and observed data when conditioned on conversion.

Finally, after arranging Eq. (3), we can obtain the relationship between true and observed conversion distributions as:

$$p(y=1|x) = \frac{q(y=1|x)}{q(dp=0|x)}, \tag{4}$$

where $q(dp=0|x) = q(y=0|x)$ when the data pipeline is the real-time pipeline.

2.3 Multi-task Delayed Feedback Modeling

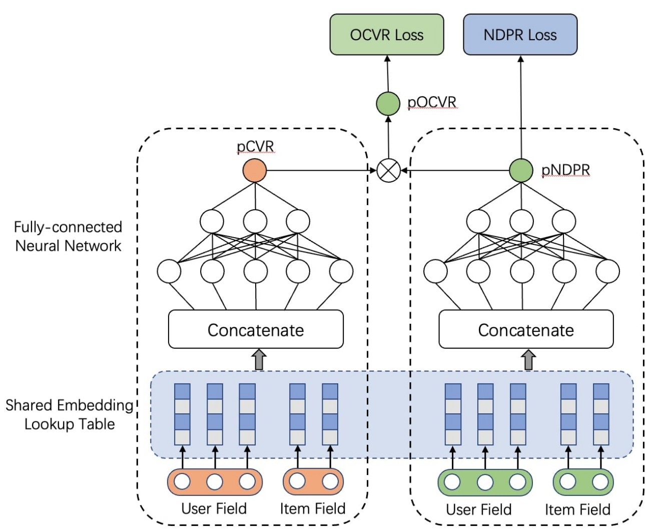

Figure 2: The Overall Architecture of MTDFM, which Consists of Two Tasks which Predict Observed Conversion and Non-conversion Behaviors.

In this section, we provide details of our proposed method, MTDFM (Multi-task Delayed Feedback Modeling). Figure 2 illustrates the overall architecture overview of MTDFM, which consists of two sub-networks: CVR (post-click conversion rate) network illustrated in the left part and NDPR (non-delayed positive rate) network in the right part. Then we conduct a multi-task learning approach to simultaneously predict the observed conversion probability $q(y=1|x)$ and the non-delayed positive probability $q(dp=0|x)$. Besides, we model $pNDPR$ instead of $pOCVR$ (observed post-click conversion rate) since divided by $pCVR$, which is usually a small number, would rise numerical instability. Inspired by ESMM [5] which consists of same structured CVR and CTR networks, we adopt fully-connected neural networks in CVR and NDPR modules and share embedding lookup table between them.

The loss function of MTDFM is defined as:

$$\begin{aligned} L(\theta_{cvr}, \theta_{ndpr}) &= \sum_{i=1}^{N} l(y_i, f_{\theta_{cvr}}(x_i)[f_{\theta_{ndpr}}(x_i)]) \\ &+ \sum_{i=1}^{N} l(1-dp_i, f_{\theta_{ndpr}}(x_i)), \end{aligned} \tag{5}$$

where θ_{cvr} and θ_{ndpr} are the parameters of CVR and NDPR networks, $l(\cdot)$ is cross-entropy loss function and the terms in $[\cdot]$ brackets are not taken into account in the gradient calculation of the loss with respect to the input. Note that the sub-networks in MTDFM apply fully-connected neural network, which can be substituted by other more sophisticated models [2, 3, 10, 11], which might obtain better performance. Due to limited space, we omit it and focus on tackling challenges of handling conversion delayed feedback in real practice.

2.4 Convergence Proof

In this section, we give a theoretical proof that $pCVR$ in MTDFM will converge to true conversion rate via online gradient descent. Note that the gradient of θ_{ndpr} is only contributed by the second part in loss function defined in Eq. (5) and the label $1-dp_i$ is unbiased, therefore $f_{\theta_{ndpr}}(x)$ will eventually converges to the true observed non-delayed positive probability. Finally, the gradient of $L(\theta_{cvr})$ with respect to $f_{\theta_{cvr}}$ can be written as:

$$\frac{\partial L}{f_{\theta_{cvr}}} = \frac{\partial \left\{ q(y=1|x) \log \left(f_{\theta_{cvr}}(x)[f_{\theta_{ndpr}}(x)] \right) \right\}}{\partial f_{\theta_{cvr}}} \tag{6}$$

$$+ \frac{\partial \left\{ (1-q(y=1|x)) \log(1-f_{\theta_{cvr}}(x)[f_{\theta_{ndpr}}(x)]) \right\}}{\partial f_{\theta_{cvr}}} \tag{7}$$

$$= \frac{p(y=1|x)q(dp=0|x)}{f_{\theta_{cvr}}(x)} \tag{8}$$

$$- \frac{(1-p(y=1|x)q(dp=0|x))[f_{\theta_{ndpr}}(x)]}{1-f_{\theta_{cvr}}(x)[f_{\theta_{ndpr}}(x)]} \tag{9}$$

$$\approx \frac{q(dp=0|x)\left(p(y=1|x)-f_{\theta_{cvr}}(x)\right)}{f_{\theta_{cvr}}(x)\left(1-f_{\theta_{cvr}}(x)q(dp=0|x)\right)}. \tag{10}$$

Eq. (10) holds when $f_{\theta_{ndpr}}(x)$ converges to $q(dp=0|x)$ after training sufficient steps. When $f_{\theta_{cvr}}(x) > p(y=1|x)$, $\partial L/\partial f_{\theta_{cvr}} < 0$. When $f_{\theta_{cvr}}(x) < p(y=1|x)$, $\partial L/\partial f_{\theta_{cvr}} > 0$. This indicates that the output of sub-network CVR $f_{\theta_{cvr}}(x)$ converges to true conversion distribution $p(y=1|x)$ and the gradients always point towards the right direction.

Table 1: Offline Results between MTDFM and Baselines of Criteo Dataset.

Methods	AUC	PR-AUC
FNC	0.8065	0.5817
FNW	0.8079	0.5935
ES-DFM	0.8082	0.5963
MTDFM	0.8103	0.5955
MTDFM-win	**0.8107**	**0.6000**
ORACLE*	0.8181	0.6126

3 EXPERIMENTS

3.1 Data Sets

To evaluate the effectiveness of different methods, we conducted experiments on a public dataset Criteo [1] and an industrial dataset from the online environment of Alipay application. The Criteo dataset is widely used for the delayed feedback problem. It cantains more than 15 million samples for 60 days and we use 7 days for our experiments. The Alipay dataset is collected from the marketing activities. We sub-sample 2% of the users and the sampled dataset contains about 2 million samples.

3.2 Experimental Setting

We chose the state-of-the-art delayed feedback models as baselines for efficiency comparison. The baselines include Fake Negative Weighted (FNW) [4], Fake Negative calibration (FNC) [4] and Elapsed-Time Sampling Delayed Feedback Model (ES-DFM) [7]. All methods include baselines and MTDFM use the same model architecture. The learning rate is set 0.01 and L2 regularization strength is set 10^{-6}. We use Area Under Curve (AUC) and Area Under the Precision-Recall Curve (PR-AUC) as metrics.

3.3 Streaming Experimental Protocol

We follow the streaming experimental setting in [7] and we construct the streaming data with the observed labels. Then the false negative data samples are added at their conversion time. The streaming data is divided into multiple sequence subsets according to their training hours. One subset only contains one hour data. These subsets will be fed into models sequentially. When the training process is completed with the data of t hours, the data of $t + 1$ hours will be used as evaluation. The AUC metrics is calculated with the means of all subsets. Due to a marketing activities last for a short time and usually do not exceed one month, in order to better evaluate different models for real marketing scenarios, we omit the pre-training phase and train all models only using streaming training.

To verify the performance of elapsed time, we train MTDFM with different settings. MTDFM adopts the same real time streaming training as FNW and FNC. MTDFM-win adopts 15 minutes elapsed time window with the same settings as ES-DFM in the paper[7].

3.4 Experimental Comparison

The results are shown in Table 1 and Table 2. The best results are marked in bold. Except baselines, we also show the performance of an ORACLE* model. The ORACLE* model uses training dataset with the ground truth labels instead of observed labels. There is no delayed feedback problem for ORACLE* model's CVR prediction. The performance of ORACLE* model is the upper bound of other methods. It can be found that MTDFM-win achieves the best performance in terms of AUC and PR-AUC of Criteo dataset. On Alipay dataset, MTDFM-win achieves the best performance in terms of AUC. The PR-AUC of MTDFM is better than MTDFM-win on Alipay dataset. The results show that using elapsed time window does not necessarily get the best result. It's a trade-off on the size of elapsed time window.

Table 2: Offline Results between MTDFM and Baselines of Alipay Dataset.

Methods	AUC	PR-AUC
FNC	0.6694	0.1203
FNW	0.7069	0.1115
ES-DFM	0.6700	0.1064
MTDFM	0.7022	**0.1245**
MTDFM-win	**0.7139**	0.1153
ORACLE*	0.8438	0.2047

4 RELATED WORK

Chapelle [1] first proposed a delayed feedback model (DFM) by applying survival time analysis under the assumption that the delay distribution is exponential. Yoshikawa and Imai [9] improved the model presented in Chapelle without assuming any parametric distributions, in which the authors suggested a non-parametric delayed feedback model (NoDeF) to capture the time delay. Yasui et al. [8] formulated the delayed feedback as a data shift where the training and test conditional label distributions are different and proposed an importance weighting approach (FSIW) to handle. One significant drawback of these methods was that they are hard to apply in the continuous training setting.

Ktena et al. [4] proposed to use loss functions of the Fake Negative Weighted (FNW) and Fake Negative calibration (FNC), which are applied for the first time on the delayed feedback problem, in online training via online gradient descent. Yang et al. [7] proposed Elapsed-Time Sampling Delayed Feedback Model (ES-DFM), which models the relationship between the observed conversion distribution and the true conversion distribution.

5 CONCLUSION

In this paper, we propose a multi-task learning approach MTDFM to address the delayed feedback problem in the streaming CVR prediction. Our approach learns the actual CVR model with the help of two auxiliary tasks of observed CVR rate and observed non-delayed positive rate. Moreover, MTDFM is a more general method comparing with previous approaches designed for a specific stream. Finally, experiments show MTDFM outperforms the previous state-of-the-art on both public and industrial data.

REFERENCES

[1] Olivier Chapelle. 2014. Modeling delayed feedback in display advertising. In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, Sofus A. Macskassy, Claudia Perlich, Jure Leskovec, Wei Wang, and Rayid Ghani (Eds.). ACM, 1097–1105. https://doi.org/10.1145/2623330.2623634

[2] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. 2016. Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems*. 7–10.

[3] 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).

[4] Sofia Ira Ktena, Alykhan Tejani, Lucas Theis, Pranay Kumar Myana, Deepak Dilipkumar, Ferenc Huszár, Steven Yoo, and Wenzhe Shi. 2019. Addressing delayed feedback for continuous training with neural networks in CTR prediction. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 187–195.

[5] 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.

[6] Markus Svensén and Christopher M Bishop. 2007. Pattern recognition and machine learning.

[7] Jia-Qi Yang, Xiang Li, Shuguang Han, Tao Zhuang, De-Chuan Zhan, Xiaoyi Zeng, and Bin Tong. 2021. Capturing Delayed Feedback in Conversion Rate Prediction via Elapsed-Time Sampling. *Intelligence (www. aaai. org)* 61921006 (2021), 61751306.

[8] Shota Yasui, Gota Morishita, Fujita Komei, and Masashi Shibata. 2020. A Feedback Shift Correction in Predicting Conversion Rates under Delayed Feedback. In *Proceedings of The Web Conference 2020*. 2740–2746.

[9] Yuya Yoshikawa and Yusaku Imai. 2018. A nonparametric delayed feedback model for conversion rate prediction. *arXiv preprint arXiv:1802.00255* (2018).

[10] Guorui Zhou, Na Mou, Ying Fan, Qi Pi, Weijie Bian, Chang Zhou, Xiaoqiang Zhu, and Kun Gai. 2019. Deep interest evolution network for click-through rate prediction. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 33. 5941–5948.

[11] 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.