User Activity Modeling under Inflated Distribution

Xiang Li*
ByteDance, Beijing, China
Peking University, Beijing, China
lixiang.2222@bytedance.com

Zhao-Yu Zhang*
ByteDance, Beijing, China
zhaoyu.1@bytedance.com

Chunyuan Zheng
ByteDance, Beijing, China
Peking University, Beijing, China
zhengchunyuan@bytedance.com

Qingying Chen
chenqingying@bytedance.com
ByteDance, Beijing, China

Huiyou Jiang†
ByteDance, Beijing, China
jianghuiyou.jhy@bytedance.com

Haoxuan Li
Peking University, Beijing, China
hxli@stu.pku.edu.cn

Zhouchen Lin
State Key Lab of General AI, School of Intelligence Science and Technology, Peking University, Beijing, China
zlin@pku.edu.cn

Abstract

User activity modeling serves as an effective approach to reflect Daily Active Users (DAU) levels by forecasting users' prospective behavioral engagement. In industrial practice, particularly in user active days prediction tasks, the data demonstrates extreme distributional imbalance. Specifically, completely inactive and fully active users constitute the vast majority, resulting in a highly skewed phenomenon known as a two-sided inflated distribution. Existing methods either rely on the assumption of a normal distribution, neglecting the optimization objectives to focus purely on designing complex model architectures for user feature extraction, which renders them ineffective in non-normally distributed real-world industrial scenarios, or they have explored modeling non-normal distributions, but still struggle to directly operate on the discrete data space in user activity modeling and mitigate the two-sided inflation prevalent in industrial data. Based on this finding, we propose a simple yet effective user activity modeling method under two-sided inflated distribution. Specifically, we decompose the problem into two objectives: first, modeling whether the target is at an inflated point; and second, modeling the distribution when the target is not at an endpoint. Finally, we apply this modeling approach to uplift modeling. Extensive offline evaluations and online A/B tests on Douyin platforms, involving over 1 billion users, demonstrate the effectiveness of the proposed method.

CCS Concepts

• **Information systems → Information retrieval**; • **Computing methodologies → Machine learning**; • **Applied computing → Electronic commerce**.

*Xiang Li and Zhao-Yu Zhang contributed equally to this submission.
†Huiyou Jiang is the corresponding author.

ACM ISBN 979-8-4007-2599-9/2026/07
https://doi.org/10.1145/3805712.3808476

Keywords

User Activity Modeling, Uplift Modeling

ACM Reference Format:
Xiang Li, Zhao-Yu Zhang, Chunyuan Zheng, Qingying Chen, Huiyou Jiang, Haoxuan Li, and Zhouchen Lin. 2026. User Activity Modeling under Inflated Distribution. In *Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '26), July 20–24, 2026, Melbourne, VIC, Australia.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3805712.3808476

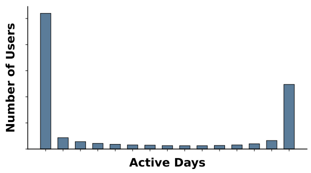
1 Introduction

In industrial recommendation and e-commerce platforms, maximizing Daily Active Users (DAU) is essential for maintaining platform health [4, 38, 39]. Achieving this relies on precise user activity modeling to quantitatively forecast a user's future behavioral feedback. Accurately forecasting user activity enables platforms to design appropriate recommendation strategies, such as targeted push notifications, personalized content delivery, and promotional campaigns, thereby maximizing overall gross revenue and significantly enhancing user retention and experience, and ultimately driving DAU growth.

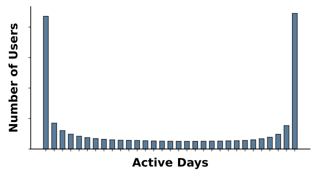
To enhance the capability of user activity modeling, existing works have primarily focused on designing sophisticated model architectures, including decoupled attention, asynchronous caching, and stacking fine-tuning [2, 14, 18, 26]. With the impressive advances in deep learning, these complex architectural designs have gathered increasing attention and achieved significant improvements in various industrial downstream tasks [28, 36, 40]. Despite these advances in model structure, loss function design has been overlooked in existing methods.

In industrial user active days prediction, a ubiquitous phenomenon is that the target data typically exhibits a two-sided inflated distribution, which specifically manifests here as a heavily two-sided inflated distribution. For example, as shown in Figure 1, users' active days for both 14 days (Lifetime 14, LT14) and 30 days (Lifetime 30, LT30) on the Douyin[1] platform show that most users are

[1]https://www.douyin.com/



(a) Douyin active days in 14-day window.



(b) Douyin active days in 30-day window.

Figure 1: Active Days (LT14/30) of the Douyin platforms involving over 1 billion users: The number of active days per user over a 14/30-day window, capturing user stickiness and retention.

either completely inactive or fully active at both extremes. Meanwhile, on other platforms like Toutiao[2] and Doubao[3], we observe a similar two-sided inflated distribution, further confirming that this extreme data inflation is a widespread characteristic across various industrial applications.

Based on this observation, directly adopting the Mean Square Error-based or the Gaussian Mixture Model-based loss function to train the uplift prediction model will lead to sub-optimal performance [41]. While existing works have proposed generalized zero-inflation strategies for modeling non-normal distributions [3, 6, 19, 33], they still struggle to directly operate on the discrete data space inherent in user activity modeling and mitigate the two-sided inflation prevalent in industrial data.

To bridge this gap, we propose a simple yet effective method for user activity modeling. Specifically, we decompose the problem into the following two objectives: first, we aim to train a multi-class classification model to predict if the user will fall into the inflated point; secondly, we model the distribution if the user is not at endpoints. Note that this method can not only address the distribution inflation problem but can also overcome the data imbalance. Finally, we apply this modeling approach to uplift modeling. The main contributions are as follows:

- We identify the prevalence of multiple point-inflation phenomena in industrial data distributions, which has been overlooked in existing user activity modeling research.
- We propose a generalized user activity modeling framework tailored to address the data imbalance inherent in two-sided inflated distributions.
- We evaluated the proposed method through large-scale online A/B testing on two platforms with a user base exceeding 1 billion. Currently, the method is fully deployed and serves the complete main traffic on these platforms.

2 Related Work

2.1 User Activity Modeling

User active days refer to the dates on which a user has engaged in valid sessions, such as logging in or browsing on the platform [15]. Early research on modeling user active days within specific time windows relied on statistical analysis to capture temporal dynamics and periodic patterns [22, 25]. With the advancement of deep learning techniques, more modeling approaches based on recurrent neural networks [24], graph neural networks [29], or mixture of experts [21] have been proposed. Although these deep models possess strong representational capabilities for complex temporal patterns in user activity behavior, they pay insufficient attention to the specific distribution characteristics of point-inflation in user active days. Consequently, their prediction accuracy for user active days at inflated points is relatively poor.

2.2 Distribution Modeling

Generalized distribution modeling serves as a crucial theoretical foundation for user activity modeling. Beyond applying common probabilistic distribution models, researchers have proposed complex parametric models such as Gaussian mixture models [41], as well as nonparametric methods like Bayesian approaches [1, 20] to capture the complexity of user behavior. Although these methods may possess desirable theoretical properties, when applied to user activity modeling, they often fail to capture the complex distribution of two-sided inflation.

2.3 Uplift Modeling

Uplift modeling estimates heterogeneous treatment effects to distinguish treatment-driven conversions from natural ones [23, 27, 34, 42], and is tightly connected to causal debiasing in recommender systems [9, 11, 31, 35]. Core approaches include meta-learners and their deep extensions [5, 44], tree-based uplift methods [23, 27], doubly robust and causal balancing techniques [7, 8, 12, 45], and counterfactual learning [13, 16, 30, 32]. Recent work further tackles practical issues such as propensity bias [37, 43], noisy labels [17], and missing-not-at-random data [10, 46]. Our work addresses the overlooked distributional challenge of two-sided inflation in user activity targets within the uplift framework.

3 Methods

3.1 Notation and setup

Following the notations in fundamental uplift modeling, we use X_i ($i \in \{1, \dots, N\}$) to represent the user features, Y_i ($i \in \{1, \dots, N\}$) to represent the objective, such as user click days, user click frequency, or life-time. In addition, we use T_i to represent the treatment for user i. The treatment T_i takes values among a pre-defined set $\{0, 1, \dots, T_0\}$ while Y_i is from 0 to the right point k. For illustrative purposes, we assume that there is two-sided inflation, i.e., the distribution is inflated at $Y = 0$ and $Y = k$.

3.2 The Proposed Two-Sided Inflated Loss (TSIL)

The overall framework of the proposed TSIL model is based on the widely used X-Net structure [44]. The core of TSIL is two hierarchical prediction subtasks. As shown in Figure 2, the overall structure of TSIL consists of a shared bottom representation layer, two task-specific transformer encoders, and a loss fusion layer.

3.2.1 *Task 1: Three-Class Classification.*

To capture the two-sided inflated characteristic, we first design a three-class classification subtask to model the probability of Y falling into three intervals

[2] https://www.toutiao.com/
[3] https://www.doubao.com/

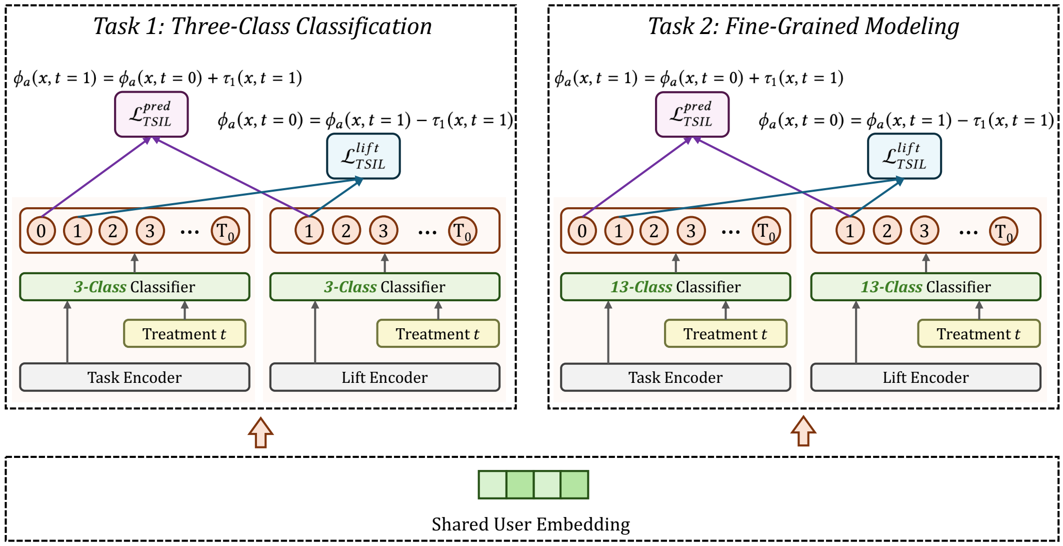


Figure 2: Overview of the model structure. Task 1 classifies users into three activity categories ($A \in \{0, 1, 2\}$); Task 2 models the discrete active-day distribution for middle-active users ($A = 1$). Both tasks share a bottom embedding layer and each contains a Task Encoder and a Lift Encoder.

based on an indicator $A \in \{0, 1, 2\}$, where:

$$\begin{cases} A = 0 \Leftrightarrow Y = 0 \quad \text{(inactive users)} \\ A = 1 \Leftrightarrow 0 < Y < 14 \quad \text{(middle active users)} \\ A = 2 \Leftrightarrow Y = 14 \quad \text{(fully active users)} \end{cases} .$$

We denote $\phi_a(x, t) = P(A = a|x, t)$ as the probability of user x falling into interval a under treatment t, with the constraint $\sum_{a=0}^{2} \phi_a(x, t) = 1$. In our model, $\phi_a(x, t) = P(A = a|x, t)$ is predicted by a task-specific transformer encoder.

3.2.2 Task 2: Fine-Grained Modeling. For the middle active user group ($A = 1$), we design a fine-grained probability modeling subtask to capture the discrete distribution of active days. We denote $p_{k,t} = P(Y = k|x, t, A = 1)$ as the probability that the user's active days is k ($k \in \{1, \ldots, 13\}$) under treatment t, with the constraint $\sum_{k=1}^{13} p_{k,t} = 1$. The probability $p_{k,t}$ is predicted by another task-specific transformer encoder, which shares the bottom layer with Task 1 to reduce parameters and capture feature commonality.

3.2.3 Loss Function Design. We design a **loss function** combining a standard cross entropy loss to train the task 1 classifier, the negative log-likelihood loss for probability modeling of the two subtasks, and regression loss for numerical active days value prediction. The total loss function is:

$$\mathcal{L}_{TSIL} = \mathcal{L}_{CE} + \mathcal{L}_{NLL} + \mathcal{L}_{Reg}.$$

Cross-Entropy Loss ($\mathcal{L}_{CE}$). The three-class probability classifier is trained by the following standardized cross-entropy loss:
$\mathcal{L}_{CE} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{a=0}^{2} \mathbb{I}(A_i = a) \cdot \log(\phi_a(x_i, t_i))$,where $\mathbb{I}(\cdot)$ is the indicator function.

Negative Log-Likelihood Loss ($\mathcal{L}_{NLL}$). The NLL loss models the joint probability of the observed active days values (A, Y) (where $Y = 0$ for $A = 0$, $Y = k$ for $A = 1$, $Y = 14$ for $A = 2$). The joint probability is:

$$\begin{cases} P(A = 0, Y = 0) = \phi_0(x, t) \\ P(A = 1, Y = k) = \phi_1(x, t) \cdot p_{k,t} \\ P(A = 2, Y = 14) = \phi_2(x, t) \end{cases} .$$

The NLL loss is then defined as:

$$\mathcal{L}_{NLL} = -\mathbb{I}(Y = 0, A = 0) \log \phi_0(x, t) - \mathbb{I}(Y = 14, A = 2) \log \phi_2(x, t) - \mathbb{I}(Y = k, A = 1) \log \left(\phi_1(x, t) \cdot p_{k,t}\right).$$

Regression Loss ($\mathcal{L}_{Reg}$). The regression loss is designed to improve the prediction accuracy of the active days value, which is:

$$\mathcal{L}_{Reg} = \left(\phi_1(x, t) \sum_{k=1}^{13} k \cdot p_{k,t} + 14 \cdot \phi_2(x, t) - Y\right)^2 ,$$

where the predicted value is the mathematical expectation of active days, which combines the results of Task 1 and Task 2.

3.2.4 Adaptation to X-Net Structure. In practice, the ultimate goal is to estimate the causal effect of a treatment on user activity, i.e., uplift modeling (see Subsection 2.3). We therefore adopt X-Net [44], a widely used uplift architecture, as the backbone to illustrate how TSIL integrates into this framework. Specifically, the TSIL model adopts a multi-tower structure with the prediction head and uplift head. Raw user features X and treatment features T are first encoded by the shared bottom transformer, generating unified embeddings. Then, prediction encoders directly output $\phi_a(x, t) = P(A = a|x, t)$ (Task 1) and $p_{k,t} = P(Y = k|x, t, A = 1)$ (Task 2), while lift encoders estimate treatment effects $\tau_1(x, t) = P(A = a|x, t) - P(A = a|x, t = 0)$ and $\tau_2(x, t) = P(Y = k|x, t, A = 1) - P(Y = k|x, t = 0, A = 1)$, quantifying difference between $T = t$ (treatment) and $T = 0$ (control).

Therefore, there are two possible ways to compute each prediction. For the treatment group with $T > 0$, taking $\phi_a(x, t = 1)$ as an

Table 1: Performance comparison on public datasets. Bold indicates the best result; underline indicates the second best. ↑ means higher is better; ↓ means lower is better.

Dataset	Method	AUC↑	Spearman↑	MAPE↓
Kaggle Shopper	MSE	0.6578	0.2919	0.0803
	ZILN	0.6793	<u>0.3150</u>	0.0732
	GMM	<u>0.6802</u>	0.3144	<u>0.0645</u>
	Ours	**0.7007**	**0.3174**	**0.0192**
KDD CUP 98	MSE	0.4774	0.0071	0.1045
	ZILN	0.5320	0.0084	0.0858
	GMM	<u>0.5420</u>	<u>0.0090</u>	<u>0.0731</u>
	Ours	**0.5547**	**0.0110**	**0.0522**

example, we can (1) directly use the prediction head to compute, or (2) compute as $\phi_a(x, t=1) = \phi_a(x, t=0) + \tau_1(x, t=1)$. Meanwhile, for the control group with $T = 0$, taking $\phi_a(x, t=0)$ as an example, besides directly using the prediction head, we can use $\phi_a(x, t=0) = \phi_a(x, t=1) - \tau_1(x, t=1)$ to compute.

Therefore, the final loss function includes two parts, $\mathcal{L}_{TSIL}^{pred}$, which directly uses the prediction head to compute, and $\mathcal{L}_{TSIL}^{lift}$, which includes the lift head to compute. In addition, we use multi-task learning to train models, and each output is activated via elu + 1 to ensure non-negativity.

4 Experiment

To demonstrate the effectiveness of the proposed method, we conduct offline experiments on both public and industrial datasets. In addition, we conduct online A/B tests involving over 1 billion users on Douyin.

4.1 Dataset and Task Description

To comprehensively evaluate the proposed TSIL method, we conduct experiments on both public datasets and an industrial dataset.

Public Datasets. Following ZILN [33], we use two public LTV prediction benchmarks: **Kaggle Shopper**[4] (311K customers, predicting 12-month purchase value) and **KDD CUP 98**[5] (200K donors, predicting donation amount).

Industrial Dataset. The **Douyin** dataset is collected via a push notification RCT with 20.59 million users. Control and treatment groups received zero and predefined-frequency notifications, respectively. We recorded 14-day active days (LT14).

For all datasets, we stratify each target into three categories ($A \in \{0, 1, 2\}$): $A = 0$ for the left inflated point ($Y = 0$), $A = 2$ for the right inflated point ($Y = k$), and $A = 1$ for the middle region ($0 < Y < k$).

4.2 Offline Experiments

4.2.1 Experiment Details. **Baseline.** We compare our method against the following baselines under the same model architecture: (1) **MSE**, a standard regression model trained with mean squared error loss; (2) **ZILN** [33], a zero-inflated lognormal model that handles zero inflation by combining a binary classifier with a lognormal distribution for positive values; and (3) **GMM** [41], a Gaussian mixture model-based loss that models the target distribution as a mixture of Gaussian components.

[4]https://www.kaggle.com/c/acquire-valued-shoppers-challenge
[5]https://kdd.ics.uci.edu/databases/kddcup98/kddcup98.html

Table 2: The AUUC and Calibration in Douyin.

Dataset	Metric	MSE	GMM	Ours
Douyin	AUUC	0.9552	0.9602	**0.9660**
	Calibration	**0.9827**	0.4557	1.0849

Table 3: Ablation study of loss components on public datasets.

Dataset	Variant	AUC↑	Spearman↑	MAPE↓
Kaggle Shopper	w/o NLL + Reg	0.6726	0.2922	0.0722
	w/o NLL	0.6877	0.3071	0.0552
	w/o Reg	0.6925	0.3157	0.0233
	All (Ours)	**0.7007**	**0.3174**	**0.0192**
KDD CUP 98	w/o NLL + Reg	0.5390	0.0074	0.0764
	w/o NLL	0.5400	0.0082	0.0779
	w/o Reg	0.5409	0.0090	0.0622
	All (Ours)	**0.5547**	**0.0110**	**0.0522**

Evaluation Metrics. For the public datasets, following prior work [33], we adopt three complementary metrics: (1) **Area Under the Curve (AUC)↑** (higher is better), measuring the model's discriminative ability in ranking users by predicted values; (2) **Spearman Correlation↑** (higher is better), capturing the monotonic rank association between predicted and actual values; and (3) **Mean Absolute Percentage Error (MAPE)↓** (lower is better), measuring the decile-level mean absolute percentage error for model calibration.

For the industrial dataset, we evaluate performance via two metrics: (1) **Area Under the Uplift Curve (AUUC)↑** (higher is better) computes the normalized area under the cumulative gain curve sorted by predicted uplift. (2) **Calibration Ratio** (closer to 1 is better) compares predicted to actual effects, and deviations indicate over- or under-estimation.

4.2.2 Performance Comparison. Table 1 and Table 2 present the offline prediction performance on public benchmarks and the Douyin industrial dataset, respectively. On the public datasets, the proposed TSIL method achieves the best performance across all metrics, consistently outperforming all baselines in both ranking quality and calibration accuracy. On the Douyin industrial dataset, TSIL similarly achieves the highest AUUC and the most proper calibration among all compared methods, confirming its effectiveness at billion-scale deployment. Among baselines, MSE performs worst due to its implicit normality assumption; ZILN [33] improves by modeling zero inflation but only addresses the left endpoint; GMM [41] further advances with mixture components yet still lacks explicit handling of two-sided inflation. TSIL's advantage stems from capturing probability mass at both inflated endpoints, while the fine-grained modeling accurately characterizes middle-region users.

4.2.3 Ablation Study. Table 3 investigates the roles of each loss component in our method: (1) **w/o NLL + Reg** retains only $\mathcal{L}_{\text{CE}}$,

Table 4: Performance of MAE across different subgroups in public datasets.

Dataset	Method	$A=0$	$A=1$	$A=2$
Kaggle Shopper	MSE	37.769	28.118	35.792
	ZILN	<u>29.979</u>	27.081	<u>31.229</u>
	GMM	34.769	<u>26.505</u>	31.371
	Ours	**28.990**	**25.710**	**28.712**
KDD CUP 98	MSE	0.758	0.625	1.280
	ZILN	<u>0.692</u>	0.619	1.144
	GMM	0.703	<u>0.608</u>	<u>0.953</u>
	Ours	**0.638**	**0.606**	**0.899**

Table 5: Performance of AUUC and Calibration metrics across different user activity subgroups in Douyin.

Activity	Metric	Baseline	Ours
Overall	AUUC	0.9552	**0.9660**
	Calibration	**0.9827**	1.0849
$A=0$	AUUC	0.8792	**0.9145**
	Calibration	0.3926	**0.9832**
$A=1$	AUUC	0.8699	**0.8714**
	Calibration	**0.9462**	1.0725
$A=2$	AUUC	0.9488	**0.9495**
	Calibration	1.1005	**1.0098**

which yields the weakest ranking quality across both datasets. (2) **w/o NLL** uses $\mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{Reg}}$, removing the fine-grained distribution modeling. While it improves over the CE-only variant, ranking quality remains limited. (3) **w/o Reg** uses $\mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{NLL}}$, removing regression regularization. This variant achieves competitive ranking but higher prediction error. (4) **All (Ours)** integrates all three components and achieves the best performance across all metrics. The improvement indicates that $\mathcal{L}_{\text{NLL}}$ and $\mathcal{L}_{\text{Reg}}$ work synergistically.

4.2.4 In-Depth Analysis on Inflated Point. Table 4 and Table 5 further decompose performance across user activity subgroups. Our method achieves the lowest MAE on all three subgroups of both public datasets, and yields steady AUUC improvements on Douyin with substantially better Calibration, especially for inactive users ($A=0$) where the baseline severely underestimates. These results provide empirical evidence supporting our claim that existing deep models have insufficient prediction accuracy at inflated points, and demonstrate the effectiveness of our method at inflated points.

4.3 Online A/B Test Experiments

4.3.1 Experiment Details. In the online A/B testing, we evaluate performance via three metrics: (1) **Active days (R_g)**: The relative change ($\times 10^{-4}$) in user active days versus the baseline. (2) **Cost (R_c)**: The relative change in sent notifications. (3) **Return on Investment (ROI)**: Evaluates resource efficiency defined as $ROI = \frac{1+R_g}{1+R_c}$. We record the cumulative relative changes for these metrics at Days 4, 7, and 10.

Table 6: Online A/B test results across user activity subgroups in Douyin.

Activity	Active days (1e-4)	Cost	ROI
Day4			
Overall	+0.410% ± 0.630%	-0.099% ± 0.026%	+0.509%
$A=0$	+29.550% ± 16.640%	+9.951% ± 0.140%	**+17.822%**
$A=1$	+0.570% ± 5.360%	+0.018% ± 0.070%	+0.553%
$A=2$	-0.030% ± 0.600%	-0.571% ± 0.040%	+0.549%
Day7			
Overall	+0.770% ± 0.570%	-0.076% ± 0.026%	+0.847%
$A=0$	+32.910% ± 14.430%	+10.381% ± 0.138%	**+20.409%**
$A=1$	+1.610% ± 5.360%	-0.002% ± 0.070%	+1.610%
$A=2$	+0.080% ± 0.560%	-0.567% ± 0.043%	+0.655%
Day10			
Overall	+0.760% ± 0.560%	+0.002% ± 0.025%	+0.758%
$A=0$	+40.390% ± 13.690%	+12.945% ± 0.130%	**+24.302%**
$A=1$	+1.210% ± 4.320%	-0.020% ± 0.071%	+1.231%
$A=2$	-0.020% ± 0.570%	-0.631% ± 0.033%	+0.613%

4.3.2 Results Analysis. Table 6 details the performance comparison in the online A/B testing, and we report the relative changes for each metric along with their corresponding confidence intervals. Specifically, our proposed method demonstrates superiority in three main aspects: (1) Regarding the improvement in user active days, a routine industrial iteration typically yields an average increase of $0.2\%\times 10^{-4}$, **our method significantly exceeds average increase**, achieving substantial increases in both active days and ROI; (2) Our method achieves precise offline calibration for inactive users ($A=0$) as indicated in Table 2, effectively correcting the severe underestimation of the MSE baseline, and guided our online strategy to increase notifications for $A=0$ users; although this required reallocating limited notification capacity from the $A=1$ and $A=2$ groups to the $A=0$ group, the ROI still realized positive growth across all scenarios, **proving the strong consistency between our offline modeling and online experimental results**. (3) Our method consistently outperforms the baseline across all evaluated time windows (Day 4, Day 7, and Day 10) and user groups, **showing its robust stability and sustained effectiveness over time**.

5 Conclusion

In this paper, we introduced a novel Two-Sided Inflated Loss (TSIL) framework for industrial user activity modeling, which effectively addresses the severe two-sided inflated distribution prevalent in real-world data. Our method decomposes prediction into an end-point classification subtask and a fine-grained distribution subtask for middle-active users. Extensive offline and online A/B experiments on billion-scale platforms validate TSIL's effectiveness, and the method is fully deployed on Douyin.

Acknowledgment

Z. Lin was supported by the NSF China (No. 62276004) and the State Key Laboratory of General Artificial Intelligence.

References

[1] Rahimpour Cami Bagher, Hamid Hassanpour, and Hoda Mashayekhi. 2017. User Trends Modeling for a Content-Based Recommender System. *Expert Systems with Applications* 87 (2017), 209–219.

[2] Xiangyi Chen, Kousik Rajesh, Matthew Lawhon, Zelun Wang, Hanyu Li, Haomiao Li, Saurabh Vishwas Joshi, Pong Eksombatchai, Jaewon Yang, Yi-Ping Hsu, et al. 2025. PinFM: Foundation Model for User Activity Sequences at a Billion-Scale Visual Discovery Platform. In *ACM Conference on Recommender Systems*.

[3] Dianliang Deng and Yu Zhang. 2015. Score Tests for Both Extra Zeros and Extra Ones in Binomial Mixed Regression Models. *Communications in Statistics-Theory and Methods* 44, 14 (2015), 2881–2897.

[4] Zhichao Feng, JunJie Xie, Kaiyuan Li, Yu Qin, Pengfei Wang, Qianzhong Li, Bin Yin, Xiang Li, Wei Lin, and Shangguang Wang. 2024. Context-Based Fast Recommendation Strategy for Long User Behavior Sequence in Meituan Waimai. In *Companion Proceedings of the ACM Web Conference*.

[5] Sören R Künzel, Jasjeet S Sekhon, Peter J Bickel, and Bin Yu. 2019. Metalearners for estimating heterogeneous treatment effects using machine learning. *Proceedings of the National Academy of Sciences* 116, 10 (2019), 4156–4165.

[6] Diane Lambert. 1992. Zero-Inflated Poisson Regression, With an Application to Defects in Manufacturing. *Technometrics* 34, 1 (1992), 1–14.

[7] Haoxuan Li, Quanyu Dai, Yuru Li, Yan Lyu, Zhenhua Dong, Xiao-Hua Zhou, and Peng Wu. 2023. Multiple Robust Learning for Recommendation. In *AAAI Conference on Artificial Intelligence*.

[8] Haoxuan Li, Yan Lyu, Chunyuan Zheng, and Peng Wu. 2023. TDR-CL: Targeted Doubly Robust Collaborative Learning for Debiased Recommendations. In *International Conference on Learning Representations*.

[9] Haoxuan Li, Kunhan Wu, Chunyuan Zheng, Yanghao Xiao, Hao Wang, Zhi Geng, Fuli Feng, Xiangnan He, and Peng Wu. 2023. Removing Hidden Confounding in Recommendation: A Unified Multi-Task Learning Approach. In *Advances in Neural Information Processing Systems*.

[10] Haoxuan Li, Yanghao Xiao, Chunyuan Zheng, and Peng Wu. 2023. Balancing Unobserved Confounding with a Few Unbiased Ratings in Debiased Recommendations. In *International World Wide Web Conference*.

[11] Haoxuan Li, Chunyuan Zheng, and Peng Wu. 2023. StableDR: Stabilized Doubly Robust Learning for Recommendation on Data Missing Not at Random. In *International Conference on Learning Representations*.

[12] Haoxuan Li, Chunyuan Zheng, Yanghao Xiao, Peng Wu, Zhi Geng, Xu Chen, and Peng Cui. 2024. Debiased Collaborative Filtering with Kernel-based Causal Balancing. In *International Conference on Learning Representations*.

[13] Xiang Li, Yanghao Xiao, Chunyuan Zheng, Qian Zou, Bing Cheng, Wei Lin, Haoxuan Li, and Zhouchen Lin. 2026. Optimizing Marketing Subsidies via Counterfactual Learning with Asymmetric Reward Function. In *International SIGIR Conference on Research and Development in Information Retrieval*.

[14] Yuening Li, Diego Uribe, Chuan He, Jiaxi Tang, Qingyun Liu, Junjie Shan, Ben Most, Kaushik Kalyan, Shuchao Bi, Xinyang Yi, et al. 2024. Short-form Video Needs Long-term Interests: An Industrial Solution for Serving Large User Sequence Models. In *ACM Conference on Recommender Systems*.

[15] Yozen Liu, Xiaolin Shi, Lucas Pierce, and Xiang Ren. 2019. Characterizing and Forecasting User Engagement with In-App Action Graph: A Case Study of Snapchat. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

[16] Christos Louizos, Uri Shalit, Joris M Mooij, David Sontag, Richard Zemel, and Max Welling. 2017. Causal effect inference with deep latent-variable models. *Advances in Neural Information Processing Systems* 30 (2017).

[17] Yan Lyu, Haoxuan Li, Tianyu Xia, Xiang Li, Xiangnan Feng, Chunyuan Zheng, and Xiao-Hua Zhou. 2026. Hierarchical Denoising Entire Space Multi-Task Model for Post-Click Conversion Rate Prediction with Noisy Labels. In *International SIGIR Conference on Research and Development in Information Retrieval*.

[18] Nijia Mo, Jianxiang Zang, Zhan Wang, and Hui Liu. 2025. DDualSE: Decoupled Dual-Head Squeeze and Excitation Attention for Sequential Recommendation. In *ACM International Conference on Web Search and Data Mining*.

[19] John Mullahy. 1986. Specification and Testing of Some Modified Count Data Models. *Journal of econometrics* 33, 3 (1986), 341–365.

[20] Xiangju Qin, Pádraig Cunningham, and Michael Salter-Townshend. 2016. Online Trans-dimensional von Mises-Fisher Mixture Models for User Profiles. *Journal of Machine Learning Research* 17, 200 (2016), 1–51.

[21] Zhen Qin, Yicheng Cheng, Zhe Zhao, Zhe Chen, Donald Metzler, and Jingzheng Qin. 2020. Multitask Mixture of Sequential Experts for User Activity Streams. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

[22] Kira Radinsky, Krysta Svore, Susan Dumais, Jaime Teevan, Alex Bocharov, and Eric Horvitz. 2012. Modeling and Predicting Behavioral Dynamics on the Web. In *International Conference on World Wide Web*.

[23] Piotr Rzepakowski and Szymon Jaroszewicz. 2012. Decision trees for uplift modeling with single and multiple treatments. *Knowledge and Information Systems* 32, 2 (2012), 303–327.

[24] Sherry Sahebi, Mengfan Yao, Siqian Zhao, and Reza Feyzi Behnagh. 2024. MoMENt: Marked Point Processes with Memory-Enhanced Neural Networks for User Activity Modeling. *ACM Trans. Knowl. Discov. Data* 18, 6 (2024), 32 pages.

[25] David C. Schmittlein, Donald G. Morrison, and Richard Colombo. 1987. Counting Your Customers: Who-Are They and What Will They Do Next? *Management Science* 33, 1 (1987), 1–24.

[26] Qijie Shen, Yuanchen Bei, Zihong Huang, Jialin Zhu, Keqin Xu, Boya Du, Jiawei Tang, Yuning Jiang, Feiran Huang, Xiao Huang, et al. 2025. AliBoost: Ecological Boosting Framework in Alibaba Platform. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

[27] Xiaogang Su, Joseph Kang, Juanjuan Fan, Richard A Levine, and Xin Yan. 2012. Facilitating score and causal inference trees for large observational studies. *The Journal of Machine Learning Research* 13, 1 (2012), 2955–2994.

[28] Zhizhong Wan, Bin Yin, Junjie Xie, Fei Jiang, Xiang Li, and Wei Lin. 2024. LARR: Large Language Model Aided Real-Time Scene Recommendation with Semantic Understanding. In *ACM Conference on Recommender Systems*.

[29] Daheng Wang, Meng Jiang, Munira Syed, Oliver Conway, Vishal Juneja, Sriram Subramanian, and Nitesh V Chawla. 2020. Calendar Graph Neural Networks for Modeling Time Structures in Spatiotemporal User Behaviors. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

[30] Hao Wang, Zhichao Chen, Zhaoran Liu, Haozhe Li, Degui Yang, Xinggao Liu, and Haoxuan Li. 2024. Entire Space Counterfactual Learning for Reliable Content Recommendations. *IEEE Transactions on Information Forensics and Security* (2024).

[31] Hao Wang, Zhichao Chen, Honglei Zhang, Zhengnan Li, Licheng Pan, Haoxuan Li, and Mingming Gong. 2025. Debiased recommendation via wasserstein causal balancing. *ACM Transactions on Information Systems* 43, 6 (2025), 1–24.

[32] Jun Wang, Haoxuan Li, Chi Zhang, Dongxu Liang, Enyun Yu, Wenwu Ou, and Wenjia Wang. 2023. CounterCLR: Counterfactual contrastive learning with non-random missing data in recommendation. In *IEEE International Conference on Data Mining*.

[33] Xiaojing Wang, Tianqi Liu, and Jingang Miao. 2019. A Deep Probabilistic Model for Customer Lifetime Value Prediction. *arXiv preprint arXiv:1912.07753* (2019).

[34] Anpeng Wu, Haoxuan Li, Chunyuan Zheng, Kun Kuang, and Kun Zhang. 2025. Classifying Treatment Responders: Bounds and Algorithms. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

[35] Peng Wu, Haoxuan Li, Yuhao Deng, Wenjie Hu, Quanyu Dai, Zhenhua Dong, Jie Sun, Rui Zhang, and Xiao-Hua Zhou. 2022. On the Opportunity of Causal Learning in Recommendation Systems: Foundation, Estimation, Prediction and Challenges. In *International Joint Conferences on Artificial Intelligence*.

[36] Xue Xia, Saurabh Joshi, Kousik Rajesh, Kangnan Li, Yangyi Lu, Nikil Pancha, Dhruvil Badani, Jiajing Xu, and Pong Eksombatchai. 2025. TransAct V2: Lifelong User Action Sequence Modeling on Pinterest Recommendation. In *ACM International Conference on Information and Knowledge Management*.

[37] Yanghao Xiao, Hao Wang, Xiang Li, Qian Zou, Bing Cheng, Wei Lin, Haoxuan Li, and Zhouchen Lin. 2026. Debiased Recommendation Beyond the Positive Propensity Assumption. In *International SIGIR Conference on Research and Development in Information Retrieval*.

[38] Yidan Xing, Haoqi Zhang, Yangsu Liu, Dagui Chen, Zhenzhe Zheng, and Fan Wu. 2025. An Auto-Coupon Framework for Advertiser Retention in Online Advertising. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

[39] Chunxu Zhang, Guodong Long, Hongkuan Guo, Zhaojie Liu, Guorui Zhou, Zijian Zhang, Yang Liu, and Bo Yang. 2025. Multifaceted User Modeling in Recommendation: A Federated Foundation Models Approach. In *AAAI Conference on Artificial Intelligence*.

[40] Chao Zhang, Shiwei Wu, Haoxin Zhang, Tong Xu, Yan Gao, Yao Hu, and Enhong Chen. 2024. NoteLLM: A Retrievable Large Language Model for Note Recommendation. In *Companion Proceedings of the ACM Web Conference*.

[41] Xu Zhao, Ruibo Ma, Jiaqi Chen, Weiqi Zhao, Ping Yang, and Yao Hu. 2025. Multi-Granularity Distribution Modeling for Video Watch Time Prediction via Exponential-Gaussian Mixture Network. In *ACM Conference on Recommender Systems*.

[42] Chunyuan Zheng, Anpeng Wu, Chuan Zhou, Taojun Hu, Qingying Chen, Hongyi Liu, Chenxi Li, Huiyou Jiang, Haoxuan Li, and Zhouchen Lin. 2026. Uplift Modeling with Delayed Feedback: Identifiability and Algorithms. In *AAAI Conference on Artificial Intelligence*.

[43] Chunyuan Zheng, Haocheng Yang, Haoxuan Li, and Mengyue Yang. 2025. Unveiling Extraneous Sampling Bias with Data Missing-Not-At-Random. In *Advances in Neural Information Processing Systems*.

[44] Kailiang Zhong, Fengtong Xiao, Yan Ren, Yaorong Liang, Wenqing Yao, Xiaofeng Yang, and Ling Cen. 2022. DESCN: Deep Entire Space Cross Networks for Individual Treatment Effect Estimation. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

[45] Chuan Zhou, Yaxuan Li, Chunyuan Zheng, Haiteng Zhang, Min Zhang, Haoxuan Li, and Mingming Gong. 2025. A Two-Stage Pretraining-Finetuning Framework for Treatment Effect Estimation with Unmeasured Confounding. *arXiv preprint arXiv:2501.08888* (2025).

[46] Chuan Zhou, Lina Yao, Haoxuan Li, and Mingming Gong. 2025. Counterfactual Implicit Feedback Modeling. In *Advances in Neural Information Processing Systems*.