A Simple and Robust Ensemble For Click-Through Rate Prediction

XINGMEI WANG, University of Science and Technology of China, China
YANKAI WANG, University of Science and Technology of China, China

This paper presents the strategy employed by the TOT team, which secured 3rd place in the ACM RecSys Challenge 2023 among academic teams. The challenge, orchestrated by ShareChat, focuses on online advertising and user privacy. The objective is to predict the probability of an application install for each anonymized entry. Our approach, though simple, proves efficient and comprises two primary components: initially, we explore state-of-the-art single-task and multi-task models, and subsequently, we integrate these via a simple yet robust ensemble.

CCS Concepts: • **Information systems** → **Recommender systems**.

Additional Key Words and Phrases: recommder systems, deep learning, neural networks

ACM Reference Format:
Xingmei Wang and Yankai Wang. 2023. A Simple and Robust Ensemble For Click-Through Rate Prediction. In *RecSysChallenge23, September 18–22, 2023, Singapore.* ACM, New York, NY, USA, 6 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Online advertising delivers personalized ads to users by recommender systems, which analyze user data, such as browsing history, past purchases, and demographics, to present users with ads tailored to their preferences. Click-through rate (CTR) prediction is a general way to provide personalized recommendations. It estimates the probability of a user clicking on a specific ad, thus ensuring that the displayed ads are those most likely to engage the user.

Nonetheless, a distinct tension exists in the dichotomy between personalization and privacy. Personalized advertising, on one side, requires the collection and analysis of substantial user data. This data-driven approach can enhance ad relevance, boost user engagement and satisfaction, and escalate both click-through and conversion rates.

Conversely, privacy has become an increasingly salient issue. Users harbor valid apprehensions about the collection, storage, and utilization of their personal data. Over-personalization may induce discomfort among users, leading them to feel as though their privacy has been breached. Moreover, if user data is not managed correctly during its collection, storage, or use, it may result in data leakage or misuse.

With a focus on user privacy, the ACM RecSys Challenge 2023 [1] provides a real-world ad dataset from the Sharechat and Moj apps with all features anonymized, to encourage novel algorithms for CTR prediction.

This paper will present the solution utilized by the TOT team, which achieved 3rd place amongst academic teams in the RecSys Challenge 2023. The solution is an ensemble that effectively incorporates DCNv2 [17], HardShare [13], PLE [16] and SENet [5]. Our code is available in https://github.com/pepsi2222/RecSys-Challenge-23.

[1] http://www.recsyschallenge.com/2023/

Manuscript submitted to ACM

2 TASK FORMULATION

The task at hand is a conventional CTR prediction, wherein each entry signifies an ad impression presented to a user. Each entry comprises the date, personal features, ad features and whether it resulted in a click on the ad and subsequently an install or not. Personal features encompass demographic features, count features of historical interactions, content preference embeddings and app affinity embeddings. Ad features embody characteristic features, video and image content. Notably, no semantics of these features is provided.

The training dataset contains 3.4 million entries from 22 consecutive preceding days, with the objective of predicting the probability of installation on the 23rd day, when the click and installation are absent. Moreover, the possibility of an installation occurring without an associated ad click exists within the given scenario.

The evaluation metric, Logloss, is defined as:

$$Logloss = -\frac{1}{N}\sum_{i=1}^{N} y_i \log(p_i) + (1 - y_i)\log(1 - p_i) \tag{1}$$

where N represents the number of entries in the test set, $y_i \in \{0, 1\}$ and p_i correspond to the ground truth and estimated probability of an installation for entry i, respectively.

3 DATA PREPARATION

3.1 Data preprocessing

Given that the out-of-vocabulary ratio for certain numerical features within the test set is minimal—most are less than 0.001—we subsequently process them as categorical features. This reclassification of data types allows us to glean more precise embeddings.

3.2 Feature Engineering

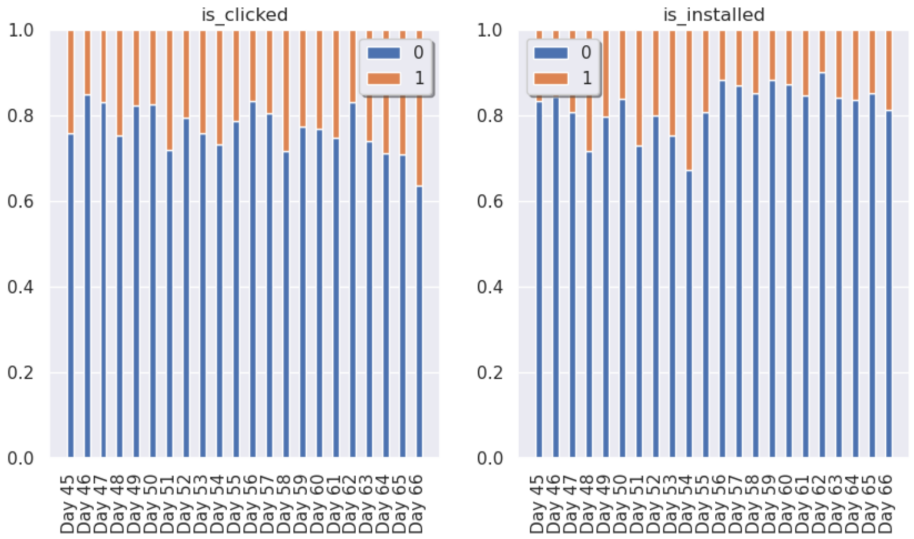

Fig. 1. The 0/1 ratios of labels over time.

Table 1. Statistics of dataset partition.

	Date	# Entry
Training	45-65	3387880
Validation	66	97972
Testing	67	160973

3.2.1 Feature Creation. Anonymized features lose their semantic meaning and interpretation, thereby complicating the manual design of useful features for the target. Many activities exhibit weekly cycles, and this weekly seasonality, if present, can serve as a valuable input feature to improve the accuracy of time-series forecasting models. As demonstrated in Figure 1, we observed the trends over time, which indeed displayed certain weekly patterns resembling the letter "M". Therefore, we grouped the dates by the day of the week they fall on as a new feature.

3.2.2 Feature Selection. To reduce the complexity of models, we exclude the feature with only a single value. Furthermore, when faced with features that have a Pearson correlation coefficient of 1, we retain only one of them.

3.2.3 Feature Transformation. As all numerical features do not satisfy a normal distribution, we explore the application of Box-Cox transformations [1], but this does not yield a visible impact. For numerical features with long-tailed distributions, we also attempt log1p transformations [2], but the effect remains inconclusive. Finally, for most numerical features, we employed standard scalers—standardizing features by removing the mean and scaling to unit variance. This approach facilitates model training, accelerates convergence, and circumvents numerical instability.

3.3 Dataset Partition

Given that the requisite models are designed to analyze and forecast future events based on past data patterns and trends, we reserve the data of the penultimate day as the validation set. The detailed partitioning of the entire dataset is shown in Table 1.

4 METHODOLOGY

We tested models with *RecStudio* [3], which is a unified, highly-modularized and recommendation-efficient recommendation library based on PyTorch. Our method blends only the five most advanced ones, as depicted in Figure 2.

4.1 Meta Learners

4.1.1 Single-Task Models. We started with single-task models, among which the stacked DCNv2 [17] outperforms. It stacks the deep network on the cross network to capture both low- and high-order feature interactions. Part of experimental results is presented in Table 2.

4.1.2 Multi-Task Models. Since the training set provides click information, it enables the use of multi-task models. These models enhance performance by sharing information across tasks, thereby exploiting similarities and variations. Additionally, multi-task models mitigate the risk of overfitting by using the click task as a form of implicit regularization. Experimental results about multi-task models are shown in Table 3.

[2] $\log1p := \ln(1+x)$

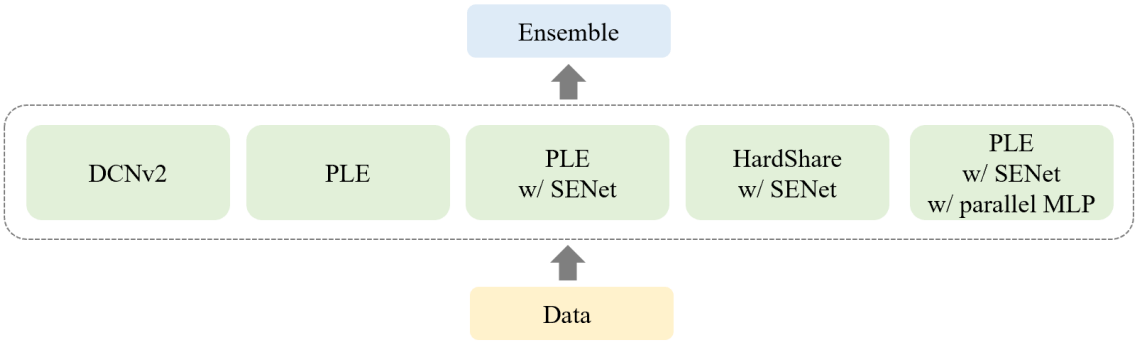

Fig. 2. The pipeline of the TOT method.

Table 2. Performance of single-task models in the validation set.

Model	Logloss
DCNv2 [17]	0.3688
DeepCrossing [14]	0.3745
DeepIM [7]	0.3822
EDCN [2]	0.3694
FwFM [11]	0.3820
NFM [4]	0.3724
PNN [12]	0.3813
xDeepFM [9]	0.3717

Table 3. Performance of multi-task models in the validation set.

Model	Logloss
AITM [18]	0.3944
HardShare [13]	0.3858
MMoE [10]	0.3791
PLE [16]	0.3745

4.1.3 SENet. The Squeeze-and-Excitation Network (SENet) [5], is a deep learning model introduced in 2017 that won the ImageNet competition. FiBiNET [6] later incorporated it into recommender systems, to highlight more useful features while disregarding others. SENet is a flexible component that can be integrated following the embedding layer, so we apply it to different models with *RecStudio* and observed outstanding results, especially in HardShare and PLE.

4.1.4 Parallel MLP. Many ranking algorithms incorporate an additional deep component, a simple parallel Multi-Layer Perceptron (MLP), such as AutoInt [15], DCNv2 [17], DeepIM [7], DESTINE [19], FiGNN [8], PPNet [3]. It's believed that a parallel MLP aids in modeling simple patterns and makes embeddings less prone to overfitting. Our experiments found that a parallel MLP has a significant effect on PLE with SENet.

[3]Proposed by Kuai in 2019.

Table 4. Performance of models in the final leaderboard.

Model	Logloss
DCNv2	6.269786
PLE	6.291424
PLE w/ SENet	6.275944
HardShare w/ SENet	6.245281
PLE w/ SENet and parallel MLP	6.247294
Naive Average	6.181586
Median	6.180274
Weighted Average	6.179481
Robust Average	**6.178771**

4.2 Blending

Having identified five superior models, we utilized blending to harness their strengths and mitigate their individual weaknesses— an approach that typically yields improved predictive performance. We explored several blending methods, the results are in Table 4. The robust average method is to remove the maximum and minimum probabilities and averaging the remaining ones. While we attempted to blend additional models, this approach resulted in a decline in performance. For RecSys Challenge 23, we adopted the robust average method for blending.

4.3 Implementation with *RecStudio*

RecStudio is a comprehensive framework supporting both retrievers and rankers, with 44 algorithms available for CTR prediction. This platform facilitates the swift implementation of pre-existing or custom-designed models with user-friendly features:

- Automated data processing workflow. Configuring the dataset necessitates only a modification to the relevant configuration file.
- Highly-modularized architectures. For the implementation of new algorithms, users can assemble modules through either drag-and-drop programming or automatic machine learning.
- Easy hyperparameter tuning. Users are only required to modify the model's corresponding configuration file when tuning.

5 CONCLUSION

In this paper, we outline our approach to the ACM RecSys Challenge 2023, a CTR prediction task with an emphasis on user privacy. We utilized both single-task and multi-task models, affirming the efficacy of the SENet and parallel MLP. Moreover, we proposed a simple yet robust blending technique that enhances performance without incurring additional computational cost. This simple yet effective model won 3rd place among academic teams in the final leaderboard.

REFERENCES

[1] George EP Box and David R Cox. 1964. An analysis of transformations. *Journal of the Royal Statistical Society Series B: Statistical Methodology* 26, 2 (1964), 211–243.

[2] Bo Chen, Yichao Wang, Zhirong Liu, Ruiming Tang, Wei Guo, Hongkun Zheng, Weiwei Yao, Muyu Zhang, and Xiuqiang He. 2021. Enhancing explicit and implicit feature interactions via information sharing for parallel deep ctr models. In *Proceedings of the 30th ACM international conference on information & knowledge management*. 3757–3766.

[3] Xiaolong Chen Jin Chen Yankai Wang Haoran Jin Rui Fan Xingmei Wang Zheng Liu Le Wu Defu Lian, Xu Huang and Enhong Chen. 2023. RecStudio: Towards a Highly-Modularized Recommender System. In *Proceedings of the 46th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*.

[4] Xiangnan He and Tat-Seng Chua. 2017. Neural factorization machines for sparse predictive analytics. In *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*. 355–364.

[5] Jie Hu, Li Shen, and Gang Sun. 2018. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7132–7141.

[6] Tongwen Huang, Zhiqi Zhang, and Junlin Zhang. 2019. FiBiNET: combining feature importance and bilinear feature interaction for click-through rate prediction. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 169–177.

[7] Yi Li, Gu Wang, Xiangyang Ji, Yu Xiang, and Dieter Fox. 2018. Deepim: Deep iterative matching for 6d pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 683–698.

[8] Zekun Li, Zeyu Cui, Shu Wu, Xiaoyu Zhang, and Liang Wang. 2019. Fi-gnn: Modeling feature interactions via graph neural networks for ctr prediction. In *Proceedings of the 28th ACM international conference on information and knowledge management*. 539–548.

[9] Jianxun Lian, Xiaohuan Zhou, Fuzheng Zhang, Zhongxia Chen, Xing Xie, and Guangzhong Sun. 2018. xdeepfm: Combining explicit and implicit feature interactions for recommender systems. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 1754–1763.

[10] 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.

[11] Junwei Pan, Jian Xu, Alfonso Lobos Ruiz, Wenliang Zhao, Shengjun Pan, Yu Sun, and Quan Lu. 2018. Field-weighted factorization machines for click-through rate prediction in display advertising. In *Proceedings of the 2018 World Wide Web Conference*. 1349–1357.

[12] Yanru Qu, Han Cai, Kan Ren, Weinan Zhang, Yong Yu, Ying Wen, and Jun Wang. 2016. Product-based neural networks for user response prediction. In *2016 IEEE 16th international conference on data mining (ICDM)*. IEEE, 1149–1154.

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

[14] Ying Shan, T Ryan Hoens, Jian Jiao, Haijing Wang, Dong Yu, and JC Mao. 2016. Deep crossing: Web-scale modeling without manually crafted combinatorial features. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 255–262.

[15] Weiping Song, Chence Shi, Zhiping Xiao, Zhijian Duan, Yewen Xu, Ming Zhang, and Jian Tang. 2019. Autoint: Automatic feature interaction learning via self-attentive neural networks. In *Proceedings of the 28th ACM international conference on information and knowledge management*. 1161–1170.

[16] 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.

[17] 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.

[18] 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.

[19] Yichen Xu, Yanqiao Zhu, Feng Yu, Qiang Liu, and Shu Wu. 2021. Disentangled self-attentive neural networks for click-through rate prediction. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 3553–3557.