Training-free Periodic Interest Augmentation in Incremental Recommendation

Heyuan Huang*
OPPO
Shenzhen, China
huangheyuan2@oppo.com

Xingyu Lou*
OPPO
Shenzhen, China
louxingyu@oppo.com

Changwang Zhang†
OPPO
Shenzhen, China
changwangzhang@foxmail.com

Chaochao Chen
Zhejiang University
Hangzhou, China
zjuccc@zju.edu.cn

Kuiyao Dong
OPPO
Shenzhen, China
dongkuiyao@oppo.com

Feng Lu
OPPO
Shenzhen, China
lufeng2@oppo.com

Han Lei
OPPO
Shenzhen, China
leihan@oppo.com

Yihao Wang
Zhejiang University
Hangzhou, China
yihaowang@zju.edu.cn

Wangchunshu Zhou
OPPO
Shenzhen, China
zhouwangchunshu@oppo.com

Jun Wang†
OPPO
Shenzhen, China
junwang.lu@gmail.com

Abstract

Industrial recommender systems usually train models incrementally to grasp recent interests of users. However, a fundamental issue of these incremental updated models is their tendency to overfit current data while neglecting past information. Specifically, we have observed that the data distribution of real systems exhibits periodic drifts, leading to periodic fluctuations of prediction bias. To alleviate the above bias fluctuations while minimizing the loss of recent interests, we propose TPIA, a **T**raining-free approach for **P**eriodic **I**nterest **A**ugmentation in incremental recommendation. Specifically, after the latest model is trained, we first calculate the importance score of each model in the previous period. Then, we merge these models based on the importance scores. To minimize information loss due to interference of parameters during model merging, we further develop a method for trimming redundant and abnormal parameters. Offline experiments on both public and private datasets demonstrate the effectiveness of TPIA. It has also been deployed on a large-scale industrial recommender system, and has shown a notable 1.61% increase in CVR and a 1.97% increase in CPM, along with enhanced stability in prediction bias.

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

SIGIR '25, Padua, Italy

ACM ISBN 979-8-4007-1592-1/2025/07
https://doi.org/10.1145/3726302.3730258

CCS Concepts

• **Information systems** → **Recommender systems**.

Keywords

Recommendation, Periodic Forgetting, Model Merging

ACM Reference Format:
Heyuan Huang, Xingyu Lou, Changwang Zhang, Chaochao Chen, Kuiyao Dong, Feng Lu, Han Lei, Yihao Wang, Wangchunshu Zhou, and Jun Wang. 2025. Training-free Periodic Interest Augmentation in Incremental Recommendation. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '25), July 13–18, 2025, Padua, Italy*. ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3726302.3730258

1 Introduction

Recommender system (RS) is a technology that provides personalized information services [3, 9, 12]. To swiftly grasp recent interests of users, internet applications increasingly implement incremental model updates in RS [6, 15]. However, a fundamental issue of these incrementally updated models is their tendency to overfit current data while neglecting past information [4, 8, 11]. As an illustration, we observe that the data distribution drift periodically in our production system (Fig. 1(left)), which is essentially caused by periodic interests of users. Existing RSs fail to effectively learn from the drift, leading to periodic fluctuations in the system's prediction bias (Fig. 1(right)), which not only hurts user experience but also damages the benefits of advertisers and advertising systems [16]. In this paper, we refer to this phenomenon as **Periodic Forgetting**.

To alleviate the periodic forgetting while minimizing the loss of recent interests, there are two main challenges: **CH1**: *How to mine periodic and recent interests better?* Existing works mine periodic

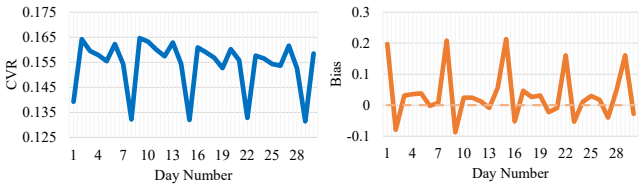

Figure 1: Periodic data distribution drift (left) and prediction bias fluctuation (right) in our production system.

interests from data within the same phase of historical periods (e.g., Sundays in different weeks) and recent interests from the latest data [1, 21]. However, users' interests change gradually. Taking the APP store as an example, users show continuous periodic interest in gaming apps around weekends. Similarly, when a popular app launches, users exhibit sustained interest in it. Thus, mining information over a broader time window can further enrich both periodic and recent interests. **CH2**: *How to utilize periodic and recent interests better?* Existing works introduce periodicity into the latest model in four ways: rehearsal [1, 15], periodic features [22, 23], parameter isolation [11, 21] and calibration [1]. However, they are either ineffective due to periodic interest overwhelming recent interest and overfitting or inefficient because of the need to retrain historical data and retain additional parameters.

To overcome the above challenges, we propose TPIA, a Training-free approach for Periodic Interest Augmentation in incremental recommendation. Specifically, we define the previous period as the time interval of one observed period preceding the latest data. For example, in Fig. 1, the data distribution drifts weekly. Therefore, When training the model on day 7, the data within a week preceding it (i.e., Days 1 to 7) are designated as the previous period. For **CH1**, we use the first and last day of this period to represent periodic and recent interests, respectively. By calculating the data distribution similarity between these two days and the other days in the previous period, we can figure out the importance scores of each model. For **CH2**, we leverage the importance scores as a guide, and employ the model merging technique [13, 17–20] to merge models in the previous period, which allows a single model to cover both periodic and recent interests. To effectively reduce the information loss caused by parameter interference during model merging, we further develop a method for trimming redundant and abnormal parameters. In short, TPIA can be integrated into existing incremental models. It alleviates the periodic forgetting without training or extra inference time, providing a practical and efficient solution to the challenges faced by incremental recommendation.

Our contributions can be summarized as follows: 1) We define the phenomenon of model forgetting periodic interests as periodic forgetting and thus propose TPIA, a Training-free approach for Periodic Interest Augmentation in incremental recommendation. 2) We design a model merging method to merge models in the previous period, thereby alleviating the periodic forgetting without requiring additional training steps or extra inference time. To the best of our knowledge, this is the first work to apply model merging to alleviate the periodic forgetting. 3) We conduct offline experiments and online A/B testing to verify that TPIA can improve the model's effectiveness while stabilizing the prediction bias.

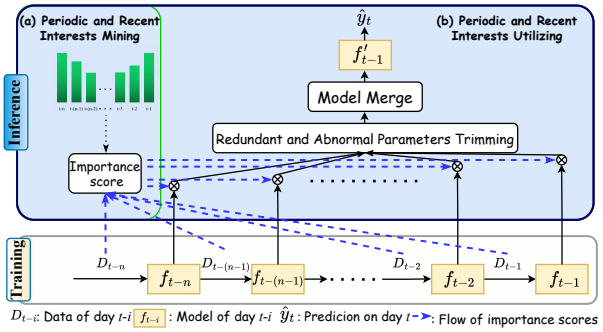

Figure 2: Overall Framework of TPIA. TPIA can be plugged into the inference. Specifically, it guides model merging by importance scores, while trims redundant/abnormal parameters to mine and utilize periodic and recent interests.

2 METHODOLOGY

2.1 Problem Formulation

In this paper, we are interested in the post-click conversion rate (CVR) prediction task. Formally, we take the data as $\mathcal{A}(x_t, y_t)$, where x_t is samples from day t, $y_t \in \{0, 1\}$ indicates whether the conversion occurs. The CVR prediction can be modeled as: $\hat{y}_t = f_{t-1}(x_t)$, where f_{t-1} is the model trained on the latest data from day $t-1$, $\hat{y}_t$ is the prediction of f_{t-1} on day t. $\mathbb{F} = \{f_{t-n}, ..., f_{t-2}, f_{t-1}\}$ represents the model set in the previous period, where n represents the duration of a period. As shown in Fig. 1, since the observed fluctuations in the data distribution are weekly in our production system, n defaults to 7 in this paper. Our objective is to merge the periodic and recent interests from models in $\mathbb{F}$ into a single model f'_{t-1} while minimizing information loss.

2.2 Overall Framework

The overall framework of TPIA is shown in Fig. 2. In training, there is a CVR prediction model with daily incremental updates, note that TPIA does not affect it. In inference, TPIA calculates the data distribution similarity between the first/last day (representing periodic/recent interests) and the other days in the previous period, then uses them as importance scores of each model to better mine periodic and recent interests (part (a) of Fig. 2). Next, the importance scores are used to guide the model merging, and further trim redundant and abnormal parameters to reduce parameter interference during merging models. Therefore, we can better utilize periodic and recent interests, allowing them to coexist in a single model while minimizing information loss (part (b) of Fig. 2).

2.3 Periodic and Recent Interests Mining

Firstly, we design a method to fully mine periodic and recent interests based on the observed fluctuation period of the data distribution. As an example, when predicting for day t, day $t-n$ and day $t-1$ are used to represent periodic and recent interests, respectively. We argue that since the incremental model overfits current data, not only information on the day $t-n$, but also on the other days of the previous period may be forgotten, which will lead to the loss of periodic and recent interests. Therefore, we try to score the

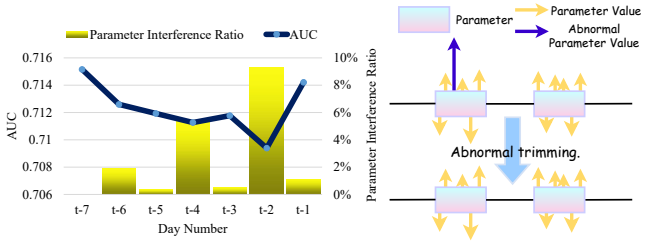

Figure 3: (left) AUC of models in the previous period and the interference ratio of each difference matrix. (right) Illustration of abnormal parameter trimming. Arrows' direction and length represent parameters' sign and magnitude.

importance of each model in the previous period from periodic and recent interests perspectives, respectively. Formally, we begin by calculating the Kullback-Leibler (KL) Divergence [10] between day $t-1$ and each of the other days in the previous period, and then repeat the same process for day $t-n$:

$$D_{t-i,t-j} = \mathcal{B}(x_{t-j}, y_{t-j}) log(\frac{\mathcal{B}(x_{t-j}, y_{t-j})}{\mathcal{B}(x_{t-i}, y_{t-i})}), \tag{1}$$

where $\mathcal{B}(x_{t-i}, y_{t-i})$ and $\mathcal{B}(x_{t-j}, y_{t-j})$ represents the data distribution on day $t-i$ and $t-j$. $i \in \{1, ..., n\}$ represents each day in the previous period, $j \in \{1, n\}$ represents periodic and recent interests respectively. Next, we sum periodic interest score $D_{t-i,t-n}$ and recent interest score $D_{t-i,t-1}$ of each model, and obtain their importance scores through the softmax layer:

$$\alpha_i = Softmax(1 - (D_{t-i,t-1} + D_{t-i,t-n})), \tag{2}$$

where α_i represents the importance score of f_{t-i}. These scores guide us to fully mine periodic and recent interests in each model.

2.4 Periodic and Recent Interests Utilizing

In this section, we will focus on how to better utilize periodic and recent interests. Specifically, we first merge models as follows:

$$\begin{aligned} f'_{t-1} &= \alpha_n * f_{t-n} + \alpha_{n-1} * f_{t-(n-1)} + ... + \alpha_1 * f_{t-1} \\ &= \alpha_n * f_{t-n} + \alpha_{n-1} * (f_{t-n} + \Delta f_{n-1}) + ... + \alpha_1 * (f_{t-n} + \Delta f_1). \end{aligned} \tag{3}$$

We further regard the model f_{t-i} as the sum of the backbone model f_{t-n} and the difference matrix Δf_i, that is, $\Delta f_i = f_{t-i} - f_{t-n}$. Model merging is actually a regularization of parameters. Through weighted sum, the optimization direction of parameters is pulled to the common space that best balances periodic and recent interests.

However, the difference matrix contains a large number of redundant parameters [20]. Retaining them may lead to interference during model merging, that is, the beneficial parameter value may be obscured by the redundant one, thereby reducing the overall model performance. Therefore, similar to [18], we only retain parameters with top-k% amplitude values and trim the rest (i.e., set them to zero). The formula is as follows:

$$\Delta \tilde{f}_i = W_k \odot \Delta f_i, \tag{4}$$

where W_k represents the de-redundant mask matrix, which is used to trim the parameters in the corresponding difference matrix whose values are below the top-k%. k is a hyperparameter and $\odot$ is the element-wise product.

In addition, we still find that there are serious parameter interference between difference matrices. As shown in the bar graph in Fig. 3(left), some difference matrices have a high proportion of parameters that are significantly differ from others. When directly using each model in the previous period to predict day t, the model corresponding to the difference matrix with a higher interference ratio has a lower AUC (line chart in Fig. 3(left)), which may be caused by random anomalies. Simply merging these parameters with others is suboptimal. So, we trim them as shown in Fig. 3(right). Formally, we create a de-anomaly mask matrix $W_{\Delta f_i}$ as follows:

$$W^{ab}_{\Delta f_i} = \begin{cases} 0 & \text{if} \Delta f^{ab}_i - min(\Delta f^{ab}_{oth}) > \beta * max(\Delta f^{ab}_{oth} - min(\Delta f^{ab}_{oth})) \\ 1 & \text{else} \end{cases}, \tag{5}$$

where Δf^{ab}_i represents the parameter at the position (a, b) in Δf_i, Δf_{oth} represents the difference matrix corresponding to models in $\mathbb{F}$ except f_i, β is a hyperparameter that represents the threshold for identifying interference caused by anomaly parameters. Next, we obtain the final difference matrix $\Delta \hat{f}_i$ as: $\Delta \hat{f}_i = W_{\Delta f_i} \odot \Delta \tilde{f}_i$.

Finally, we use $\Delta \hat{f}_i$ to replace Δf_i in Eq. (3), and obtain f'_{t-1} for predicting on the day t: $\hat{y}_t = f'_{t-1}(x_t)$. Since f'_{t-1} incorporates both periodic and recent interests effectively, it can alleviate the periodic forgetting while minimizing recent interests loss, ultimately improving the prediction performance while stabilizing its bias.

3 Experiments

3.1 Experimental Setup

3.1.1 ***Datasets.*** We conduct offline experiments on a private dataset and a public dataset Alimama[1]: **Private Dataset.** We collect in our production system from 2024/11/01 to 2024/12/01. It contains features of about 200 user fields, 160 item fields and 120 content fields. We model the conversion type "retention" (if the user stays the next day after download) with severe periodic data distribution drift. **Alimama.** Provided by Alibaba. We model the purchase behavior that has periodic data distribution drift in logs from 2017/04/22 to 2017/05/13. The last day is used as the test set in both datasets.

3.1.2 ***Evaluation Metrics & Settings.*** We use AUC and bias as metrics and report the average results over 5 runs. AUC is used to measure the prediction accuracy, and bias is used to measure the difference between the predicted value $pcvr$ and the actual value cvr, which can be formally calculated as: $Bias = pcvr/cvr - 1$. Note that a larger AUC is better, while a bias closer to 0 is better. Furthermore, we utilize the Adam [7] optimizer with a learning rate of 0.001 and set the number of epochs to 1. We tune the trimming ratio of redundant parameters k among {0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9}, and tune the threshold for identifying anomalies β among {1,2,5,10,50}.

3.1.3 ***Baselines.*** We use the following four categories of industry-common method to alleviate the periodic forgetting as baselines: **Rehearsal** (Replay, Fine-tuning), **Periodic features** (Interest Clock [22]), **Parameter Isolation** (ASYS [11]), **Calibration** (HDR [1]). Replay means using a dataset containing historical and latest samples for training. Fine-tuning means using samples from the same phase of previous period to fine-tune the latest model. Interest

[1] https://tianchi.aliyun.com/dataset/dataDetail?dataId=56

Table 1: Overall Experiment. ***The best results are in bold and the second best results are underlined.***

Method	Private						Alimama					
	DCN		DeepFM		Wide&Deep		DCN		DeepFM		Wide&Deep	
	AUC*(Impr)*	Bias	AUC*(Impr)*	Bias	AUC*(Impr)*	Bias	AUC*(Impr)*	Bias	AUC*(Impr)*	Bias	AUC*(Impr)*	Bias
Base	0.8266*(+0.00%)*	0.1589	0.8247*(+0.00%)*	0.1984	0.8242*(+0.00%)*	0.2675	0.7142*(+0.00%)*	0.0851	0.7144*(+0.00%)*	0.0963	0.7142*(+0.00%)*	0.3125
Replay	0.8197*(-0.69%)*	0.1052	0.8175*(-0.72%)*	0.1265	0.8179*(-0.53%)*	0.1196	0.7128*(-0.14%)*	-0.1244	0.7124*(-0.20%)*	0.0650	0.7141*(-0.01%)*	0.0599
Fine-tuning	0.8234*(-0.32%)*	**0.0527**	0.8210*(-0.37%)*	-0.1029	0.8224*(-0.18%)*	**-0.0535**	0.7077*(-0.65%)*	-0.0252	0.7143*(-0.01%)*	**0.0072**	0.7032*(-1.10%)*	-0.0647
Interest Clock	0.8262*(-0.04%)*	0.0941	0.8270*(+0.24%)*	0.1550	0.8249*(+0.07%)*	0.1526	0.7163*(+0.21%)*	0.0256	0.7153*(+0.09%)*	0.0929	0.7169*(+0.27%)*	0.0396
ASYS	0.8284*(+0.17%)*	0.1200	0.8259*(+0.17%)*	0.2058	0.8261*(+0.14%)*	0.2449	0.7165*(+0.23%)*	0.0590	0.7148*(+0.04%)*	-0.0428	0.7154*(+0.12%)*	-0.0693
HDR	0.8312*(+0.46%)*	0.0586	0.8278*(+0.36%)*	0.0596	0.8254*(+0.07%)*	0.1722	0.7178*(+0.36%)*	-0.0476	0.7165*(+0.21%)*	0.0345	0.7159*(+0.17%)*	0.2167
TPIA	**0.8335*(+0.69%)***	0.0618	**0.8303*(+0.61%)***	**0.0594**	**0.8317*(+0.70%)***	0.0582	**0.7196*(+0.54%)***	**0.0045**	**0.7196*(+0.52%)***	0.0585	**0.7171*(+0.29%)***	**-0.0223**

Table 2: Ablation experiment on Alimama dataset. ***The best results are in bold and the second best results are underlined.***

Exp.	Model	AUC*(Impr)*	Bias
1	Base (DCN)	0.7142*(+0.00%)*	0.0851
2	TPIA (Model on day $t-7$ & $t-1$)	0.7174*(+0.32%)*	0.0554
3	TPIA (w/o Importance score)	0.7182*(+0.40%)*	**0.0024**
4	TPIA (Trim redundant parameters randomly)	0.7155*(+0.13%)*	0.0488
5	TPIA (w/o Trim redundant parameters)	0.7190*(+0.48%)*	0.0199
6	TPIA (w/o Trim abnormal parameters)	0.7191*(+0.49%)*	-0.0050
7	TPIA	**0.7196*(+0.54%)***	0.0045

Clock, ASYS, and HDR are recent works that model the periodic forgetting at different temporal granularities. We modify them to adapt to the weekly periodic forgetting we face. In addition, to verify the impact of plugging TPIA into different models, we used DCN [14], DeepFM [5], and Wide&Deep [2] as backbone, respectively.

3.2 Offline Experiments

3.2.1 ***Overall Experiments.*** As shown in Table 1: 1) TPIA enhances the AUC and bias on both datasets across multiple backbones by alleviating periodic forgetting while preserving recent interests. 2) Compared with other methods, TPIA works best in most cases. This is because it can better mine periodic and recent interests. Note that Fine-tuning performs well in bias, which is attributed to its explicit augmentation of periodicity. In addition to the metrics in Table 1, it is worth noting that TPIA requires no additional training steps or extra inference time when compared with other methods above. Therefore, it can be efficiently integrated with existing online incremental models.

3.2.2 ***Ablation Experiments.*** Table 2 presents the experiments that verify the effects of various variants of TPIA: 1) Exp.2 performs better than Exp.1. Exp.1 only uses the model from day $t-1$ for prediction, while Exp.2 merges the model parameters from day $t-7$ and day $t-1$. This highlights the necessity of introducing periodic interest. 2) Exp.7 performs significantly better than Exp.2, indicating that mining information over a broader time window can further enrich both periodic and recent interests. 3) Exp.3, 5~7 indicate that importance scores and the trimming of abnormal/redundant parameters can further improve the effect. Exp.4 shows that trim redundant parameters randomly [20] may result in more information loss compared to only trimming parameters with small values.

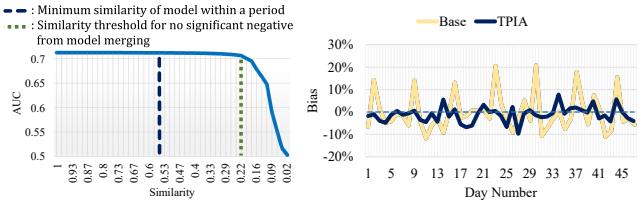

Figure 4: (left) Prerequisite verification experiment of TPIA. (right) Bias of baseline and TPIA in online experiment.

3.2.3 ***Validity Verification Experiments.*** We conduct experiments to illustrate why TPIA is effective in incremental recommendation. As shown in Fig. 4(left), the horizontal axis represents the cosine similarity of the day $t-1$'s model before and after adding noise of different amplitudes, and the vertical axis represents the AUC of the model obtained by merging the aforementioned models. We can find that when the similarity is lower than 0.22, the AUC drops sharply. However, in incremental training, since the model is initialized by the previous day's model, they usually have high similarity (Greater than 0.54 in our experiments). Therefore, model merging in TPIA will not have a significant negative impact but may also be beneficial by combining periodic and recent interests.

3.3 Online A/B Testing

We conduct online A/B testing on a large-scale industrial recommender system, the App Store homepage. We mainly focused on the indicators of CVR, Cost Per Mille (CPM), and bias. The testing lasted for 48 days from 2024/11/22 to 2025/01/08. We integrate TPIA with the online baseline which has been incrementally trained for an extremely long time (more than one year), and achieve improvements of 1.61% in CVR and 1.97% in CPM. In addition, as shown in Fig. 4(right), most of the time, the bias of TPIA is more stable. Overall, the number of instances where the bias falls within ±20% has increased by 23%. TPIA has now been deployed on the App Store homepage, serving tens of millions of users daily.

4 Conclusion

In this paper, we propose a Training-free approach for Periodic Interest Augmentation in incremental recommendation named TPIA, which can efficiently mine and utilize periodic and recent interests, thereby alleviating the periodic forgetting. Offline and online experiments verify the superiority of TPIA.

References

[1] Zhangming Chan, Yu Zhang, Shuguang Han, Yong Bai, Xiang-Rong Sheng, Siyuan Lou, Jiacen Hu, Baolin Liu, Yuning Jiang, Jian Xu, et al. 2023. Capturing conversion rate fluctuation during sales promotions: A novel historical data reuse approach. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 3774–3784.

[2] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, Rohan Anil, Zakaria Haque, Lichan Hong, Vihan Jain, Xiaobing Liu, and Hemal Shah. 2016. Wide & Deep Learning for Recommender Systems *(DLRS 2016)*. Association for Computing Machinery, New York, NY, USA, 7–10. doi:10.1145/2988450.2988454

[3] Zeshan Fayyaz, Mahsa Ebrahimian, Dina Nawara, Ahmed Ibrahim, and Rasha Kashef. 2020. Recommendation Systems: Algorithms, Challenges, Metrics, and Business Opportunities. *Applied Sciences* 10, 21 (2020). doi:10.3390/app10217748

[4] Robert M French. 1999. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences* 3, 4 (1999), 128–135.

[5] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: a factorization-machine based neural network for CTR prediction. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence* (Melbourne, Australia) *(IJCAI'17)*. AAAI Press, 1725–1731.

[6] Petros Katsileros, Nikiforos Mandilaras, Dimitrios Mallis, Vassilis Pitsikalis, Stavros Theodorakis, and Gil Chamiel. 2022. An Incremental Learning framework for Large-scale CTR Prediction. In *Proceedings of the 16th ACM Conference on Recommender Systems* (Seattle, WA, USA) *(RecSys '22)*. Association for Computing Machinery, New York, NY, USA, 490–493. doi:10.1145/3523227.3547390

[7] Diederik P Kingma. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[8] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences* 114, 13 (2017), 3521–3526.

[9] Hyeyoung Ko, Suyeon Lee, Yoonseo Park, and Anna Choi. 2022. A Survey of Recommendation Systems: Recommendation Models, Techniques, and Application Fields. *Electronics* 11, 1 (2022). doi:10.3390/electronics11010141

[10] Solomon Kullback and Richard A Leibler. 1951. On information and sufficiency. *The annals of mathematical statistics* 22, 1 (1951), 79–86.

[11] Congcong Liu, Fei Teng, Xiwei Zhao, Zhangang Lin, Jinghe Hu, and Jingping Shao. 2023. Always Strengthen Your Strengths: A Drift-Aware Incremental Learning Framework for CTR Prediction. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Taipei, Taiwan) *(SIGIR '23)*. Association for Computing Machinery, New York, NY, USA, 1806–1810. doi:10.1145/3539618.3591948

[12] Deepjyoti Roy and Mala Dutta. 2022. A systematic review and research perspective on recommender systems. *Journal of Big Data* 9, 1 (2022), 59.

[13] George Stoica, Daniel Bolya, Jakob Brandt Bjorner, Pratik Ramesh, Taylor Hearn, and Judy Hoffman. 2024. ZipIt! Merging Models from Different Tasks without Training. In *The Twelfth International Conference on Learning Representations*.

[14] Ruoxi Wang, Bin Fu, Gang Fu, and Mingliang Wang. 2017. Deep & Cross Network for Ad Click Predictions. In *Proceedings of the ADKDD'17* (Halifax, NS, Canada) *(ADKDD'17)*. Association for Computing Machinery, New York, NY, USA, Article 12, 7 pages. doi:10.1145/3124749.3124754

[15] Yichao Wang, Huifeng Guo, Ruiming Tang, Zhirong Liu, and Xiuqiang He. 2020. A practical incremental method to train deep ctr models. *arXiv preprint arXiv:2009.02147* (2020).

[16] Penghui Wei, Weimin Zhang, Ruijie Hou, Jinquan Liu, Shaoguo Liu, Liang Wang, and Bo Zheng. 2022. Posterior Probability Matters: Doubly-Adaptive Calibration for Neural Predictions in Online Advertising. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Madrid, Spain) *(SIGIR '22)*. Association for Computing Machinery, New York, NY, USA, 2645–2649. doi:10.1145/3477495.3531911

[17] Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. 2022. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International conference on machine learning*. PMLR, 23965–23998.

[18] Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. 2024. TIES-MERGING: resolving interference when merging models. In *Proceedings of the 37th International Conference on Neural Information Processing Systems* (New Orleans, LA, USA) *(NIPS '23)*. Curran Associates Inc., Red Hook, NY, USA, Article 310, 23 pages.

[19] Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. 2024. Extend model merging from fine-tuned to pre-trained large language models via weight disentanglement. *arXiv preprint arXiv:2408.03092* (2024).

[20] Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. 2024. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *Forty-first International Conference on Machine Learning*.

[21] Yuting Zhang, Yiqing Wu, Ran Le, Yongchun Zhu, Fuzhen Zhuang, Ruidong Han, Xiang Li, Wei Lin, Zhulin An, and Yongjun Xu. 2023. Modeling Dual Period-Varying Preferences for Takeaway Recommendation. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Long Beach, CA, USA) *(KDD '23)*. Association for Computing Machinery, New York, NY, USA, 5628–5638. doi:10.1145/3580305.3599866

[22] Yongchun Zhu, Jingwu Chen, Ling Chen, Yitan Li, Feng Zhang, and Zuotao Liu. 2024. Interest Clock: Time Perception in Real-Time Streaming Recommendation System. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Washington DC, USA) *(SIGIR '24)*. Association for Computing Machinery, New York, NY, USA, 2915–2919. doi:10.1145/3626772.3661369

[23] Yongchun Zhu, Guanyu Jiang, Jingwu Chen, Feng Zhang, Xiao Yang, and Zuotao Liu. 2025. Long-Term Interest Clock: Fine-Grained Time Perception in Streaming Recommendation System. *arXiv preprint arXiv:2501.15817* (2025).