

Deep Multiple Quantization Network on Long Behavior Sequence for Click-Through Rate Prediction

Zhuoxing Wei
Meituan
Beijing, China
weizhuoxing@meituan.com

Qi Liu*
University of Science and Technology
of China
Hefei, China
qiliu67@mail.ustc.edu.cn

Qingchen Xie
Meituan
Beijing, China
xieqingchen@meituan.com

ABSTRACT

In Click-Through Rate (CTR) prediction, the long behavior sequence, comprising the user's long period of historical interactions with items has a vital influence on assessing the user's interest in the candidate item. Existing approaches strike efficiency and effectiveness through a two-stage paradigm: first retrieving hundreds of candidate-related items and then extracting interest intensity vector through target attention. However, we argue that the discrepancy in target attention's relevance distribution between the retrieved items and the full long behavior sequence inevitably leads to a performance decline. To alleviate the discrepancy, we propose the Deep Multiple Quantization Network (DMQN) to process long behavior sequence end-to-end through compressing the long behavior sequence. Firstly, the entire spectrum of long behavior sequence will be quantized into multiple codeword sequences based on multiple independent codebooks. Hierarchical Sequential Transduction Unit is incorporated to facilitate the interaction of reduced codeword sequences. Then, attention between the candidate and multiple codeword sequences will output the interest vector. To enable online serving, intermediate representations of the codeword sequences are cached, significantly reducing latency. Our extensive experiments on both industrial and public datasets confirm the effectiveness and efficiency of DMQN. The A/B test in our advertising system shows that DMQN improves CTR by 3.5% and RPM by 2.0%.

CCS CONCEPTS

• Information systems → Recommender systems.

KEYWORDS

Click-Through Rate Prediction, Long Behavior Sequence Modeling

ACM Reference Format:

Zhuoxing Wei, Qi Liu, and Qingchen Xie. 2025. Deep Multiple Quantization Network on Long Behavior Sequence for Click-Through Rate Prediction. 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/10.1145/3726302.3730177>

*Corresponding author

1 INTRODUCTION

Click-through rate (CTR) prediction is a key stage in the industrial recommendation system (RS). Items with a higher CTR ranking will be displayed to the user. Thus, the accuracy of CTR prediction has a great influence on the evolution of RS. Existing researches have demonstrated that extracting a more accurate interest intensity vector towards the candidate from the long behavior sequence that contains long-term interactions can significantly boost the performance of CTR prediction. Current approaches [12] typically employ a two-stage paradigm to balance efficiency and effectiveness. Approximately one hundred candidate-related items are retrieved from the long behavior sequence first. Then, target attention is performed between the candidate and retrieved items to obtain an interest vector.

The primary focus of the research lies in the first stage. Researchers propose various solutions to improve retrieved items' accuracy with lightweight modules. SIM [13] applies the item's category as a relatedness metric to retrieve items. ETA [4] takes hamming distance computed by efficient locality-sensitive hashing to assess and retrieve items. SDIM [1] uses multi-round hash collision to increase the accuracy of proxy hamming metric. TWIN [3] solves the distribution discrepancy by make two stages share an efficient target attention network. Following research change to retrieve relevant clusters rather than original items, preserving more information about users' long-term preferences. TWIN V2 [14] employs a hierarchical clustering method to group items with similar characteristics into a cluster. DGIN [11] groups the long behavior sequence using the defined relevance key (like item_id) to enhance efficiency. The relevance distribution gap between the full long behavior sequence and the retrieved items/clusters paradigm will inevitably lead to biased and incomplete interest estimation.

To address retrieval-induced relevance distribution discrepancy, we propose the Deep Multiple Quantization Network (DMQN) for end-to-end long sequence processing. DMQN employs multiple learnable codebooks with codewords as cluster keys, capturing multi-aspect item features while minimizing information loss. This quantization transforms long behavior sequences into compact codeword representations. HSTU [16] facilitates interactions among codeword sequences, enhancing the model's understanding of user interest structures and yielding precise preference representations. Target attention between codewords and candidates then extracts long-term interests. This end-to-end approach alleviates relevance distribution discrepancies while maintaining online efficiency. Crucially, as quantization and interaction are candidate-agnostic, intermediate representations can be precomputed and cached, enabling low-latency inference.



Overall, we make the following contributions:

- We propose the Deep Multiple Quantization Network (DMQN) to quantize the long behavior sequence into approximately a hundred learnable codewords.
- The HSTU is incorporated to facilitate the interaction of each short codeword sequence. It enables a more profound exploration of how different interests interact and influence each other.
- To evaluate the effectiveness of DMQN, we conduct extensive experiments on both industrial and public datasets confirm the effectiveness and efficiency of DMQN. The A/B test in our advertising system shows that DMQN improves CTR by 3.5% and Revenue per Mile (RPM) by 2.0%.

2 METHODS

2.1 Preliminaries

CTR prediction focuses on estimating the likelihood that a user will click on a candidate item within a given context in the ranking stage. Each instance in this task can be represented as (\mathbf{x}, y) where $\mathbf{x} = [\mathbf{x}^u, \mathbf{x}^s, \mathbf{x}^t, \mathbf{x}^c]$, $y \in \{0, 1\}$ indicates click or not. $\mathbf{x}^u, \mathbf{x}^s, \mathbf{x}^t$ and \mathbf{x}^c represent the features' set of user, user behavior sequence, candidate item, and context respectively. Given training dataset $D = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)\}$, we need to learn a model f to predict the CTR, which can be formulated as the following Eq. (1):

$$\hat{y}_i = f(\mathbf{x}). \quad (1)$$

where \hat{y}_i is the estimated probability and f is the CTR model. CTR model is usually trained as a binary classification problem by minimizing the negative log-likelihood loss on the training dataset:

$$L_{ctr}(D) = -\frac{1}{N} \sum_{i=1}^N y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i). \quad (2)$$

where N is the size of the training dataset. For conciseness, we omit the subscript i in the following description when no confusion. As shown in Figure 1, DMQN is composed of the Multi-Cluster Quantization Module (MCQM), the Interest Clusters Interaction Module (ICIM), the Cluster-aware Target Attention Module (CTAM), and Multi-Layer Perception (MLP).

2.2 Multi-Cluster Quantization Module

Since DMQN aims at long behavior sequence modeling, we provide detail about \mathbf{x}^s . The long behavior sequence consists of the user's various interactions (e.g. view, click, add-to-cart, browse-dishes, etc) with items in chronological order for a long period. There \mathbf{x}^s can be represented as $\mathbf{x}^s = [\mathbf{x}_1^s, \mathbf{x}_2^s, \dots, \mathbf{x}_L^s] \in R^{L \times D}$, where L is the length of the long behavior sequence, D is embedding dimension. The goal of this module is to transform the entire user's long behavior sequence into sequences of approximately a hundred learnable interest clusters aka codewords by executing multiple times quantization.

2.2.1 Multi-cluster Codebook. Our quantization method is motivated by the idea of using codebooks to compress the embedding matrix [5, 7, 9, 10, 15]. We encode items with a set of C cluster codebooks to represent global cluster interests, each codebook contains W rows codewords, where each row is a D -dimensional vector that

serves as a cluster interest basis of the latent space. The multi-cluster codebooks can be encoded as:

$$\mathbf{C} = [\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_N] \quad (3)$$

where $\mathbf{c}_j \in R^{W \times D}$ is the j -th cluster codebook and N is the number of cluster codebooks. A linear projection is employed to map the representations of the behavior sequence into the same D -dimensional space as the multi-cluster codebook. This can be represented as:

$$\mathbf{h}_{c_j}^s = \text{concat}(\mathbf{e}_1^s, \mathbf{e}_2^s, \dots, \mathbf{e}_L^s) \mathbf{W}_2^{c_j} \quad (4)$$

where $\mathbf{h}_{c_j}^s \in R^{L \times D}$ is the transformed representation under the j -th cluster codebook, respectively. $\mathbf{W}_2^{c_j} \in R^{D \times D}$ is the corresponding weight matrix. Hereafter, the \mathbf{c}_j is omitted for the sake of brevity.

2.2.2 Multi-Cluster Quantization. It is reasonable that multiple times quantization is performed using dot product scores instead of Euclidean distance for computational and memory efficiency. The quantization of the user's behaviors can be encoded as follows:

$$\text{score}^s = \text{concat}(\mathbf{h}_1^s, \mathbf{h}_2^s, \dots, \mathbf{h}_L^s) \mathbf{c}^T \quad (5)$$

Where $\text{score}^s \in R^{L \times W}$ are the dot-product score of the user's behaviors at all rows of the cluster codebook.

The quantized scores are typically processed using a softmax function after acquiring the scores or distances. The Gumbel-Softmax technique [8] is then applied to introduce Gumbel noise into the scores, enabling the probabilistic selection of cluster indices rather than argmax always choosing the highest score. This approach promotes a more uniform assignment of cluster indices. The final probability distribution can be encoded as:

$$\text{prob}_k^{s_i} = \frac{\exp((\log(\text{score}_k^{s_i}) + g_k)/\tau)}{\sum_{p=1}^W \exp((\log(\text{score}_p^{s_i}) + g_p)/\tau)} \quad (6)$$

Where $\text{prob}_k^{s_i}$ are the probability of the j -th user's behavior at the k -th row of the cluster codebook. g_1, \dots, g_W are i.i.d samples drawn from Gumbel(0, 1). As the softmax temperature τ approaches 0, samples from the Gumbel-Softmax distribution become one-hot and the Gumbel-Softmax distribution converges to the categorical distribution of cluster interest.

The goal of MCQM is to obtain the cluster indices of the user's long behaviors. Following the Gumbel-Softmax trick, the argmax function is applied. This can be represented as:

$$\mathbf{z}_i^s = \arg \max_k (\text{prob}_k^{s_i}) \quad (7)$$

$$\mathbf{z}^s = \text{concat}(\mathbf{z}_1^s, \mathbf{z}_2^s, \dots, \mathbf{z}_L^s) \quad (8)$$

Where $\mathbf{z}^s \in \{1, \dots, W\}^L$ is the cluster indices of the user's behaviors.

We perform pooling on the embeddings of the behavior sequence corresponding to the identical indices among them. Through this operation, interest cluster representations are generated. Which can be encoded as:

$$\mathbf{r}_k = \text{avgpool}(\mathbf{e}_i^s | \mathbf{z}_i^s = k, i \in \{1, \dots, L\}), k \in \{1, \dots, W\} \quad (9)$$

$$\mathbf{R} = \text{concat}(\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_W) \quad (10)$$

Where \mathbf{r}_k represents the k -th interest representation within the user's behavior sequence. Since a vast quantity of behaviors can be clustered and represented by a relatively small number of interest

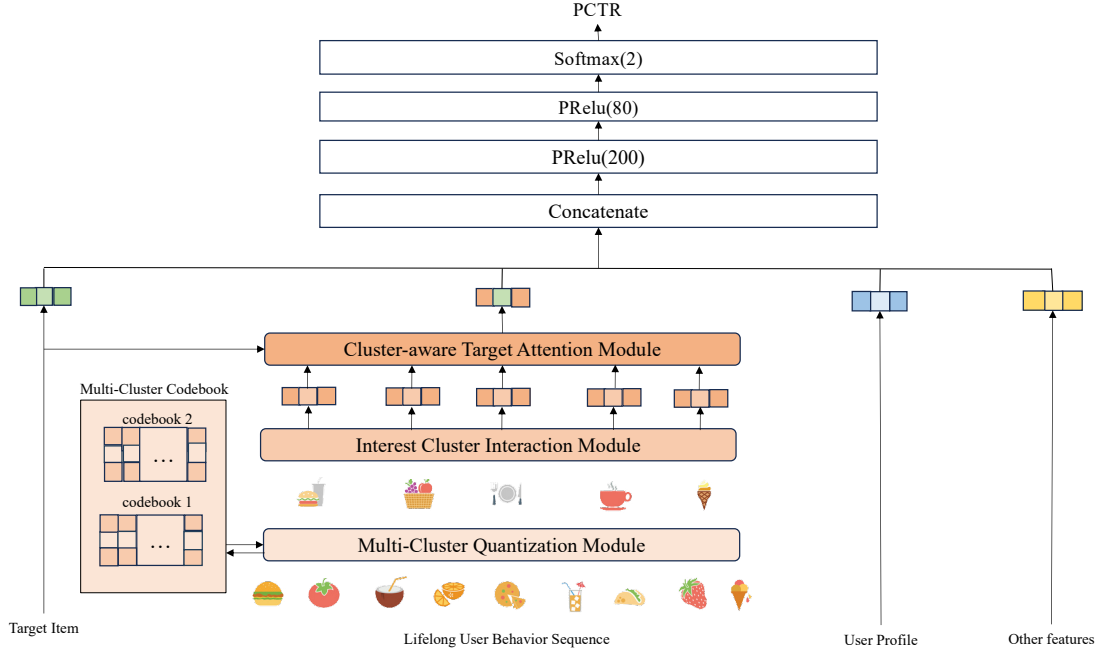


Figure 1: The overall framework of Deep Multiple Quantization Network (DMQN). DMQN consists of Multi-Cluster Quantization Module (MCQM), Interest Cluster Interaction Module(ICIM), and Cluster-aware Target Attention Module (CTAM).

clusters, it subsequently becomes feasible to conduct a complex attention mechanism.

2.3 Interest Clusters Interaction Module

In ICIM, the HSTU is incorporated to facilitate the interaction among the user’s interest clusters. It enables a more profound exploration of how different interest groups interact and influence each other. This interaction mechanism enriches the model’s understanding of the user’s interest structure, allowing for a more accurate representation of their preferences based on historical behavior patterns. Which can be represented as:

$$U(R), V(R), Q(R), K(R) = \text{Split}(\varphi_1(f_1(R))) \quad (11)$$

$$A(R)V(R) = \varphi_2(Q(R)K(R) + \text{bias}^p)V(R) \quad (12)$$

$$Y(R) = f_2(\text{Norm}(A(R)V(R)) \odot U(R)) \quad (13)$$

where $f_i(X)$ denotes an MLP; we use one linear layer, $f_i(R) = W_i(R) + b_i$ for f_1 and f_2 to reduce compute complexity and further batches computations for queries $Q(R)$, keys $K(R)$, values $V(R)$, and gating weights $U(R)$ with a fused kernel; φ_1 and φ_2 denote nonlinearity, for both of which we use SiLU; Norm is layer norm; and bias^p denotes relative cluster bias.

2.4 Cluster-aware Target Attention Module

To extract the user’s long-term interest representation, we conduct target attention between the interest cluster and the candidate item. This step is crucial as it bridges the gap between the user’s existing interests, as encapsulated by the interest clusters, and the potential items that might be of interest to them.

2.5 Caching The Intermediate Representation

Since the output of ICIM is irrelevant to the candidate item, we can precompute the representation of each user’s long behavior sequence and cache them in a key-value database such as Redis [2] after finishing training. When serving online, we can get the output from the key-value database based on user_id rather than computing online which is computationally expensive. Thus, we can launch DMQN successfully to meet the strict latency requirement.

2.6 Complexity

For the sake of brevity, the Multi-Cluster Quantization, which is similar to Multi-Head Attention, is omitted. The main sources of complexity are the Multi-Cluster Quantization Module and the Interest Clusters Interaction Module. The time complexity of the former is $O(L \cdot W \cdot D)$, and that of the latter is $O(N \cdot W^2 \cdot D + N \cdot W \cdot D^2)$. Therefore, the overall complexity of our method is $O(L \cdot W \cdot D + N \cdot W^2 \cdot D + N \cdot W \cdot D^2)$. Notably, it holds that $W \ll L$, which indicates that the length of the user’s behavior sequence far exceeds the number of interest clusters. Compared with the standard method HSTU with a complexity of $O(N \cdot L^2 \cdot D + N \cdot L \cdot D^2)$, our method has a relatively lower complexity.

3 EXPERIMENT SETUP

3.1 Datasets

Experiments are performed using both industrial and publicly available datasets. The statistical characteristics of each dataset are presented in Table 1. **Industry** utilized is the CTR dataset sourced from our online LBS platform. **Taobao** [19] is a prominent resource for

CTR prediction studies, comprising user interactions from Taobao’s industrial recommendation system.

Table 1: Statistics of datasets.

Datasets	#Users	#Items	#Fields	#Instances
Taobao	988K	4M	7	0.1B
Industry	400M	5M	317	6.6B

4 EXPERIMENT RESULTS

4.1 Overall Performance

Table 2 shows the results of all methods. DMQN obtains the best performance in both the Industry and Taobao datasets, which shows the effectiveness of DMQN. There are some insightful findings from the results.

The proposed DMQN reaches the best performance on both datasets. Compared with existing user behavior sequence modeling methods, DMQN compresses the extensive range of long user behaviors into approximately a hundred interest clusters. By incorporating the HSTU, it effectively promotes the interaction among these interest clusters. The HSTU’s self-attention mechanisms enable a detailed exploration of the relationships and interdependencies between different interest groups, thereby enhancing the model’s understanding of the user’s interest architecture.

DSIN’s better performance than DIN vividly illustrates the crucial role of interaction in extracting users’ session interests. As data complexity rises, DIN has limits, while DSIN’s enhanced interaction dissects session interests more precisely. This underlines the need for stronger interaction. The HSTU is thus introduced. It aims to supercharge interaction via hierarchical and self-attention architectures, unearthing deeper insights and more accurate interest representations, advancing user behavior modeling.

The performance boost from the two-step long behavior sequence modeling process highlights its significance. SIM, SDIM, TWIN, and TWIN V2 outperform DIN, DIEN, and DSIN. Short

Table 2: Performance of all methods on both datasets. The best result is in boldface and the second best is underlined. * indicates that the superiority to the best baseline.

	Industry AUC	Taobao AUC
DIN Small [18]	0.7011	0.7043
DSIN [6]	0.7029	0.7057
DIEN [17]	0.7035	0.7094
SIM [13]	0.7043	0.7419
TWIN [3]	0.7051	0.7523
SDIM [1]	0.7073	0.7347
DIN Middle [18]	0.7077	0.7573
TWIN V2 [14]	0.7078	0.7572
DIN Full [18]	<u>0.7087</u>	<u>0.7684</u>
DMQN	0.7104*	0.7724*

Table 3: Results of Integrating ICIM Successively.

	Industry AUC
TWIN V2	0.7078
DIN Full	0.7087
DMQN-simple	0.7089
+ICIM(DMQN)	0.7103

sequences offer a limited view, while long ones comprehensively mirror users’ complex, evolving interests and behaviors. Leveraging the full long sequence, as we plan, is crucial for deeper user understanding and application optimization.

4.2 Ablation Study

In this section, we investigate the effect of ICIM in DMQN and display the result in Table 3. The best baseline TWIN V2 and DIN Full are for comparison. In the ablation experiment where ICIM is removed from DMQN, DMQN-simple operates by compressing the entirety of the behavior sequence into roughly a hundred interest clusters and subsequently directly executing target attention. By contrast, the DMQN integrated with ICIM undertakes intricate interactions among the interest clusters before the stage of target attention. More precisely, we sequentially introduce the HSTU into the DMQN framework. The HSTU plays a crucial role in facilitating the interaction and information exchange among the interest clusters, thereby potentially augmenting the model’s capacity to capture and understand the complex relationships and patterns within the user’s behavior data, which could ultimately lead to improved performance and more accurate user interest representation.

4.3 A/B Test on Performance and Cost

We conducted an A/B test in the online LBS advertising system from 2024-05 to 2024-06 to measure the benefits of DMQN compared with the online baseline SIM Hard. DMQN was allocated **10%** experiment serving traffic while SIM Hard held **70%** main traffic. The online result showed the relative promotion of CTR and Revenue Per Mille (RPM) during one month’s testing. DMQN achieved **3.5%** and **2.0%** accumulated relative promotion on the CTR and RPM respectively during the A/B test period, proving its effectiveness in the online LBS advertising system. The parameter storage costs of SIM and DMQN were **2.85 GB** and **4.00 GB** respectively. SIM had an average inference latency of **4.6 ms** and DMQN had **7.4 ms**. The resource cost bought by DMQN was negligible.

5 CONCLUSION

In this paper, we propose the DMQN for full long user behavior sequence modeling in the CTR prediction task. DMQN, consisting of a Multi-Cluster Quantization Module, an Interest Cluster Interaction Module and a Target Module, aims at extracting fine-grained comprehensive unbiased interest and psychological decision interest to achieve a deep understanding of the user’s preference. To the best of our knowledge, DMQN is the first to achieve efficient end-to-end full long user behavior sequence modeling.

REFERENCES

- [1] Yue Cao, Xiaojiang Zhou, Jiaqi Feng, Peihao Huang, Yao Xiao, Dayao Chen, and Sheng Chen. 2022. Sampling is all you need on modeling long-term user behaviors for CTR prediction. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 2974–2983.
- [2] J Carlson. 2013. *Redis in Action*. Manning.
- [3] Jianxin Chang, Chenbin Zhang, Zhiyi Fu, Xiaoxue Zang, Lin Guan, Jing Lu, Yiqun Hui, Dewei Leng, Yanan Niu, Yang Song, et al. 2023. TWIN: Two-stage Interest Network for Lifelong User Behavior Modeling in CTR Prediction at Kuaishou. *arXiv preprint arXiv:2302.02352* (2023).
- [4] Qiwei Chen, Yue Xu, Changhua Pei, Shanshan Lv, Tao Zhuang, and Junfeng Ge. 2022. Efficient Long Sequential User Data Modeling for Click-Through Rate Prediction. *arXiv preprint arXiv:2209.12212* (2022).
- [5] Ting Chen, Martin Renqiang Min, and Yizhou Sun. 2018. Learning k-way d-dimensional discrete codes for compact embedding representations. In *International Conference on Machine Learning*. PMLR, 854–863.
- [6] Yufei Feng, Fuyu Lv, Weichen Shen, Menghan Wang, Fei Sun, Yu Zhu, and Keping Yang. 2019. Deep session interest network for click-through rate prediction. *arXiv preprint arXiv:1905.06482* (2019).
- [7] Tiezheng Ge, Kaiming He, Qifa Ke, and Jian Sun. 2013. Optimized product quantization. *IEEE transactions on pattern analysis and machine intelligence* 36, 4 (2013), 744–755.
- [8] Eric Jang, Shixiang Gu, and Ben Poole. 2016. Categorical Reparameterization with Gumbel-Softmax. *arXiv preprint arXiv:1611.01144* (2016).
- [9] Herve Jegou, Matthijs Douze, and Cordelia Schmid. 2010. Product quantization for nearest neighbor search. *IEEE transactions on pattern analysis and machine intelligence* 33, 1 (2010), 117–128.
- [10] Defu Lian, Haoyu Wang, Zheng Liu, Jianxun Lian, Enhong Chen, and Xing Xie. 2020. Lightrec: A memory and search-efficient recommender system. In *Proceedings of The Web Conference 2020*. 695–705.
- [11] Qi Liu, Xuyang Hou, Haoran Jin, Zhe Wang, Defu Lian, Tan Qu, Jia Cheng, Jun Lei, et al. 2023. Deep Group Interest Modeling of Full Lifelong User Behaviors for CTR Prediction. *arXiv preprint arXiv:2311.10764* (2023).
- [12] Qi Pi, Weijie Bian, Guorui Zhou, Xiaoqiang Zhu, and Kun Gai. 2019. Practice on long sequential user behavior modeling for click-through rate prediction. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2671–2679.
- [13] Qi Pi, Guorui Zhou, Yujing Zhang, Zhe Wang, Lejian Ren, Ying Fan, Xiaoqiang Zhu, and Kun Gai. 2020. Search-based user interest modeling with lifelong sequential behavior data for click-through rate prediction. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2685–2692.
- [14] Zihua Si, Lin Guan, ZhongXiang Sun, Xiaoxue Zang, Jing Lu, Yiqun Hui, Xingchao Cao, Zeyu Yang, Yichen Zheng, Dewei Leng, et al. 2024. TWIN V2: Scaling Ultra-Long User Behavior Sequence Modeling for Enhanced CTR Prediction at Kuaishou. *arXiv preprint arXiv:2407.16357* (2024).
- [15] Yongji Wu, Defu Lian, Neil Zhenqiang Gong, Lu Yin, Mingyang Yin, Jingren Zhou, and Hongxia Yang. 2021. Linear-time self attention with codeword histogram for efficient recommendation. In *Proceedings of the Web Conference 2021*. 1262–1273.
- [16] Jiaqi Zhai, Lucy Liao, Xing Liu, Yueming Wang, Rui Li, Xuan Cao, Leon Gao, Zhaojie Gong, Fangda Gu, Michael He, Yinghai Lu, and Yu Shi. 2024. Actions Speak Louder than Words: Trillion-Parameter Sequential Transducers for Generative Recommendations. *arXiv:2402.17152 [cs.LG]* <https://arxiv.org/abs/2402.17152>
- [17] 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.
- [18] Guorui Zhou, Nan Mou, Yukuai Fan, Qiang Pi, Wu Bian, Xing Zhou, and Hui Yang. 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.
- [19] Han Zhu, Daqing Chang, Ziru Xu, Pengye Zhang, Xiang Li, Jie He, Han Li, Jian Xu, and Kun Gai. 2019. Joint optimization of tree-based index and deep model for recommender systems. *Advances in Neural Information Processing Systems* 32 (2019).