Deep Group Interest Network on Full Lifelong User Behaviors for CTR Prediction

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

Xuyang Hou
Meituan
Beijing, China
houxuyang@meituan.com

Haoran Jin
University of Science and Technology of China
Hefei, China
haoranjin@mail.ustc.edu.cn

Xiaolong Chen
University of Science and Technology of China
Hefei, China
chenxiaolong@mail.ustc.edu.cn

Jin Chen
University of Electronic Science and Technology of China
Chengdu, China
chenjin@std.uestc.edu.cn

Defu Lian
University of Science and Technology of China
Hefei, China
liandefu@ustc.edu.cn

Zhe Wang
Meituan
Beijing, China
wangzhe65@meituan.com

Jia Cheng
Meituan
Beijing, China
jia.cheng.sh@meituan.com

Jun Lei
Meituan
Beijing, China
leijun@meituan.com

ABSTRACT

Modeling user interest based on lifelong behavior sequences is key for improving Click-Through Rate (CTR) predictions. Current approaches typically use a two-step process to balance efficiency with effectiveness. Initially, they use an effective algorithm to identify relevant historical behaviors to candidates in the first phase, and then they focus on a shorter subsequence to ascertain user interest via target attention. However, this two-step approach, despite its effectiveness, unavoidably results in some loss of information. Moreover, limiting interest modeling to just click behaviors introduces bias, as other historical behaviors like purchases also affect the likelihood of a click. These issues prevent the CTR prediction model from achieving its full potential. In our study, we introduce the **D**eep **G**roup **I**nterest **N**etwork (**DGIN**), a comprehensive method that processes the entire spectrum of a user's lifelong behavior, including clicks, collections and purchases, in an end-to-end manner. We start by organizing the complete lifelong behavior sequences into groups based on a specific interest key, significantly reducing the sequence length from tens of thousands to hundreds. To overcome the potential information loss from this grouping, we make the following two designs. Firstly, we analyze behaviors within each group using both simple statistics and self-attention to capture group traits and then pinpoint user interests by applying target attention to these groups. Secondly, we refine the user's decision-making interest by employing the attention mechanism to identify the user's candidate-specific interests, based on behavior subsequences that share the same interest key. Our extensive experiments on both industrial and public datasets confirm the effectiveness and efficiency of DGIN. The A/B test in our LBS advertising system shows that DGIN improves CTR by 4.5% and Revenue per Mile by 2.0%.

CCS CONCEPTS

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

KEYWORDS

Click-Through Rate Prediction, Lifelong Behaviors Modeling

1 INTRODUCTION

In contemporary recommendation systems, predicting the Click-Through Rate (CTR) is crucial for determining the likelihood of a user engaging with suggested items. This prediction significantly influences the algorithm that ranks and presents items to users. Over recent years, there has been a surge in methods aimed at improving CTR prediction accuracy, with a significant focus on extracting patterns of interest from users' past activities. Studies have shown that analyzing sequences of user behavior over their entire history is more advantageous than just considering recent actions, as it offers a more complete picture of their preferences. However, the need for low latency in online platforms presents a considerable obstacle to efficiently processing these extensive behavior sequences. Therefore, there's a pressing need to explore the full potential of these lifelong behavior sequences for improved recommendation accuracy.
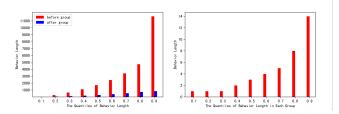
Figure 1: The distribution of behaviors' length affected by grouping on *item_id*. We rank samples on behaviors' length in ascending order. The left half shows the corresponding behavior's length at different quantiles. The right half displays the distribution of the behavior quantity of each group.

Existing approaches to modeling lifelong behavior sequences typically adopt a two-stage methodology. Initially, a rapid and lightweight retrieval module sifts through hundreds of historical behaviors pertaining to the candidate item from a vast pool of lifelong behaviors. Subsequently, target attention is employed to infer the user's interest based on the retrieved subsequence. The primary focus of research has predominantly revolved around the initial stage. For instance, SIM [20] retrieves behaviors belonging to the same category or cluster as the candidate item. UBR4CTR [22] employs BM25 [24] for relevance scoring and utilizes an inverted index for retrieving relevant behaviors. ETA [4] employs locality-sensitive hashing (LSH) [8] for efficient user behavior retrieval. SDIM [2] selects behaviors sharing the same hash signature as the candidate item through multi-round hash collision. TWIN [3] addresses the inconsistency in relevance scoring between the two stages by employing shared efficient target attention. This two-stage approach embodies a compromise between effectiveness and efficiency.

While effective, these algorithms still face two significant limitations: biased and incomplete interest estimation stemming from retrieval-based modeling and reliance solely on historical click behaviors, respectively. Biased interest estimation arises from the retrieval process in the first stage, which retains only the most relevant behaviors while discarding at least 95% of historical behaviors. Incomplete interest estimation results from the only use of historical click behaviors to construct the lifelong behavior sequence, driven by concerns about consistency with the Click-Through Rate (CTR) task. However, there exists a wide array of user-item interactions. For example, in an online location-based services (LBS) platform, the behaviors of interaction range from click, browse-dishes, and view-comments to add-to-cart and purchase. Relying solely on lifelong click behavior sequences cannot adequately depict a user's interests since the significance of interest varies across different behavior types [11]. Hence, utilizing the entirety of lifelong user behaviors presents an opportunity to capture fine-grained interest patterns comprehensively.

To tackle the issues of biased and incomplete interest estimation, we introduce the Deep Group Interest Network (DGIN) for comprehensive modeling of lifelong user behaviors. DGIN efficiently extracts user interests from the entirety of lifelong behavior sequences in an end-to-end manner. It is well-known that interests play a significant role in shaping behavior, and behavior often provides clues about underlying interests. Therefore, the rate of growth

in the number of behaviors over time exceeds the ratio of the number of interests by orders of magnitude. Consequently, we initially organize the full lifelong behaviors into interest groups based on a designated interest key. This key serves as the focal point of interest and can either be a predefined concept (e.g., category_id, item_id) or learned from the data. The grouping operator transforms the lifelong behavior sequence into interest groups, significantly reducing the behavior length. Illustrated in Figure 1, this grouping operation diminishes the magnitude of behavior length from $O(10^4)$ to $O(10^2)$. In our methodology, we utilize the item_id as the grouping key, considering the recurrent consumption habits observed in our online Location-Based Services (LBS) platform. Additionally, our experiments demonstrate performance improvements when grouping behavior sequences by category_id. Given that category information is prevalent across various recommendation scenarios, our proposed approach exhibits universality.

To mitigate the loss of information resulting from behavior grouping, we make the following two designs. First, we conduct an analysis of the behaviors within each group by utilizing both simple statistics and self-attention mechanisms to identify the distinctive traits of each group, and then we apply target attention to these groups to accurately identify user interests. Specifically, we calculate statistics, such as the frequency of different behaviors within a group, to gauge the user's level of interest. Simultaneously, we employ self-attention on each group of behaviors to capture its unique characteristics, since the details of the behaviors (like the type of behavior and the time it occurs) reveal how a user's interests evolve over time. For instance, a user might only buy coffee during weekdays, not weekends. This approach aims to retain as much unique behavior information within each group as possible. The subsequent use of target attention is to precisely determine the user's interests from these interest groups. This comprehensive approach reduces the information loss. Second, the complete history of a user's behavior reveals patterns of how their interests evolve towards certain items, which can help predict how likely they are to click on the item. To do this, we use self-attention to deduce interests specific to a candidate item from subsequences of behaviors that align with the candidate's interest key. We then refine this with target attention, allowing us to understand the user's decision-making process towards the candidate, particularly in Location-Based Services (LBS) platforms where repeated interactions are common. Overall, we make the following contributions:

- We are the first to achieve efficient end-to-end interest extraction by taking all behaviors into calculation in lifelong/long behavior sequence modeling. We reveal the necessity of introducing multiple types of behaviors into the lifelong sequence modeling.
- We propose a Deep Group Interest Network for capturing the user's long-term interest, where we organize lifelong behavior sequences into interest groups, remarkably reducing computation overload. We also sample a subsequence to capture the user's decision pattern towards the candidate.
- To evaluate the effectiveness of DGIN, we conduct offline experiments on both industrial and public datasets. The results demonstrate a remarkable improvement achieved by DGIN. The A/B test in our LBS advertising system shows DGIN improves CTR by 4.5% and Revenue per Mile by 2.0%.

2 RELATED WORK

2.1 Click-Through Rate Prediction

CTR prediction has been a long-standing research hotspot in RS. Early CTR methods [16, 23, 27] mainly focus on the low-order feature interactions. Recently, methods based on deep learning have achieved amazing progress in CTR prediction. Wide&Deep [6] utilizes the linear model to memorization of feature interaction and takes the deep neural network to achieve generalization. DeepFM [13] replaces the linear model with FM to emphasize the second-order feature interactions. DCN [26] applies a cross-vector network to learn informative feature interactions automatically.

User behavior sequence modeling [5, 9, 12, 29, 33, 35, 36] also attracts lots of attention and develops fast. It focuses on extracting the user's interest from historical behaviors to improve CTR prediction accuracy. Limited by the online latency, most existing methods design algorithms on the truncated short user behavior sequence, which only contains the user's instant interest. DIN [36] first performs attention between candidate item and behavior sequence to extract interest by emphasizing candidate-relevant behaviors and suppressing candidate-irrelevant ones. DIEN [35] further takes a two-layer GRU [7] to model the temporal shifting and mine interest in the interest level. DSIN [9] divides behavior sequence into sessions and uses self-attention together with Bi-LSTM [10] to obtain session representation. Then it extracts interest from sessions' representation through target attention. However, when applying DSIN to lifelong behavior sequence modeling, the setting of the session's time interval is troublesome. When the time interval is small, there are lots of sessions leading to inefficiency. When the time interval is large, there will many heterogeneous behaviors that hold different item_ids and different category_ids within each session and the session aggregation will cause severe information loss. NINN [33] just partition the behavior sequence into different categories and capture the interactions among them. However, it does not supplement any statistical or dynamic information, which will give rise to performance degradation. Meanwhile, some works introduce multiple types of behavior sequences [12, 14, 18, 28, 34, 37] into CTR modeling to obtain comprehensive fine-grained interest. ATRANK [34] utilizes the Tansformer [25] encoder-decoder network to acquire interest from the mixed behavior sequence which consists of various types of behaviors in chronological order. While DMT [12] uses different behavior modeling networks to process different types of behavior sequences. However, the mere use of the short behavior sequence can not obtain important long-term interest patterns and restrict the performance improvement of CTR prediction.

2.2 Long User Behavior Sequence Modeling

Due to the effectiveness of user behavior sequence modeling, long behavior sequences modeling [2, 4, 19–21, 30, 32] have been explored. MIMN [19] applied the memory network and GRU to induce interest stored at the user interest center. However, MIMN barely processes sequences longer than 10^3, and the induction process without the candidate item causes lots of information loss. After MIMN, the two-stage solution becomes the mainstream. The first stage retrieves hundreds of candidate-relevant behaviors from the

long behavior sequence in an efficient way and then performs target attention with the retrieved behaviors to extract interest. SIM Hard [20] takes behaviors that have the same category as the candidate item. SIM Soft [20] selects top-ranked behaviors according to the inner product between the candidate item and historical behaviors with pre-trained item embeddings. UBR4CTR [22] chooses the BM25 as the retrieve metric. Recently, ETA [4] and SDIM [2] have tried to do retrieving and extracting in an end-to-end manner. ETA [4] uses LSH to perform item embedding binarization and takes the Hamming distance as the metric to filter behaviors. SDIM [2] selects behaviors that hold the same hash signature as the candidate item through multi-round hash collision and the embedding of selected behaviors will be aggregated directly. TWIN [3] solves the relevance inconsistency problem by sharing an efficient target attention network between two stages. Although end-to-end, the interest is still extracted from the retrieved subsequence and can't escape the problem of biased interest. All those methods only retrieve candidate-relevant subsequence from the click behaviors, which leads to biased and incomplete interest modeling. The above problems motivate us to explore an efficient end-to-end method of taking all information for full lifelong user behavior sequence modeling. Unlike existing ETA, SDIM, and TWIN, which adhere to a two-stage end-to-end framework where only the candidate-relevant behaviors contribute to the gradient, DGIN takes a different path. DGIN utilizes a grouping strategy, enabling all behaviors in the lifelong behavior sequence to actively participate in extracting interest. This design ensures that each gradient update is associated with all behaviors during the backward process, facilitating end-to-end training with full information.

3 METHODS

3.1 Preliminaries

CTR prediction aims at estimating the probability of the user clicking a candidate item under a specific context in the ranking stage. The instance can be represented by $(\mathbf{x}, y)$, where $\mathbf{x} = [\mathbf{x}^u, \mathbf{x}^s, \mathbf{x}^i, \mathbf{x}^c]$, $y \in \{0, 1\}$ indicates click or not. $\mathbf{x}^u$, $\mathbf{x}^s$, $\mathbf{x}^i$, and $\mathbf{x}^c$ represent the features' set of user, user behavior sequence, candidate item, and context respectively. Given training dataset $D = \{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_N, y_N)\}$, we need to learn a model f to predict the CTR, which can be formulated as the following Eq. (1):

$$p_i = f(\mathbf{x}). \tag{1}$$

where p_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:

$$l(D) = -\frac{1}{N} \sum_{i=1}^{N} y_i \log(p_i) + (1 - y_i) \log(1 - p_i). \tag{2}$$

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 2, DGIN is composed of the Embedding layer, the Group Module (GM), the Target Module (TM), and Multi-Layer Perception (MLP). The original input $\mathbf{x}$ of the CTR model is sparse one-hot vectors. It will go through the embedding layer to get the low-dimensional dense vectors as representation. The GM takes the already grouped lifelong behavior sequence as input. It first
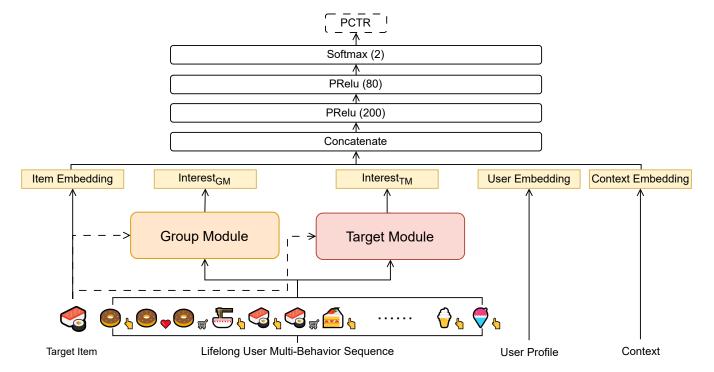
Figure 2: The overall framework of Deep Group Interest Network (DGIN). DGIN consists of Group Module (GM) and Target Module (TM), which capture long-term and decision interest from lifelong user multi-behavior sequence.

applies the self-attention mechanism to aggregate the unique characteristics of each behavior within the group and then performs target attention to extract user interest in an end-to-end manner. Meanwhile, the TM takes the subsequence whose behaviors hold the same interest key as the candidate item and then obtains the psychological decision interest towards the candidate through another attention network. The extracted long-term interest and psychological decision interest will serve as the input of the MLP.

3.2 Embedding Layer

Each field has its own embedding matrix $\mathbf{E} = [\mathbf{e}_1; \mathbf{e}_2; \cdots ; \mathbf{e}_K] \in \mathbb{R}^{K \times d}$, where K represents the cardinality of the field and d donates the embedding dimensional. The $\mathbf{e}_i$ severs as the embedding of the feature assigned index i in the field. We make all fields share the same embedding dimensional.

Since DGIN focuses on full lifelong user behavior sequence modeling, we provide processing detail about $\mathbf{x}^s$. The lifelong user behavior sequence consists of the user's various interactions (e.g. click, add-to-cart, browse-dishes, etc) with items in chronological order after registration. There $\mathbf{x}^s$ can be represented as $\mathbf{x}^s = [\mathbf{x}_1^s, \mathbf{x}_2^s, ..., \mathbf{x}_L^s]$, where L is the length of the lifelong behavior sequence. To fully describe each behavior, each $\mathbf{x}_i^s$ has lots of attributes, such as *item_id, category_id, price, timestamp, location, behavior_type* etc. The embedding layer transform each attribute into corresponding embeddings $\{\mathbf{e}_{i,item\_id}^s, ..., \mathbf{e}_{i,timestamp}^s, ..., \mathbf{e}_{i,behavior\_type}^s\}$. We concatenate them together to form the behavior representation $\mathbf{e}_i^s = [\mathbf{e}_{i,item\_id}^s, ..., \mathbf{e}_{i,timestamp}^s, ..., \mathbf{e}_{i,behavior\_type}^s]$.

3.3 Group Module

Figure 3 shows the details of the Group Module. As the goal is to extract the user's interest towards the candidate item from the lifelong behavior sequence, it is reasonable to cluster the behaviors in a coarse interest level and then mine interest from the clustered interests, which will be more computationally efficient. In the offline data processing stage, GM first groups the lifelong behavior sequence into interest groups based on interest key. Within each group, the behavior subsets are still stored in chronological order. However, grouping behaviors will damage the information integrality such as the occurrence times of different types of interaction, the spatio-temporal relationship among behaviors, etc. To make up for the lost information, we designed two types of features as the attributes of the interest groups, including the statistical and aggregated attributes. The statistical attributes are statistics of behaviors within the group from different perspectives and the aggregated attributes are obtained by applying self-attention to the unique attributes of original behaviors, such as timestamp. The common attributes (e.g. item_id, category_id) shared by behaviors within the group together with the supplementary attributes form the attributes of the interest groups. Due to the limited interest groups, we employ Multi-Head Target Attention (MHTA) to extract unbiased and comprehensive interest from the interest groups.

3.3.1 Statistical Attributes. Inspired by **Quantity Breeds Quality**, we calculate the various statistics about quantity within the group. More behaviors towards the item mean more preference. Within the group, we count the total number of behaviors, total behavior
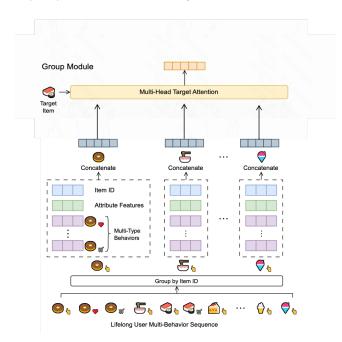
Figure 3: The detail of GM's architecture. We take item_id as the interest key for illustration.

types, and individual numbers of different types of behaviors. From the duration perspective, we calculate the average dwell time of all behaviors. And we will also obtain the average consumption amount of all purchase behaviors. The quantity, time, and money are three straightforward factors which reflect the user's interest. All the statistical attributes are processed during the offline data processing stage, and there is no extra inference cost. We represent the statistical attributes as Eq. (3).

$$\mathbf{attr}_s = [\mathbf{attr}_{counts}, \mathbf{attr}_{types}, ..., \mathbf{attr}_{avg\_price}] \tag{3}$$

3.3.2 Aggregated Attribute. Statistical attributes only reflect interest intensity, but can not tell the interest evolution. Within the group, the spatio-temporal attributes like *timestamp, location* of behavior can represent the user's interest evolution process on the coarse interest. Meanwhile, we can also observe the user's attitude about the item from the heterogeneity interactions. For example, the recent click behavior may strongly impact the current click, while clicks long ago have little influence. However, purchases long ago may have a greater impact on current clicks. Thus, we use the attributes' sequence: *timestamp, location, and behavior_type* to supplement the information on the interest dynamics.

Specifically, suppose there are maximum of B behaviors in each group as $\mathbf{b} =< \mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_B >$. We concatenate the embeddings of *timestamp, location, and behavior_type* as the behavior's representation $\mathbf{e}_{b_i} = [\mathbf{e}_{b_i,timestamp}, \mathbf{e}_{b_i,location}, \mathbf{e}_{b_i,behavior\_type}]$. The intra-group behavior sequence can be expressed as $\mathbf{e}_b = [\mathbf{e}_{b_1}, \mathbf{e}_{b_2}, ..., \mathbf{e}_{b_B}]$. Due to the ability to model behavior pairs from multiple perspectives, we apply Multi-Head Self Attention MHSA to capture the interest evolution. The MHSA can be expressed as follows:

$$MHSA(\mathbf{e}_b) = concat(head_1, ..., head_h)W^O, \tag{4}$$

$$head_i = Softmax(\frac{\mathbf{e}_b W_i^Q (\mathbf{e}_b W_i^K)^T}{\sqrt{d'}})\mathbf{e}_b W_i^V, \tag{5}$$

where h is the number of heads, $W_i^Q, W_i^K, W_i^V \in R^{3d \times d'}$, $W^O \in R^{3d \times 3d}$. $3d$ and d' are the dimension of the input and weight vectors while $d' = \frac{3d}{h}$. Then, mean pooling is taken to process the $MHSA(\mathbf{e_b})$ and acquire the aggregated attribute as Eq. (6).

$$\mathbf{attr}_a = mean\_pool(MHSA(\mathbf{e}_b)). \tag{6}$$

where $\mathbf{attr_a} \in R^{3d}$ is the aggregated attribute.

3.3.3 Attention On Interest Set. The grouping operation changes the full lifelong behavior sequence into limited coarse interest groups. Three types of attributes are utilized to describe the characteristics of each interest group. The first is the identity attributes $\mathbf{attr_i}$ including interest_key_id, category_id, etc. Secondly, statistical attributes are calculated to represent the interest intensity. Finally, aggregated attribute reflects the interest dynamic. After obtaining all attributes, we exploit the MHTA to perform interest activation to get the user's interest towards each candidate item. MHTA holds the network structure that takes the candidate item as query and the interest sets as key and value. In the ideal attention mechanism, the identity attributes (like category_id) of different behaviors holding the same interest key will participate in the calculation of attention score many times, which will give rise to redundant computation. GM avoids lots of redundant computation and achieves computation efficiency.

Specifically, suppose there are maximum of G interest groups in each full lifelong behavior sequence as $\mathbf{g} =< \mathbf{g}_1, \mathbf{g}_2, ..., \mathbf{g}_G >$. The representation of each interest group is $\mathbf{e}_{g_i} = [\mathbf{e_{attr_i}}, \mathbf{e_{attr_s}}, \mathbf{e_{attr_a}}]$. We can get the interest groups' representation $\mathbf{e_g} = [\mathbf{e}_{g_1}, \mathbf{e}_{g_2}, ..., \mathbf{e}_{g_G}]$. The fine-grained interest in the candidate item can be expressed as:

$$interest_{GM} = MHTA(\mathbf{e^i}, \mathbf{e_g}) \tag{7}$$

where $\mathbf{e^i}$ is the representation of the candidate item, $\mathbf{interest_{GM}}$ is the unbiased and comprehensive interest extracted from the full lifelong behavior sequence. To summarize, we believe that the computation efficiency results from reducing lots of redundant computations.

3.4 Target Module

Figure 4 shows the details of TM. TM focuses on capturing the user's historical evolution process on the candidate item, which is ignored by previous methods. Firstly, TM retrieves behaviors holding the same interest key as the candidate item from the lifelong behavior sequence. The behavior pattern contained in this subsequence is a strong signal indicating the user's habit of the candidate item. For example, the user may click the candidate item every week or do a lot of micro behaviors and purchase finally. The signal is flooded to some extent in GM because other items take away some attention. However, behaviors in the subsequence are similar because of belong to the same interest key but the subsequence is short. Thus, we first use the MHSA to strengthen the subtle differences of behaviors by capturing behaviors' mutual relatedness. And then MHTA is applied to extract the psychological decision interest from the refined and differentiated behavior representation. Specifically, this candidate aware subsequence can be expressed as $\mathbf{t} =< \mathbf{t}_1, \mathbf{t}_2, ..., \mathbf{t}_T >$. We
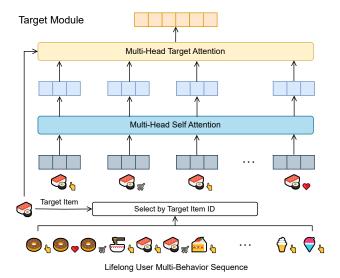
Figure 4: The detail of TM's architecture.

use the original $\mathbf{e}_i^s$ to represent each behavior, and the subsequences representation is $\mathbf{e_t} =< \mathbf{e_{t_1}}, \mathbf{e_{t_2}}, ..., \mathbf{e_{t_T}} >\in \mathbb{R}^{T \times md}$, where m is the number of attributes. The modeling process follows Eq. (8):

$$
\begin{aligned}
out_{MHSA}^{\mathbf{t}} &= LN(\mathbf{e_t} + MHSA(\mathbf{e_t})), \\
out_{enc}^{\mathbf{t}} &= LN(out_{MHSA}^{\mathbf{t}} + FFN(out_{MHSA}^{\mathbf{t}})), \\
out_{MHSA}^{\mathbf{t}} &= LN(\mathbf{e}^i + MHTA(\mathbf{e}^i, out_{enc}^{\mathbf{t}})), \\
interest_{TM} &= LN(out_{MHSA}^{\mathbf{t}} + FFN(out_{MHSA}^{\mathbf{t}})),
\end{aligned}
\tag{8}
$$

where $\mathbf{interest_{TM}}$ is the the psychological decision interest.

4 EXPERIMENT SETUP

4.1 Datasets

Experiments are conducted on both industrial and public datasets. The statics values of each dataset are shown in Table 1.

Industry is the CTR dataset collected from our online LBS platform. The last 30 days' logs are used for training and samples of the following day are for testing. To exploit abundant behavior information, we collect various historical behaviors of each user from the past 2 years. The maximum length of the full lifelong behavior sequence is 10,000. There are 6 types of behaviors in our RS including click, add-to-cart, add-to-favorite, browse-dishes, view-comments, and purchase.

Taobao [38] is a widely used dataset for CTR prediction research. It is composed of user behaviors from Taobao's industrial recommendation system. The dataset contains about 1 million users whose behaviors include clicking, adding-to-cart, and purchasing. We take the various behaviors of each user and sort them based on timestamps to construct the full lifelong behavior sequence. The maximum behavior sequence length is set to 500. Following MIMN [19], we use the former $T - 1$ behaviors to predict whether the user will click the $T - th$ item.

Table 1: Statistics of datasets.

Datasets	#Users	#Items	#Fields	#Instances
Taobao	988K	4M	7	0.1B
Industry	40M	417K	168	6.6B

4.2 Baselines

We choose baselines for a comparison from four perspectives. First, we choose methods focusing on short behavior sequence modeling, including **DIN** [36], **DIEN** [35], **DSIN** [9] which extracts the user's interest from short click behavior sequence. Second, we compare the proposed method with the modeling baselines over multiple types of short behavior sequences **DMT** [12], **DMBIN** [15], **TEM4CTR** [31]. **SIM** [20], **ETA** [4], **SDIM** [2], **NINN** [33] are all methods of mining user interest from lifelong click behavior sequence, which serve as the third type of baselines. Fourth, **TWIN** [3] concentrates on full lifelong behavior sequence modeling. We add **SIM-TM** which integrates SIM with our Target Module on category_id to show the effectiveness of MHSA in TM. As the hierarchical attention structure of **DSIN** is also suitable for long behavior sequence modeling efficiently, we let **DSIN** directly process the long behavior sequence.

4.3 Evaluation Metric

Two widely used metrics AUC [1], and LogLoss are chosen. The AUC (Area Under the ROC Curve) measures the ranking accuracy. A higher AUC indicates better performance. The LogLoss measures the accuracy of the estimated probability depending on the ground-truth label. Even a slight improvement is considered a significant boost for the industry recommendation task [13], as it leads to a significant increase in revenue.

4.4 Implementation Details

As we observe that the user accesses the same item repeatedly on both the industrial and public datasets, we choose the item_id as the interest key. We will show that category_id is also a reasonable choice of interest key in the ablation study. We choose the existing concepts like item_id and category_id as the interest key because such choice is engineering friendly as SIM [20]. The grouping operation can be done efficiently during offline data processing. For a fair comparison, we keep the identical network structures except for the behavior sequence modeling module. For SIM, ETA, SDIM, and TWIN, we retrieve the top 50 most candidate-relevant behaviors into the second interest extraction stage. We implement DGIN with Tensorflow. For the industry dataset, the embedding size is 16 and the learning rate is $5e - 4$. We train the model using eight $80G$ $A100$ GPUs with the batch size 1500 of a single card. For the Taobao dataset, we set the embedding size to be 18, the learning rate to be $1e - 3$, and use a single 80 $A100$ for training with batch size 1024. We use Adam [17] as the optimizer for both datasets. We run all experiments five times and report the average result.

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 is statistically significant at 0.01 level.

	Industry		Taobao	
	AUC	Logloss	AUC	Logloss
DIN	0.6910	0.0606	0.6622	0.0626
DIEN	0.6916	0.0605	0.6806	0.0577
DMT	0.6946	0.0603	0.6955	0.0520
DMBIN	0.6953	0.0603	0.7252	0.0507
TEM4CTR	0.6959	0.0602	0.7311	0.0495
SIM	0.6948	0.0603	0.7137	0.0558
SIM-TM	0.6958	0.0602	0.7256	0.0500
ETA	0.6962	0.0602	0.7334	0.0500
SDIM	0.6979	0.0601	0.7355	0.0498
NINN	0.6965	0.0602	0.7017	0.0515
DSIN	0.6932	0.0604	0.7286	0.0503
TWIN	0.6984	0.0600	0.7394	0.0493
DGIN	**0.7028***	**0.0598***	**0.7663***	**0.0488***

5 EXPERIMENT RESULTS

5.1 Overall Performance

Table 2 shows the results of all methods. DGIN obtains the best performance in both the Industry and Taobao datasets, which shows the effectiveness of DGIN. There are some insightful findings from the results. (1) The proposed DGIN reaches the best performance on both datasets. Compared with existing user behavior sequence modeling methods, DGIN can extract comprehensive un-biased interest and psychological decision interest from the lifelong behavior sequence in an end-to-end manner. Both two interests achieve more fine-grained user understanding, which contributes the performance improvement. (2) DIEN performs better than DIN, which indicates the necessity of temporal information in extracting the user's interest. (3) The performance is significantly improved by modeling multiple types of behavior sequences. DMT, DMBIN, and TEM4CTR take various behavior sequences to achieve better performance than methods DIN and DIEN. Compared to click behavior sequence, multiple types of behavior sequences can comprehensively reflect the user's various interests from different perspectives. (4) SIM-TM outperforms SIM, which indicates the effectiveness of refinement and then activation paradigm on subsequence who holds the same interest key with candidate item. (5) Long-term interest obtained from the lifelong/long behavior sequence boosts the CTR prediction accuracy further. For example, ETA, SDIM, and TWIN gain higher AUC than the former methods of focusing on the short single/multiple types of behavior sequence(s). SIM also holds an improvement compared to those methods except for DMBIN and TEM4CTR. The result reveals that the long-term interest reflects drifting and periodicity, and this long correlation is necessary for understanding the user comprehensively, which is impossible for a short behavior sequence. (6) The performance-increasing trend among SIM, ETA, SDIM, NINN, and TWIN demonstrates finer granularity end-to-end training is effective. The totally two-stage

Table 3: Results of Integrating Each Component Successively.

	Industry		Taobao	
	AUC	Logloss	AUC	Logloss
TWIN	0.6984	0.0600	0.7394	0.0493
DGIN-simple	0.6960	0.0602	0.7006	0.0517
+Statistical Attributes	0.6973	0.0602	0.7364	0.0500
+Aggregated Attribute	0.6998	0.0600	0.7423	0.0495
+TM (DGIN)	**0.7028**	**0.0598**	**0.7663**	**0.0488**

method SIM performs worst. SDIM uses more precise multi-round hash collision than LSH & Hamming distance in ETA to approximate the relevance truth between candidate and behaviors in an end-to-end manner, which achieves better ranking ability. TWIN performs best among baselines with the help of consistent relevance score between the two stages brought by shared efficient target attention. (7) DSIN encounters severe performance degradation compared to other lifelong behavior sequence modeling methods. This result indicates that taking session as interest key leads to much information loss. We think the reason is that behaviors of the same session belong to various interests and self-attention together with Bi-LSTM are hard to preserve the behaviors' heterogeneity. Our proposed DGIN really does the end-to-end full information training on the full lifelong sequence by means of reasonable interest grouping. There is hardly any information loss in the DGIN, which is the key factor for improvement.

5.2 Ablation Study

In this section, we investigate the effect of each component in DGIN and display the result in Table 3. The best baseline TWIN is for comparison. All the variants stem from DGIN-simple. DGIN-simple just groups the lifelong behavior sequence and only keeps the identity attributes, which abandons lots of information. We integrate statistical attributes, aggregated attributes, and the candidate-aware subsequence to the DGIN one after another. From Table 3, we can find that all three components are beneficial for CTR prediction on both datasets. The contributions of the interest intensity contained in statistical attributes, the temporal-space information involved in aggregated attributes, and the psychological decision interest brought by candidate-aware subsequence to performance improvement are not mutually exclusive. When we add each of them into the DGIN-simple successively, the effectiveness gains improvement step by step. The results indicate that the interests hidden in the lifelong behavior sequence are multifarious. It's necessary to design dedicated features or algorithms to capture the corresponding interest character. That is what DGIN does.

5.3 The Effect of Multiple Types of Behaviors

In this section, we explore the influence of introducing multiple types of behaviors. The click-only means that we apply the DGIN to the lifelong click sequence and the *behavior_type* related attributes are removed. From Table 4, We have two findings. First, the click-only variant of DGIN still outperforms the strong competitor TWIN. This comparison verifies that the information exploitation of one-stage is better than the two-stage methods. Second, DGIN beats

Table 4: AUC and Logloss of modeling different types of behavior sequences on both datasets.

	Industry		Taobao	
	AUC	Logloss	AUC	Logloss
TWIN	0.6984	0.0600	0.7394	0.0493
click-only	0.7003	0.0600	0.7442	0.0497
DGIN	**0.7028**	**0.0598**	**0.7663**	**0.0488**

Table 5: AUC and Logloss of taking different interest key.

	Industry		Taobao	
	AUC	Logloss	AUC	Logloss
TWIN	0.6984	0.0600	0.7394	0.0493
session (DSIN)	0.6932	0.0604	0.7286	0.0503
category_id/wo TM	0.6985	0.0601	0.7408	0.0496
category_id	0.7015	0.0599	0.7639	0.0490
item_id/wo TM	0.6998	0.0600	0.7423	0.0495
item_id (DGIN)	**0.7028**	**0.0598**	**0.7663**	**0.0488**

the click-only variant on both datasets. Multiple types of behaviors indeed empower the CTR model to understand the user's preference more fine-grained and comprehensively.

5.4 The Choice of Interest Key

We investigate the choice of interest key to performance in this section. We first use session as interest key and then coarse-grained category_id. We treat different item_ids within each group when grouping on category_id also as spatio-temporal attribute and use MHSA to aggregate them. Meanwhile, we also add some new statistical attributes including the total number of item_id and the number of unique item_id. To analyze the impact of the interest key on each component of DGIN thoroughly, we include the result of DGIN's variant which has no TM represented as "wo TM". The result is shown in Table 5 and we have the following observations. DSIN, taking session as the interest key, performs worst because the heterogeneity within each session is hard to model. Both the DGIN and its variant gain better results than TWIN, which shows the robustness of DGIN. The performance drop can be alleviated by supplementing informative human-designed attributes. On the other hand, item_id performs better than category_id as the grouping key. The result indicates that fine granularity grouping retains more information and benefits the latter interest extraction.

5.5 Deployment

The system for deploying DGIN as shown in Figure 5 contains three sub-systems: data processing, offline training, and online serving.

Data Processing: We decouple the lifelong behavior sequence processing from the other features due to its requirements of storage and latency, named **U**ser **B**ehavior **P**rocessing **E**ngine (**UBPE**). UBPE collects the user's various behaviors during the past two years and then groups them based on the item_id. We build a two-level index structure for storing the grouped full lifelong behavior
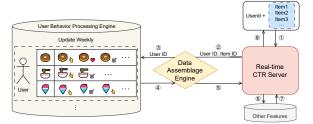

Figure 5: The whole system architecture for deploying DGIN.

sequence. The first level key is user_id and the second level key is item_id. UBPE will calculate the statistic values and discretize them within each group, which can free the latency of processing statistical attributes in the model stage. As the long-term interest of the user remains stable in a short period, UBPE updates the data once a week. When a new week of data comes, the UBPE distributes each behavior based on the two-level keys and then updates the statistic values in a streaming manner.

offline training: During the offline training stage, each training instance contains user_id and item_id. The data assemblage engine retrieves the corresponding grouped full lifelong behavior sequence based on the given keys. The two-level index structure allows high throughput query, which will not bottleneck the model training.

Online Serving: The computation bottleneck of DGIN lies in the GM as it needs to process more behaviors. Luckily, the unbiased and comprehensive interest $\text{interest}_{\text{GM}}$ can be extracted quickly by precomputing and caching the interest groups' representation $\mathbf{e_g}$ after finishing the CTR model's updating each day. When a batch of candidates comes, the inference engine only needs to retrieve the candidate-aware subsequence and obtain the psychological decision interest $\text{interest}_{\text{TM}}$.

5.6 A/B Test on Performance and Cost

We conducted an A/B test in the online LBS advertising system to measure the benefits of DGIN compared with the online baseline SIM Hard from 2023-05 to 2023-06. The DGIN is allocated with 10% experiment serving traffic and the SIM Hard holds the 70% main traffic. The online result shows the relative promotion of CTR and Revenue Per Mille (RPM) during one month's testing. DGIN achieves 4.5% and 2.0% accumulated relative promotion on the CTR and RPM respectively during the A/B test period. This is a significant improvement in the online LBS advertising system and proves the effectiveness of DGIN. The parameter storage costs of SIM and DGIN are 2.85 GB and 3.00 GB respectively. SIM spends an average inference latency of 4.6 ms and DGIN is 5.4 ms. The resource cost bought by DGIN is negligible.

6 CONCLUSION

In this paper, we propose the DGIN for full lifelong user behavior sequence modeling in the CTR prediction task. DGIN, consisting of a Group 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, DGIN is the first to achieve efficient end-to-end full lifelong user behavior sequence modeling.

REFERENCES

[1] Christopher M Bishop and Nasser M Nasrabadi. 2006. *Pattern recognition and machine learning*. Vol. 4. Springer.

[2] 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.

[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] Qiwei Chen, Huan Zhao, Wei Li, Pipei Huang, and Wenwu Ou. 2019. Behavior sequence transformer for e-commerce recommendation in alibaba. In *Proceedings of the 1st international workshop on deep learning practice for high-dimensional sparse data*. 1–4.

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

[7] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555* (2014).

[8] Mayur Datar, Nicole Immorlica, Piotr Indyk, and Vahab S Mirrokni. 2004. Locality-sensitive hashing scheme based on p-stable distributions. In *Proceedings of the twentieth annual symposium on Computational geometry*. 253–262.

[9] 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).

[10] Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional LSTM networks. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, Vol. 4. IEEE, 2047–2052.

[11] Mihajlo Grbovic and Haibin Cheng. 2018. Real-time personalization using embeddings for search ranking at airbnb. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 311–320.

[12] Yulong Gu, Zhuoye Ding, Shuaiqiang Wang, Lixin Zou, Yiding Liu, and Dawei Yin. 2020. Deep multifaceted transformers for multi-objective ranking in large-scale e-commerce recommender systems. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2493–2500.

[13] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: a factorization-machine based neural network for CTR prediction. *arXiv preprint arXiv:1703.04247* (2017).

[14] Long Guo, Lifeng Hua, Rongfei Jia, Binqiang Zhao, Xiaobo Wang, and Bin Cui. 2019. Buying or browsing?: Predicting real-time purchasing intent using attention-based deep network with multiple behavior. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 1984–1992.

[15] Tianqi He, Kaiyuan Li, Shan Chen, Haitao Wang, Qiang Liu, Xingxing Wang, and Dong Wang. 2023. DMBIN: A Dual Multi-behavior Interest Network for Click-Through Rate Prediction via Contrastive Learning. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1366–1375.

[16] Yuchin Juan, Yong Zhuang, Wei-Sheng Chin, and Chih-Jen Lin. 2016. Field-aware factorization machines for CTR prediction. In *Proceedings of the 10th ACM conference on recommender systems*. 43–50.

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

[18] Wentao Ouyang, Xiuwu Zhang, Li Li, Heng Zou, Xin Xing, Zhaojie Liu, and Yanlong Du. 2019. Deep spatio-temporal neural networks for click-through rate prediction. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2078–2086.

[19] 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.

[20] 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.

[21] Jiarui Qin, Weinan Zhang, Rong Su, Zhirong Liu, Weiwen Liu, Guangpeng Zhao, Hao Li, Ruiming Tang, Xiuqiang He, and Yong Yu. 2023. Learning to Retrieve User Behaviors for Click-through Rate Estimation. *ACM Transactions on Information Systems* 41, 4 (2023), 1–31.

[22] Jiarui Qin, Weinan Zhang, Xin Wu, Jiarui Jin, Yuchen Fang, and Yong Yu. 2020. User behavior retrieval for click-through rate prediction. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2347–2356.

[23] Steffen Rendle. 2010. Factorization machines. In *2010 IEEE International conference on data mining*. IEEE, 995–1000.

[24] Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends® in Information Retrieval* 3, 4 (2009), 333–389.

[25] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[26] Ruoxi Wang, Bin Fu, Gang Fu, and Mingliang Wang. 2017. Deep & cross network for ad click predictions. In *Proceedings of the ADKDD'17*. 1–7.

[27] Raymond E Wright. 1995. Logistic regression. (1995).

[28] Ruobing Xie, Cheng Ling, Yalong Wang, Rui Wang, Feng Xia, and Leyu Lin. 2021. Deep feedback network for recommendation. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*. 2519–2525.

[29] Weinan Xu, Hengxu He, Minshi Tan, Yunming Li, Jun Lang, and Dongbai Guo. 2020. Deep interest with hierarchical attention network for click-through rate prediction. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*. 1905–1908.

[30] Bei Yang, Jie Gu, Ke Liu, Xiaoxiao Xu, Renjun Xu, Qinghui Sun, and Hong Liu. 2023. Empowering General-purpose User Representation with Full-life Cycle Behavior Modeling. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2908–2917.

[31] Hengyu Zhang, Chang Meng, Wei Guo, Huifeng Guo, Jieming Zhu, Guangpeng Zhao, Ruiming Tang, and Xiu Li. 2023. Time-aligned Exposure-enhanced Model for Click-Through Rate Prediction. *arXiv preprint arXiv:2308.09966* (2023).

[32] Yuren Zhang, Enhong Chen, Binbin Jin, Hao Wang, Min Hou, Wei Huang, and Runlong Yu. 2022. Clustering based behavior sampling with long sequential data for CTR prediction. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2195–2200.

[33] Keke Zhao, Xing Zhao, Qi Cao, and Linjian Mo. 2022. A Non-sequential Approach to Deep User Interest Model for CTR Prediction. In *Proceedings of the 2022 SIAM International Conference on Data Mining (SDM)*. SIAM, 531–539.

[34] Chang Zhou, Jinze Bai, Junshuai Song, Xiaofei Liu, Zhengchao Zhao, Xiusi Chen, and Jun Gao. 2018. Atrank: An attention-based user behavior modeling framework for recommendation. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

[35] 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.

[36] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. Deep interest network for click-through rate prediction. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 1059–1068.

[37] Meizi Zhou, Zhuoye Ding, Jiliang Tang, and Dawei Yin. 2018. Micro behaviors: A new perspective in e-commerce recommender systems. In *Proceedings of the eleventh ACM international conference on web search and data mining*. 727–735.

[38] 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).