

Deep Evolutional Instant Interest Network for CTR Prediction in Trigger-Induced Recommendation

Zhibo Xiao
Alibaba Group
Hangzhou, Zhejiang, China
xiaozhibo.xzb@alibaba-inc.com

Luwei Yang*
Alibaba Group
Hangzhou, Zhejiang, China
luwei.ylw@alibaba-inc.com

Tao Zhang
Alibaba Group
Hangzhou, Zhejiang, China
selous.zt@alibaba-inc.com

Wen Jiang
Alibaba Group
Hangzhou, Zhejiang, China
wen.jiangw@alibaba-inc.com

Wei Ning
Alibaba Group
Hangzhou, Zhejiang, China
wei.ningw@alibaba-inc.com

Yujiu Yang
SIGS, Tsinghua University
Shenzhen, Guangdong, China
yang.yujiu@sz.tsinghua.edu.cn

ABSTRACT

The recommendation has been playing a key role in many industries, e.g., e-commerce, streaming media, social media, etc. Recently, a new recommendation scenario, called Trigger-Induced Recommendation (TIR), where users are able to explicitly express their instant interests via trigger items, is emerging as an essential role in many e-commerce platforms, e.g., Alibaba.com and Amazon. Without explicitly modeling the user's instant interest, traditional recommendation methods usually obtain sub-optimal results in TIR. Even though there are a few methods considering the trigger and target items simultaneously to solve this problem, they still haven't taken into account temporal information of user behaviors, the dynamic change of user instant interest when the user scrolls down and the interactions between the trigger and target items. To tackle these problems, we propose a novel method – Deep Evolutional Instant Interest Network (DEI2N), for click-through rate prediction in TIR scenarios. Specifically, we design a User Instant Interest Modeling Layer to predict the dynamic change of the intensity of instant interest when the user scrolls down. Temporal information is utilized in user behavior modeling. Moreover, an Interaction Layer is introduced to learn better interactions between the trigger and target items. We evaluate our method on several offline and real-world industrial datasets. Experimental results show that our proposed DEI2N outperforms state-of-the-art baselines. In addition, online A/B testing demonstrates the superiority over the existing baseline in real-world production environments.

CCS CONCEPTS

• **Information systems** → **Personalization; Recommender systems; Learning to rank.**

*Corresponding author.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

WSDM '24, March 4–8, 2024, Merida, Mexico

© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 979-8-4007-0371-3/24/03...\$15.00

<https://doi.org/10.1145/3616855.3635829>

KEYWORDS

Recommender Systems; Click-Through Rate Prediction; User Instant Interests; Trigger-Induced Recommendation

ACM Reference Format:

Zhibo Xiao, Luwei Yang, Tao Zhang, Wen Jiang, Wei Ning, and Yujiu Yang. 2024. Deep Evolutional Instant Interest Network for CTR Prediction in Trigger-Induced Recommendation. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining (WSDM '24)*, March 4–8, 2024, Merida, Mexico. ACM, New York, NY, USA, 9 pages. <https://doi.org/10.1145/3616855.3635829>

1 INTRODUCTION

Personalized recommendation systems are extensively employed in the industry. Taking an e-commerce app as an example, we describe two important recommendation scenarios in real industrial platforms applying CTR prediction extensively, User-Induced Recommendation and Trigger-Induced Recommendation, which are shown by Figure 1. The left part shows the *Just for You* module, which is responsible for recommending items according to the user's past interests or behaviors (if permitted by the user). The recommended items in this module are diversified according to the user's historical interests. This scenario is referred to as User-Induced Recommendation (UIR).

Once the user clicks an item, he/she is introduced to a new module, named *Mini Detail*, which is shown in the middle part of Figure 1. Note that, the clicked item in the previous step is presented at the top, which is referred to as the trigger item. The user is able to either click an item to enter the *Item Detail* page (the right part), or scroll down to access more recommended items. These recommended items in *Mini Detail* should be related to the trigger item to some extent. This scenario is often referred to as Trigger-Induced Recommendation (TIR). Besides the *Mini Detail* module, it is very common to see other TIRs, e.g. a *Detail Recommendation* module in *Item Detail* page. Nowadays, TIR is playing an increasingly significant role in many industrial domains, such as e-commerce platforms [15] and messaging APPs [21]. In our app, more than 50% of active buyers are contributed by TIR among all recommendation scenarios.

Click-through rate (CTR) prediction plays a crucial role in the recommendation. The main goal is to estimate the likelihood that an item will be clicked by a user. It has a direct and immediate

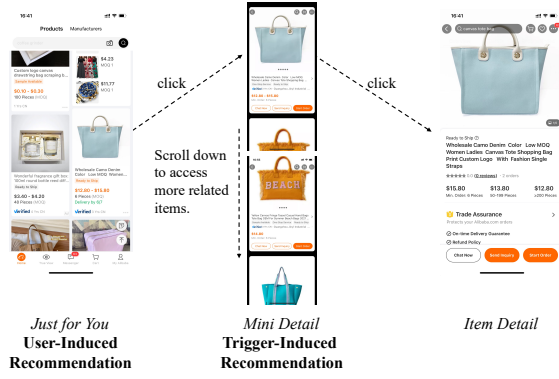


Figure 1: Recommendation scenarios at an e-commerce app. Left: User-Induced Recommendation, middle: Trigger-Induced Recommendation, right: Item Detail.

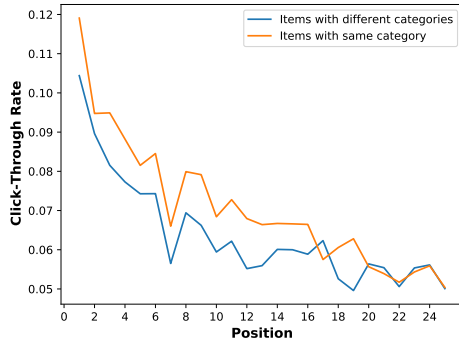


Figure 2: The average click-through rate of items with the same category as the trigger item and items with different categories along the position.

impact on website revenues and user satisfaction, especially in e-commerce. Traditional CTR methods [3, 8, 20, 22, 23], which are more suitable for UIR, have been used extensively in many domains. However, applying it rigidly to TIR could fail to model the instant interest of the user, which results in sub-optimal results.

In this paper, we focus on how to accurately estimate the click-through rate of items in Trigger-Induced Recommendation scenarios. TIR has attracted growing interest in the industry, nonetheless, there is a lack of research on it. R3S [21] introduced feature interaction, semantic similarity and information gain to capture users' instant interests. However, it doesn't consider users' historical behaviors, which is one of the most important features in click-through rate prediction modeling. DIHN [15] proposed an interest highlight network to learn the instant interest from the trigger item and the user's historical behaviors. DIAN [19] proposed an intent-aware network to learn the user's intention. However, the temporal information of behaviors, the dynamic change of user instant interest when the user scrolls down and the interactions between the trigger and target items haven't been considered. Specifically, when the user scrolls down, the intensity of instant interest will change dynamically. As shown in Figure 2, which is based on statistics from a real TIR scene, this phenomenon is confirmed by the decreasing gap of CTR between the items with the same category as the trigger item and the items with different categories when the user scrolls down. Therefore, it is highly beneficial

to keenly capture the dynamic change of the intensity of instant interest, which is neglected in existing methods [15, 19].

To tackle the aforementioned challenges, we propose a novel method called Deep Evolutional Instant Interest Network (DEI2N¹) for CTR in TIR scenarios. Specifically, we introduce a User Instant Interest Modeling Layer to predict the dynamic change in the intensity of instant interest when the user scrolls down. This layer is responsible for modeling user instant interest by considering the trigger item and user behaviors simultaneously. Moreover, we integrate temporal information into the attention units to improve the sequence modeling and capture better relevance of the user's interests with respect to the target item and trigger item respectively. Additionally, an Interaction layer is utilized to learn the interaction relationship between the features of the trigger item and target items.

The main contributions of this paper are summarized as follows:

- We emphasize an emerging industrial recommendation scenario, Trigger-Induced Recommendation, and highlight the challenges of existing CTR methods applied in TIR.
- We propose a novel method DEI2N, which further improves CTR performances in TIR scenarios by considering the dynamic change of user instant interest, temporal information, and the interactions between the trigger and target items.
- We evaluate our method DEI2N on three real-world industrial datasets with state-of-the-art methods. Our method achieves the best performance among competitors. The ablation experiments further verify the effectiveness of the proposed components.
- We implement DEI2N in industrial production environments and launch it in five industrial e-commerce TIR scenes. The results of online A/B testing demonstrate the superiority over the existing baseline.

2 RELATED WORK

As Trigger-Induced Recommendation is an emerging recommendation scenario, we will start with an overview of the CTR prediction task to the most related works in this area. We briefly review three groups of existing methods, 1) Feature Interaction Modeling, 2) User Behavior Modeling, and 3) Trigger-Induced Recommendation.

The first group is **Feature Interaction Modeling**. Noticing the disability of feature interaction in the linear regression method, a large number of researchers have proposed alternatives to solve this problem. A Factorization Machine model is proposed by [13] to model pairwise (second-order) feature interactions. To alleviate the efforts of feature engineering, DeepCrossing [14] utilizes ResNet [6] to automatically learn interactions of features. Wide&Deep [2] creatively combines the linear model and deep network together for better memorization and generalization. Later, DeepFM [4] employs a factorization machine instead of a linear model in the wide part. By noticing the implicit feature interaction of neural networks, DCN [18] proposes a more efficient cross-network in addition to a deep network. AutoInt [16] applies the self-attention mechanism to automatically learn feature interactions. Recently, Fi-GNN [9], GraphFM [10], and DG-ENN [5] take advantage of

¹The code is released at <https://github.com/mengxiaozhibo/DEI2N>

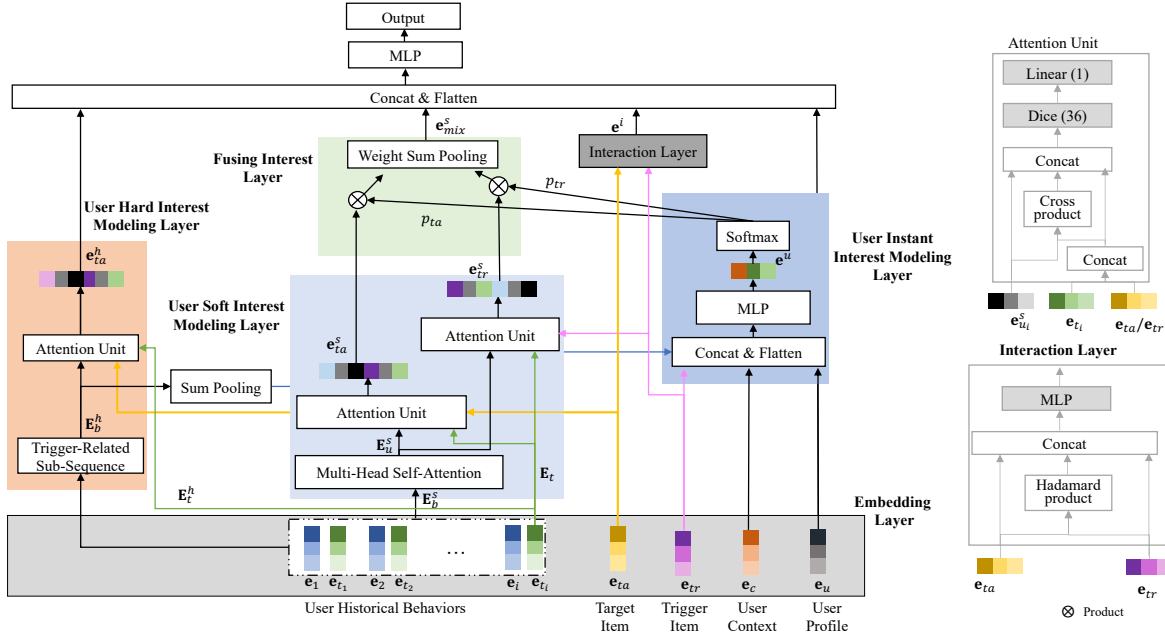


Figure 3: The architecture of the DEI2N model, which consists of Embedding Layer, User Instant Interest Modeling Layer, User Soft Interest Modeling Layer, User Hard Interest Modeling Layer, Fusing Interest Layer and Interaction Layer.

Graph Neural Networks for better feature interactions in feature sparsity scenarios.

The second group is **User Behavior Modeling**. These methods are more common in the industry. By considering user behaviors, it is able to speculate accurate short-term interests and long-term periodic interests from users. A naive usage of user behavior is just averaging or summing all the behavior embedding vectors to feed into subsequent MLPs. A representative method is DIN [23]. It innovatively uses target attention to obtain the relevant user behaviors respective to the target item. As a result, all relevant user behaviors are activated to calculate the final click-through rate, which in turn obtains better prediction. A subsequent upgraded version DIEN [22] refines GRUs to model the evolution of user interests from the user behaviors. Considering the multiple interests of user behaviors, DMIN is proposed by [20] to capture the diverse interests of users from his/her behaviors.

As an emerging recommendation scenario, there are few CTR models dedicated to **Trigger-Induced Recommendation**. Although the above traditional methods are able to serve in TIR rigidly, the lack of modeling user instant interest motivates researchers to search for better solutions. The most relevant method is DIHN [15], which is used in an industrial travel e-commerce platform. It introduces a user intent network to predict to what extent the user is interested in the trigger item. The output from this network is able to supervise the fusion of interests from the trigger or user behaviors. The second relevant method is DIAN [19], which uses an intent-aware network to learn the user's intention. The output of this network is used to dynamically balance the results of trigger-free and trigger-based recommendations. The third most relevant method is R3S [21]. It is used in reading recommendation scenarios, where the extended recommendation readings should be relevant to the current clicked reading. The current clicked reading is considered the trigger item in TIR. The

recommendations are constructed by taking into account feature interaction, semantic similarity and information gain between the current clicked reading and candidate readings. Nonetheless, the temporal information of behaviors, the dynamic change of user instant interest when the user scrolls down and the interactions between the trigger and target items haven't been considered.

3 THE PROPOSED METHOD

In this section, we introduce our proposed method, Deep Evolutional Instant Interest Network (DEI2N), for CTR in TIR scenarios. The overall architecture is illustrated by Figure 3.

We follow the basic CTR paradigm of Embedding & MLP (Multi-layer Perceptron) model [23]. There are five main components in the middle to better capture user instant interest in TIR. *User Instant Interest Modeling Layer* is responsible for modeling user instant interest by considering the trigger item and user behaviors simultaneously. Additionally, it is able to predict the dynamic change in the intensity of instant interest when the user scrolls down. *User Soft Interest Modeling Layer* and *User Hard Interest Modeling Layer* are applied to extract the user's interests from his/her behaviors according to the trigger and target items. *Fusing Interest Layer* utilizes the results of the User Instant Interest Modeling Layer to fuse the user's interests extracted from the User Soft Interest Modeling Layer. *Interaction Layer* learns the interaction relationship between the features of the trigger item and target items. Finally, all of the resulting features and remaining features are concatenated and fed into MLP layers for final CTR prediction. In the remaining section, we will describe these layers in detail.

3.1 Embedding Layer

There are five groups of input features: *User Profile*, *User Historical Behaviors*, *Trigger Item*, *Target Item* and *User Context*. *User Profile*

contains *user ID*, *country ID* and so on. *User Historical Behaviors* is a sequential list of items that the user has clicked or bought. *Trigger Item* and *Target Item* contain *item ID*, *category ID*, *company ID*, etc. *User Context* contains the *page number* that a user is currently browsing. Each feature is normally encoded into a high-dimensional one-hot vector and further is transformed into low dimensional dense features by utilizing embedding layers [2]. For example, the *user ID* can be represented by a matrix $\mathbf{E} \in \mathbb{R}^{K \times d_o}$, where K is the total number of users and d_o is the embedding size with $d_o \ll K$. Transformed by embedding layers, *User Profile*, *User Historical Behaviors*, *Trigger Item*, *Target Item* and *User Context* are represented as \mathbf{e}_u , \mathbf{E}_b , \mathbf{e}_{tr} , \mathbf{e}_{ta} and \mathbf{e}_c , respectively. Note that, $\mathbf{E}_b = \{\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_T\} \in \mathbb{R}^{T \times d_{model}}$, where T represents the length of user historical behaviors and d_{model} is the dimension of item embedding \mathbf{e}_i .

As temporal information is crucial in sequence modeling [8], we introduce the time interval between the historical behavior interaction and the recommendation time. In the formula, the time interval of a historical behavior item is calculated by $t_i = \lfloor ((t - \hat{t}_i)/T_f) \rfloor$, where t is the recommendation timestamp, \hat{t}_i is the behavior interaction timestamp for item i , $\lfloor \cdot \rfloor$ is the floor function, and T_f is an adjustable normalization factor. We then apply embedding lookup to obtain the time interval embedding \mathbf{e}_{t_i} . Thus, the time interval representation of the user's historical behaviors is formulated as $\mathbf{E}_t = \{\mathbf{e}_{t_1}, \mathbf{e}_{t_2}, \dots, \mathbf{e}_{t_T}\} \in \mathbb{R}^{T \times d_{time}}$, where d_{time} is the dimension of the time interval embedding.

3.2 User Instant Interest Modeling Layer

In TIR scenarios, the clicked trigger item explicitly represents the user's instant interests. Thus, at the beginning, the user is more interested in the items with the same category as the trigger item. However, when the user scrolls down, the intensity of instant interest will change dynamically. This phenomenon is confirmed by the decaying gap of CTR between the items with the same category as the trigger item and the items with different categories when the user scrolls down. Therefore, it is highly beneficial to keenly capture the dynamic change of the intensity of instant interest upon scrolling down, which is neglected in existing methods [15, 19].

We propose the User Instant Interest Modeling Layer to predict the dynamic change of the intensity of instant interest upon the user scrolls down. In this layer, we utilize four categories of features, i.e., *User Profile*, *User Context*, *Trigger Item* and the results of sum pooling of the trigger-related sub-sequence as inputs and then feed them into MLPs to generate two probability scores, p_{tr} and p_{ta} with $p_{tr} + p_{ta} = 1$. They are formulated as,

$$p_{tr}, p_{ta} = \text{Softmax}(\text{MLP}(\mathbf{e}_u, \mathbf{e}_c, \mathbf{e}_{tr}, \text{sum}(\mathbf{E}_b^h))), \quad (1)$$

where \mathbf{E}_b^h represents the trigger-related sub-sequence containing the behaviors with the same category as the trigger item. Note that \mathbf{E}_b^h will comprise the most recently interacted item only if this item belongs to the same category as the trigger item. Thus, p_{tr} and p_{ta} represent the extent of how relevant the trigger item and the target item are to user historical behaviors respectively. In other words, it is responsible for determining to what extent the user is interested in the trigger item or target item. Note that \mathbf{e}_c contains the page number that the user is currently browsing, as we find the page

number is a strong signal indicating the evolution of the intensity of user instant interest.

3.3 User Soft Interest Modeling Layer

In traditional CTR prediction methods [3, 22, 23], user interest modeling is usually implemented by calculating the relevant weights between user historical behaviors and the target item. However, applying this technique rigidly to TIR would result in non-optimal results. Because the trigger item indicates a strong signal of the user's instant interest. It is inevitable to take both the trigger and target items into account simultaneously.

We propose the User Soft Interest Modeling Layer to extract users' interests with respect to the trigger and target items simultaneously by following [15]. In addition to using Multi-Head Self-Attention (MHSA) [17] to refine the item representation from user historical behaviors, we introduce residual connection [6], dropout [7] and layer normalization [1] to further improve the item representation. To explicitly introduce temporal information, the input of MHSA is denoted as \mathbf{E}_b^s , which is a concatenation of user historical behavior embeddings \mathbf{E}_b and time interval embeddings \mathbf{E}_t . The MHSA is formulated as:

$$\mathbf{E}_u^s = \text{MHSA}(\mathbf{E}_b^s) = \text{Concat}(\text{head}_1, \text{head}_2, \dots, \text{head}_{H_R}) \mathbf{W}^O, \quad (2)$$

$$\begin{aligned} \text{head}_h &= \text{Attention}(\mathbf{E}_b^s \mathbf{W}_h^Q, \mathbf{E}_b^s \mathbf{W}_h^K, \mathbf{E}_b^s \mathbf{W}_h^V) \\ &= \text{Softmax}\left(\frac{\mathbf{E}_b^s \mathbf{W}_h^Q \cdot (\mathbf{E}_b^s \mathbf{W}_h^K)^\top}{\sqrt{d_h}}\right) \cdot \mathbf{E}_b^s \mathbf{W}_h^V, \end{aligned} \quad (3)$$

where $\mathbf{W}_h^Q, \mathbf{W}_h^K, \mathbf{W}_h^V \in \mathbb{R}^{d_{model} \times d_h}$ are projection matrices of the h -th head for query, key and value respectively. The H_R is the number of heads and $\mathbf{W}^O \in \mathbb{R}^{d_{model} \times d_{model}}$ is a linear matrix. The d_h represents the dimension of each head and head_h represents a latent item representation in subspace.

Next, we apply two attention units to extract the user's interests with respect to the target item and the trigger item separately. Besides and more importantly, the temporal information \mathbf{E}_t is utilized in these two attention units to improve the sequence modeling and capture better relevance of the user's interests with respect to the target item and trigger item respectively. Note that, the temporal information is the time interval representation of the user's historical behaviors. The process of the target and trigger attention mechanism can be formulated as:

$$\mathbf{e}_{ta}^s = \sum_{j=1}^T a(\mathbf{e}_{u_j}^s; \mathbf{e}_{t_j}; \mathbf{e}_{ta}) \mathbf{e}_{u_j}^s = \sum_{j=1}^T w_{ta_j} \mathbf{e}_{u_j}^s, \quad (4)$$

$$\mathbf{e}_{tr}^s = \sum_{j=1}^T a(\mathbf{e}_{u_j}^s; \mathbf{e}_{t_j}; \mathbf{e}_{tr}) \mathbf{e}_{u_j}^s = \sum_{j=1}^T w_{tr_j} \mathbf{e}_{u_j}^s, \quad (5)$$

where $\mathbf{e}_{u_j}^s \in \mathbb{R}^{d_{model}}$ represents the j -th item representation after applying MHSA, $\mathbf{e}_{t_j} \in \mathbb{R}^{d_{time}}$ represents the j -th item time interval embedding, a is the attention unit which is shown on the top right of Figure 3.

3.4 User Hard Interest Modeling Layer

Motivated by the hard sequential modeling used in SIM [12] and DIHN [15], we propose the User Hard Interest Modeling Layer. A trigger-related sub-sequence, containing the behaviors with the

same category as the trigger item, is aggregated to complement the extraction of users' instant interests. This mechanism helps to filter out irrelevant noise and covers a longer period of user historical behaviors. It is formulated as $\mathbf{E}_b^h = \{\mathbf{e}_{b_1}^h, \mathbf{e}_{b_2}^h, \dots, \mathbf{e}_{b_{T_h}}^h\} \in \mathbb{R}^{T_h \times d_{model}}$, where T_h is the length of the sub-sequence. Similarly, the time interval representation of sub-sequence can be formulated as $\mathbf{E}_t^h = \{\mathbf{e}_{t_1}^h, \mathbf{e}_{t_2}^h, \dots, \mathbf{e}_{t_{T_h}}^h\} \in \mathbb{R}^{T_h \times d_{time}}$. Then, we apply the same attention unit used in the previous section to capture the relevance of the user's interests with respect to the target item. Since this sub-sequence is already related to the trigger item, it is not necessary to apply the attention unit with respect to the trigger item.

Finally, the output of this layer is calculated as,

$$\mathbf{e}_{ta}^h = \sum_{j=1}^{T_h} a(\mathbf{e}_{b_j}^h; \mathbf{e}_{t_j}^h; \mathbf{e}_{ta}) \mathbf{e}_{b_j}^s = \sum_{j=1}^{T_h} w_{ta_j} \mathbf{e}_{b_j}^s, \quad (6)$$

where $\mathbf{e}_{b_j}^h \in \mathbb{R}^{d_{model}}$ represents the j -th item representation in trigger-related sub-sequence. The $\mathbf{e}_{t_j}^h \in \mathbb{R}^{d_{time}}$ represents the j -th item time interval embedding in trigger-related sub-sequence.

3.5 Fusing Interest Layer

In order to better model user instant interest by considering the trigger item, target item and user behaviors simultaneously. We propose the Fusing Interest Layer to utilize the results of the User Instant Interest Modeling Layer to fuse the two user interest representations extracted from the User Soft Interest Modeling Layer. Mathematically, it is defined as:

$$\mathbf{e}_{mix}^s = p_{tr} \cdot \mathbf{e}_{tr}^s + p_{ta} \cdot \mathbf{e}_{ta}^s, \quad (7)$$

where p_{tr}, p_{ta} are the predicted probabilities extracted in the User Instant Interest Modeling Layer and $\mathbf{e}_{tr}^s, \mathbf{e}_{ta}^s$ are user interest representations extracted in the User Soft Interest Modeling Layer with respect to the trigger and target items respectively.

3.6 Interaction Layer

An Interaction Layer, shown on the bottom right of Figure 3, is introduced to learn the explicit interaction relationship between the features of the trigger and target items. It takes the trigger and target items as input and then applies Hadamard product and MLP layers to learn high-order feature interactions,

$$\mathbf{e}^i = \text{MLP}(\mathbf{e}_{ta}; \mathbf{e}_{tr}; \mathbf{e}_{tr} \times \mathbf{e}_{ta}), \quad (8)$$

where \times means the Hadamard product, aka element-wise product.

3.7 Loss Function

Finally, all the feature vectors $\mathbf{e}_{mix}^s, \mathbf{e}_{ta}^h, \mathbf{e}^i$ and \mathbf{e}_u are concatenated and then fed into MLP layers for CTR prediction. We adopt the binary cross-entropy loss as the loss function, which is widely used in CTR prediction tasks [15, 20, 22, 23],

$$L = -\frac{1}{N} \sum_{(\mathbf{x}, y) \in S} (y \log(f(\mathbf{x})) + (1 - y) \log(1 - f(\mathbf{x}))), \quad (9)$$

where S is the training set of size N , \mathbf{x} is the input of network which is a concatenation of $\mathbf{e}_{mix}^s, \mathbf{e}_{ta}^h, \mathbf{e}^i$ and \mathbf{e}_u , $y \in \{0, 1\}$ is the click label and $f(\mathbf{x})$ is the prediction output of the proposed model.

Table 1: Statistics of the offline datasets.

Dataset	Users	Items	Categories	Samples
Alibaba.com	373,852	4,715,150	6,736	5,200,000
Alimama	500,000	846,812	12,978	8,552,702
ContentWise	26,186	1,268,988	117,693	2,585,070

4 EVALUATION

4.1 Datasets

We use three real-world datasets for evaluation. One of them is collected from our real-world industrial e-commerce TIR scenario, named Alibaba.com. The other two of them, Alimama and ContentWise, are tailored from existing public e-commerce and media service datasets to fit the TIR problem. The two of them are sampled from e-commerce platform user logs, and the one among them is obtained from media service. The statistics of them are summarized in Table 1.

Alibaba.com. As there is no public TIR dataset, we create a dataset from an Alibaba.com TIR scenario, *Mini Detail*, which is shown in the middle of Figure 1. The clicked item in the previous step, which is presented at the top in *Mini Detail*, is referred to as the trigger item. The label is set to positive when a user clicks an item on an exposed list of items in *Mini Detail*, otherwise the label is set to 0.

Alimama². To demonstrate the effectiveness of our method, we tailored this dataset to fit the TIR problem by manually creating trigger items. We follow the scheme in [15], the latest clicked item within 4 hours before a sample is deemed as the trigger item. Samples that can not be associated with a trigger will not be selected. The label is obtained same as Alibaba.com dataset.

ContentWise [11]. To evaluate our method on different domains, we introduce a media service dataset which is constructed from an Over-The-Top Media service. The media contents, including movies, movies and clips in series, TV movies or shows, and episodes of TV series, are provided to users by Internet connections. Represented by a recommendation list, the users are able to view the media items, access the media item detail, purchase the media items, or rate the media items. Since it lacks trigger information, we follow a similar scheme in [15] to manually create trigger items. Due to the sparsity of the dataset, the latest clicked item within 8 hours before a sample is deemed as the trigger item. Samples that cannot be associated with a trigger will be eliminated. The label is set to positive when a user either views, purchases, rates or accesses the media items. As there is no exposed list items, we follow [22] to randomly select items as negative samples.

4.2 Compared Methods

To demonstrate the effectiveness of our proposed method, we compare it with several state-of-the-art methods: Wide&Deep [2], DIN [23], DIEN [22], DMIN [20], DIHN [15] and DIAN³ [19]. Besides, we equip some compared methods with the capacity of instant interest modeling for fair and solid comparisons.

- **Wide&Deep+TR** adds the trigger item as input to capture the user's instant interests.

²<https://tianchi.aliyun.com/dataset/dataDetail?dataId=56>

³The code is not open-sourced, we reproduce it by ourselves.

- **DIN+TRA** applies an attention mechanism to extract the user’s instant interests with respect to the trigger item, besides the existing target attention.
- **DIEN+TRA** utilizes the similar attention strategy used in DIN+TRA to better model both the user’s instant interest and the user’s evolved interest.
- **DMIN+TRA** employs the similar attention strategy used in DIN+TRA to capture the user’s instant interest while extracting multiple interests from user historical behaviors. This is the baseline method that we compared in the online A/B testing experiments.

4.3 Parameter Settings

For parameter setting, d_{model} , d_{time} , the dimension of the *user profile* and *user context* are set as 72, 36, 36, and 10, respectively. The learning rate is set as 0.001 and the dropout rate is set as 0.1. The number of heads H_R used in MHSA is set as 2. The normalization factor T_f is uniformly set to 60, which means we calculate time interval features in minutes. The maximum length of the user behavior sequence is set as 20, 30, and 30 for Alibaba.com, Alimama and ContentWise, respectively; and the maximum length of user trigger-related behavior sub-sequence as 10, 20, and 10 for Alibaba.com, ContentWise and Alimama, respectively. The hidden layer dimensions of the final MLP align with that of the DIEN model at 200 and 80. Additionally, the MLP in the Interaction Layer employs hidden layers of size 144 and 72, while the User Instant Interest Modeling Layer employs hidden layers with sizes 72 and 36. The implementations of baselines are acquired from their released repositories. The Grid Search technique is applied to find the optimal hyper-parameters.

4.4 Performance Comparison

We use the Area Under ROC (AUC) and RelAImpr as evaluation metrics, which are widely applied in CTR prediction tasks [3, 20, 22, 23]. The experimental results on three real-world datasets are shown in Table 2.

We find that the traditional methods, namely Wide&Deep, DIN, DIEN, and DMIN, do not perform well in the TIR scenario, especially in the Alibaba.com dataset. The gaps between the original version and the one equipped with the trigger item are more than 20 percent. The main reason for this is that these methods do not take the trigger item into account. Once we equip them with the capacity of instant interest modeling, their performances are further improved. These results also show the necessity of elaborate modeling in TIR by considering the trigger item.

For the sake of fairness, we compare the proposed method DEI2N with DIHN, DIAN and traditional methods equipped by the trigger item. DMIN+TRA is a strong competitor among traditional competitors, which achieves the best results among them. Additionally, DIAN, a specialized method for TIR, achieves better results compared with improved versions of traditional methods except for ContentWise. Because it is able to adaptively model both the trigger and target items simultaneously.

Our proposed method DEI2N obtains the highest AUC value among all state-of-the-art methods. The results demonstrate the effectiveness of explicitly considering the dynamic change of user instant interest when the user scrolls down. It allows the model

to be aware of the context in order to adaptively fuse user interest representations with respect to the trigger and target items. Besides, modeling of temporal information of user historical behaviors, and the explicit interactions between the trigger and target items contribute to these results as well. We find that the AUC gains of DEI2N over DIAN on Alimama (0.19%) and ContentWise (0.78%) are not obvious as on Alibaba.com (2.55%). One of the possible reasons is that these two datasets are not directly collected from TIR scenarios. The synthesized trigger item may not reflect the real situation in TIR. Furthermore, the lack of context features (e.g., page number) on these two datasets prevents us from modeling the dynamic change of the user’s instant interest when the user scrolls down. Consequently, it may limit our model’s performance.

4.5 Ablation Study

To understand the effectiveness of the proposed components, we evaluate our proposed method DEI2N in ablation settings. As the Alibaba.com dataset is directly collected from a real-world TIR scenario, we will present ablation results on this dataset. These results are more realistic and better to show the value of our proposed model. The ablation experimental results are shown in Table 3.

To evaluate the effects of the User Instant Interest Modeling layer, we remove this layer as DEI2N-NO-UI2M and compare it with DEI2N. Without explicitly modeling user instant interest, the performance is degraded from an AUC value of 0.7671 to 0.7534. This explicitly shows the benefits of UI2M, which is responsible for predicting the dynamic change of the intensity of instant interest as the user scrolls down. It controls the proportion of the recommended items related to the trigger item. The temporal information on user behaviors is very important. Without the temporal information modeling, DEI2N-NO-TIM degrades the performance from 0.7671 to 0.7652. Temporal information is used in the sequence modeling MHSA and the trigger and targets attention mechanisms in User Soft Interest Modeling Layer and User Hard Interest Modeling Layer. The necessity of explicitly modeling the interactions between the trigger and target items is shown by comparing DEI2N-NO-IL with DEI2N. DEI2N obtains 1.20% relative improvement by introducing explicit interactions between the trigger and target items. The significance of user hard interest and soft interest modeling is represented by the ablation results of DEI2N-NO-UHIM and DEI2N-NO-USIM compared with DEI2N. Without User Hard Interest Modeling Layer and User Soft Interest Modeling Layer, the AUC values are degraded from 0.7671 to 0.7651 and 0.7504 respectively.

4.6 Time Analysis

In this section, we compare our proposed method with other baselines in training and prediction time. We recorded the training time of these methods on the training set and the test set respectively on the Alibaba.com dataset. The epoch number is set to one. The machine has 41 CPU cores of Intel(R) Xeon(R) Platinum 8163 CPU @ 2.50GHz and 330 GB memory, with 1 NVIDIA Tesla V100 GPU.

The training and prediction time results are shown in Table 4, which indicates the efficiency of DEI2N is comparable to that of these baselines. Because of the extra execution of the trigger item information, all of the traditional baselines are slower than that of their original versions. DIEN+TRA and DMIN+TRA cost 641

Table 2: Experimental AUC results on real-world datasets. The bold number in each column indicates the best result, while the underlined number in each column is the second best result.

Model	Alibaba.com		Alimama		ContentWise	
	AUC	RelaImpr	AUC	RelaImpr	AUC	RelaImpr
Wide&Deep	0.6096 \pm 0.0019	−0.99%	0.6062 \pm 0.0008	−7.97%	0.9469 \pm 0.0003	−7.28%
DIN	0.6042 \pm 0.0016	−5.87%	0.6154 \pm 0.0007	0.00%	0.9774 \pm 0.0002	−0.95%
DIEN	0.6047 \pm 0.0025	−5.42%	0.6155 \pm 0.0005	0.09%	0.9779 \pm 0.0013	−0.85%
DMIN	0.6107 \pm 0.0011	0.00%	0.6154 \pm 0.0002	0.00%	0.9820 \pm 0.0002	0.00%
Wide&Deep+TR	0.7412 \pm 0.0014	111.89%	0.6075 \pm 0.0018	−6.84%	0.9713 \pm 0.0004	−2.22%
DIN+TRA	0.7425 \pm 0.0021	119.06%	0.6155 \pm 0.0015	0.09%	0.9803 \pm 0.0019	−0.35%
DIEN+TRA	0.7419 \pm 0.0019	118.52%	0.6157 \pm 0.0004	0.26%	0.9796 \pm 0.0015	−0.50%
DMIN+TRA	0.7454 \pm 0.0007	121.68%	0.6157 \pm 0.0003	0.26%	<u>0.9822 \pm 0.0003</u>	<u>0.04%</u>
DIHN	0.7462 \pm 0.0006	122.40%	0.6166 \pm 0.0008	1.04%	0.9786 \pm 0.0012	−0.75%
DIAN	<u>0.7480 \pm 0.0016</u>	<u>124.03%</u>	<u>0.6168 \pm 0.0002</u>	<u>1.21%</u>	0.9764 \pm 0.0003	−1.16%
DEI2N	0.7671 \pm 0.0012 *	141.28%	0.6180 \pm 0.0005 *	2.25%	0.9840 \pm 0.0002 *	0.41%

* Asterisks represent where DEI2N’s improvement over compared methods is significant (one-sided rank-sum p-value <0.01).

Table 3: Ablation experimental results on Alibaba.com dataset.

Model	Alibaba.com	
	AUC	RelaImpr
DEI2N-NO-UI2M ^a	0.7534 \pm 0.0012	−5.13%
DEI2N-NO-TIM ^b	0.7652 \pm 0.0013	−0.71%
DEI2N-NO-IL ^c	0.7639 \pm 0.0008	−1.20%
DEI2N-NO-UHIM ^d	0.7651 \pm 0.0004	−0.75%
DEI2N-NO-USIM ^e	0.7504 \pm 0.0010	−0.7%
DEI2N	0.7671 \pm 0.0012	0.00%

^a DEI2N without User Instant Interest Modeling Layer

^b DEI2N without temporal information modeling

^c DEI2N without Interaction Layer

^d DEI2N without User Hard Interest Modeling Layer

^e DEI2N without User Soft Interest Modeling Layer

and 598 minutes respectively in training, which is the top two of the slowest models. The main reason is that DIEN+TRA uses the GRU module to model the evolution of the user interests and DMIN+TRA has to extract multiple interests from user behaviors. The proposed method DEI2N has almost equivalent time cost as DIHN and DIAN, with 576, 577 and 580 minutes respectively in training. Thus, modeling the temporal information of behaviors, the dynamic change of user instant interest when the user scrolls down, and the interactions between the trigger and target items doesn’t introduce much more time cost. The prediction time in the test set has the same tendency.

4.7 Online A/B Testing Results

Besides the offline performance comparison, we have deployed our proposed method DEI2N in the production environment to do A/B testing. The DEI2N is deployed in Alibaba.com online recommendation systems by leveraging several algorithm platforms in Alibaba Group.

Figure 4 demonstrates the flowchart of online deployment. Basically, there are two main parts in this deployment, online and offline parts. The online part is responsible for generating the final top-k items that will be exposed to end users. Specifically, The Personalization Platform (TPP) accepts real-time request which contains

Table 4: Execution time (minutes) on Alibaba.com dataset. Training Time is recorded on the training set for one epoch, and prediction time is recorded on the test set for one epoch.

Model	Training Time (m)	Prediction Time (m)
Wide&Deep	537	12
DIN	544	12
DIEN	616	15
DMIN	581	14
Wide&Deep+TR	540	12
DIN+TRA	566	12
DIEN+TRA	641	16
DMIN+TRA	598	14
DIHN	577	13
DIAN	580	13
DEI2N	576	13

the trigger item and context features such as page number. It then processes the match and rank modules in sequence. The match role is taken by Basic Engine (BE), which will generate thousands of candidate items from tens of millions of candidate item pools. All Basic Feature Service (ABFS) is utilized here to return necessary user features, such as user profile features, real-time user historical behaviors, etc. The rank role is played by Real-Time Prediction (RTP), where our proposed model DEI2N is deployed. It is responsible for calculating CTR scores for the candidate items generated by BE. Then the final top-k items will be exposed to the end user. Note that it is possible to deploy multiple models in RTP, which makes it possible to do A/B testing conveniently. For the offline part, it records the user logs and processes the logs by a big data platform MAXCOMPUTE. Algorithm One Platform (AOP) will accept the processed training samples and train the proposed model DEI2N. Once the training is finished, it will be pushed to RTP for online serving.

We have done A/B testing experiments for several weeks on five different Trigger-Induced Recommendation scenes including *Mini Detail* and *Detail Recommendation*. Considering the facts that DIAN only has a relatively small improvement over DMIN+TRA in offline experiments, usually the efforts to deploy a new model

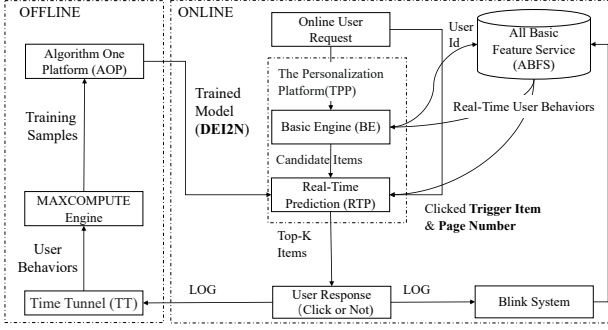


Figure 4: The flowchart of online deployment for DEI2N at Alibaba.com.

in the production environment are not trivial, and under the business growth pressure, we use the DMIN+TRA as an online baseline model which already has been deployed online. DEI2N improves the conversion rate, which is the most important metric in our recommendation scene, by 1.31%, 0.56%, 1.53%, 1.13%, and 0.89% for five scenes respectively. These improvements are statistically significant by using an unpaired t-test. It is worth mentioning that the online average response time between DEI2N and DMIN+TRA are almost the same. Thus DEI2N has been launched in all of the above TIR scenes serving millions of users every day. It demonstrates the effectiveness of DEI2N in real and scalable production environments.

5 CASE STUDY

Figure 5 shows the proportion of the recommended items with the same category as the trigger item along the page number. The proportion from DEI2N declined more slowly than the baseline when the page number increased (when the user scrolls down). Thus DEI2N maintains the dynamic change of the user's instant interest more gently, which indicates better modeling of the user's instant interest evolution.

When the user scrolls down (shown as the page number increasing in Figure 5), the proportion of the recommended items with the same category as trigger item is dropping down, which means that more diverse items are recommended.

We present two detailed user cases of DEI2N as shown by Figure 6. The left part shows a user coming into our TIR scenario by clicking a sports/gym bag. Thus the clicked bag is the trigger item. The first page shows a large proportion of sports or gym-related bags. When the user scrolls down, for example, to the fifth page, more diverse but relevant to the trigger item to some extent items are recommended. Specifically, an electronic scooter and sport muscle tapes are recommended. It makes sense that these two items are sport-related. The electronic scooter provides a light solution for going to the gym or sports arena, sport muscle tapes are helpful in muscle pain relief and joint support.

The right part of Figure 6 shows another user case by clicking a black T-shirt. On the first page, almost all of the items are T-shirts. When the user scrolls down to the tenth page, diverse items are presented. We can see that there are two long sleeve shirts among

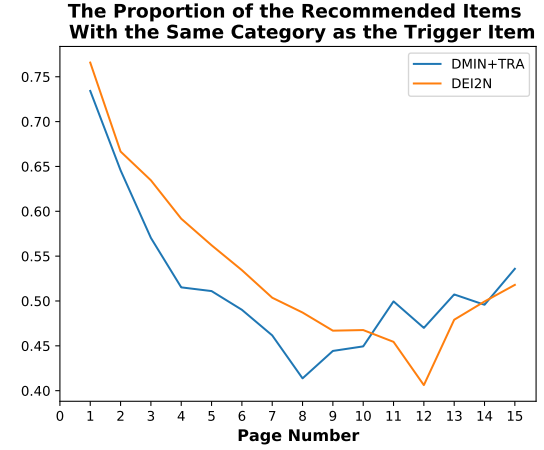


Figure 5: The proportion of the recommended items with the same category as the trigger item along the page number.

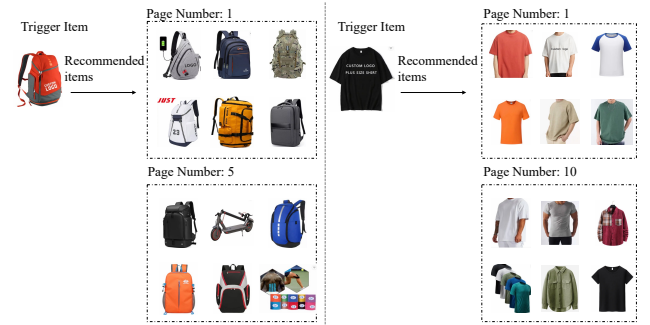


Figure 6: Two cases of DEI2N when the user scrolls down.

the T-shirts. This diversity is attributed to the elaborated modeling of dynamic change of the intensity of instant interests.

6 CONCLUSIONS

In this paper, we have proposed a novel method, Deep Evolutional Instant Interest Network (DEI2N), to model user instant interest for click-through rate prediction in TIR scenarios. DEI2N applies a User Instant Interest Modeling Layer to predict the dynamic change of the intensity of instant interest when the user scrolls down in order to extract the user's evolutionary instant interests. Temporal information is utilized in modeling layers related to user historical behaviors for better user interest representation. An Interaction Layer is used to explicitly learn better interactions between the trigger and target items. Offline experimental results show that our proposed DEI2N achieves the best performance among various state-of-the-art methods in CTR prediction tasks. DEI2N has been deployed in real-world industrial production environments, and the results of online A/B testing demonstrate the superiority over the existing baseline. Improving the conversion rate by several percents, DEI2N has been launched in five industrial TIR scenarios. In the future, we will apply graph learning and contrastive learning to model user's instant interest by considering the trigger item and user historical behaviors simultaneously, and capture better interactions between the trigger and target items.

REFERENCES

- [1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. In *arXiv preprint arXiv:1607.06450*.
- [2] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ipsir, et al. 2016. Wide & deep learning for recommender systems. In *the 1st workshop on deep learning for recommender systems*. 7–10.
- [3] 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. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*.
- [4] 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 (IJCAI)*.
- [5] Wei Guo, Rong Su, Renhao Tan, Huifeng Guo, Yingxue Zhang, Zhirong Liu, Ruiming Tang, and Xiuqiang He. 2021. Dual Graph enhanced Embedding Neural Network for CTR Prediction. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining (KDD)*.
- [6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [7] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580* (2012).
- [8] Xiang Li, Chao Wang, Bin Tong, Jiwei Tan, Xiaoyi Zeng, and Tao Zhuang. 2020. Deep time-aware item evolution network for click-through rate prediction. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management (CIKM)*.
- [9] Zekun Li, Zeyu Cui, Shu Wu, Xiaoyu Zhang, and Liang Wang. 2019. Fi-gnn: Modeling feature interactions via graph neural networks for ctr prediction. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM)*.
- [10] Zekun Li, Shu Wu, Zeyu Cui, and Xiaoyu Zhang. 2021. GraphFM: Graph Factorization Machines for Feature Interaction Modeling. *arXiv preprint arXiv:2105.11866* (2021).
- [11] Fernando B Pérez Maurera, Maurizio Ferrari Dacrema, Lorenzo Saule, Mario Scriminaci, and Paolo Cremonesi. 2020. Contentwise impressions: An industrial dataset with impressions included. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management (CIKM)*. 3093–3100.
- [12] 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 (CIKM)*.
- [13] Steffen Rendle. 2010. Factorization machines. In *2010 IEEE International Conference on Data Mining (ICDM)*. 995–1000.
- [14] Ying Shan, T Ryan Hoens, Jian Jiao, Haijing Wang, Dong Yu, and JC Mao. 2016. Deep crossing: Web-scale modeling without manually crafted combinatorial features. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*.
- [15] Qijie Shen, Hong Wen, Wanjie Tao, Jing Zhang, Fuyu Lv, Zulong Chen, and Zhao Li. 2022. Deep Interest Highlight Network for Click-Through Rate Prediction in Trigger-Induced Recommendation. In *The World Wide Web Conference (WWW)*.
- [16] Weiping Song, Chence Shi, Zhiping Xiao, Zhijian Duan, Yewen Xu, Ming Zhang, and Jian Tang. 2019. AutoInt: Automatic feature interaction learning via self-attentive neural networks. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM)*.
- [17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems (NeurIPS)*. 5998–6008.
- [18] 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.
- [19] Yaxian Xia, Yi Cao, Sihao Hu, Tong Liu, and Lingling Lu. 2023. Deep Intention-Aware Network for Click-Through Rate Prediction. In *The World Wide Web Conference (WWW)*.
- [20] Zhibo Xiao, Luwei Yang, Wen Jiang, Yi Wei, Yi Hu, and Hao Wang. 2020. Deep multi-interest network for click-through rate prediction. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management (CIKM)*.
- [21] Ruobing Xie, Rui Wang, Shaoliang Zhang, Zhihong Yang, Feng Xia, and Leyu Lin. 2021. Real-time Relevant Recommendation Suggestion. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining (WSDM)*.
- [22] 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 (AAAI)*, Vol. 33. 5941–5948.
- [23] 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 (KDD)*. 1059–1068.