FIMN: Fusing Multi-Interest and Scenario-Mutual Network for Multi-Scenario Learning

Anonymous Author(s)

ABSTRACT

Recently, multi-scenario learning has achieved flourishing development in recommendation and retrieval systems in E-commerce platforms. Current numerous models have been proposed that attempt to use a unified model to serve multiple scenarios. However, three critical challenges still remain to be carefully addressed. First, users in different scenarios explicitly have different behavior interests, which is vital for modeling but has been neglected in previous works. Second, it is intuitive that relationships between scenarios is intricate as various scenarios generally have commonalities and specific characteristics, while previous solutions neglect the complicated interrelations among scenarios. Moreover, current state-of-the-art unified models may not work well in all scenarios, since they usually face head scenario domination phenomenon due to the different data distribution. To resolve these problems, we propose a novel approach named Fusing multi-Interest and scenario-Mutual Network (FIMN), which mainly consists of four modules. FIMN performs explicitly multi-interest fusing corresponding to specific scenario and learns correlations across scenarios dynamically, meanwhile the scenario distribution discrepancy problem can be mitigated. Extensive experiments show the superiority of FIMN towards the state-of-the-art methods. FIMN has been successfully deployed in our online retrieval platform.

CCS CONCEPTS

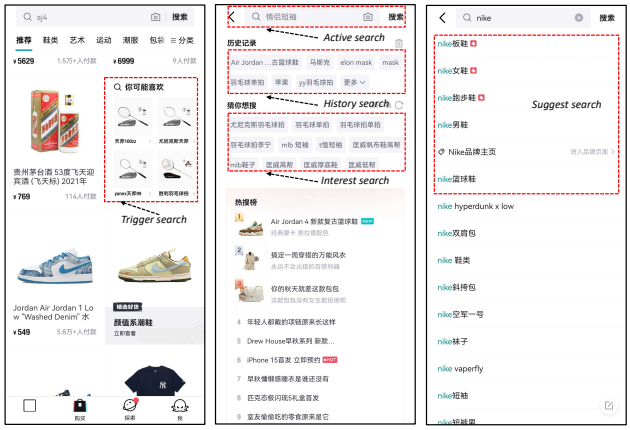
• **Information systems** → **Retrieval models and ranking**; **Recommender systems**.

KEYWORDS

Multi-scenario Learning, Click-through Rate Prediction, User Interest Modeling

1 INTRODUCTION

With rapid development of the E-commerce platforms, recommendation and retrieval systems play an increasingly critical role in boosting business revenue and improving users' online experience. Naturally, multiple shopping scenarios are rapidly developed to meet the diversified needs of users. Figure 1 lists five typical scenarios of our app: (1) *Trigger search*: this scenario displays four related queries according to users' clicked item (i.e., trigger item). (2) *Active search*: users typing queries in search bar means that they have explicit intentions. (3) *History search*: this scenario lists queries that users recently searched. (4) *Interest search*: this scenario lists queries that may be of interest to users. (5) *Suggest search*: this scenario complements and suggests terms entered by users. Clearly, these five scenarios are very different from each other, and the user intents and interests (e.g., user preference for a particular brand, price or category) behind them also vary significantly. Traditional methods [7, 9, 13, 22, 23, 35] deploy an independent model for each scenario merely based on the feedback data collected from its own. However,



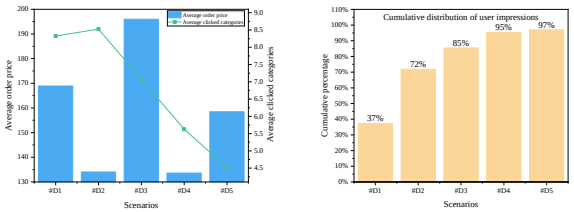
(a) Homepage (b) Search page (c) Suggest page

Figure 1: Examples of multiple scenarios in our app

separate models in scenarios with long-tail data distributions (e.g., *Trigger search* in our app usually has smaller traffic than others as users pay less attention to the entrance of it when browsing in the homepage) can not be trained sufficiently and may lead to non convergence. Also, when considering online serving deployment, multiple models usually require more complex computation and maintenance cost. Currently, state-of-the-art models generally train a unified model with merged data of all scenarios based on multi-task learning methods [11, 14, 15, 24, 27, 28, 36]. Multi-task learning is to develop a single model that outputs a couple of predictions where each is for a certain task (e.g., CTR or CVR prediction), while multi-scenario learning is to implement a single model that deals with data in various scenarios. However, unified model leads to the phenomenon that training process is dominated by the head scenarios. Furthermore, all of them neglect the significant interests discrepancy among scenarios and lack effective interrelations and correlations modeling between scenarios to further boost the performance.

In summary, three major challenges of multi-scenario learning problems are still ignored and urgently needed to be solved:

User interest discrepancy across scenarios. The interests and intentions of users vary in different scenarios. Specifically, the same interest has different intensities in different scenarios (e.g., users in *Suggest search* prefer purchasing items with lower price as they are more conservative compared with other scenarios, and they prefer browsing more categories of items as having more divergent interests in this scenario). To clarify this issue more clearly, Figure 2(a) shows the statistics of Average Order Price (AOP) and Average Clicked Categories (ACC) in different scenarios (named as #D1 to #D5), which are derived from our app from the date 21/08/2023 to



(a) Interests discrepancy in different scenarios

(b) User impressions distribution

Figure 2: Illustration of user interest and data distribution discrepancy in our APP

the date 27/08/2023. #D1 to #D5 represent *Active search*, *History search*, *Suggest search*, *Interest search* and *Trigger search* in order. The ranges of AOP and ACC are shifted with a linear transformation for data security simultaneously. From Figure 2(a), significant differences in purchasing price range and potential categories of interest can be observed, and the intra-interest intensity discrepancy in different scenarios is also demonstrated. Moreover, various user interests have different contribution degrees under a specific scenario. For example, users in *Trigger search* are more concerned about price factor when making decisions and category factor has less effect as recommended items and trigger item usually belong to the same category in *Trigger search*. However, when a user types *"nike"* in *Active search*, the price and category factor should be considered simultaneously and the ranking model should decide which factor leads from his historical behaviors. Therefore, the contributions of various interests is different in a specific scenario. So far, many works based on behavior sequence modeling [3, 10, 12, 20, 26, 35] are not able to explicitly uncover the discrepancy of user interests across scenarios.

Interrelations modeling across scenarios. Commonalities and specific characteristics coexist among various scenarios because of label space sharing and data overlapping. Effective scenario-scenario interrelation learning will largely optimize performance of each scenario as different scenarios can well benefit from each other when using appropriate interaction methods. Although current state-of-the-art methods [15, 27, 36] have made some efforts to sufficiently leverage homogeneous and heterogeneous information from different scenarios, we argue that they can not learn interrelations and correlations between scenarios adaptively.

Distribution discrepancy across scenarios. Discrepancy of scenarios' data distributions is obvious as shown in Figure 2(b). The top four scenarios take over 90% of user impressions (number of users' visits in a scenario), which leads to inferior performance in the long-tail scenarios as training process is dominated by the head scenarios. Both solutions [15, 27, 36] ignore this issue. Hence, the influence of marginal scenarios need to be enhanced during training.

Coping with the above three limitations, we propose FIMN, a novel deep neural network for multi-scenario learning. Specifically, we first devise the Scenario Enhancing Module (SEM), which is inspired by the idea of monotonic learning [10, 30] to emphasize the importance of scenarios with long-tail distributions. SEM thus enables FIMN to handle the distribution problem mentioned in the third challenge. The output representation of SEM is then delivered to the Multi-aspect Interest Extractor (MIE) and Scenario-aware Interest Fusion (SIF). In MIE, output from SEM is first crossed with each field representation of target item using Factorization Machines (FM) [21] to get a new 2-order interactive representation. A field-wise target attention (FTA) mechanism is designed to capture users' multi-aspect interests with the user behavior embeddings as key and value, the interactive representation as query. MIE explicitly outputs several user interest representations corresponding to different feature fields like price or category. Then, SIF performs explicitly interests refinement corresponding to different scenarios, and achieves fusing users' scenario-specific multi-interest. Notice that MIE and SIF can effectively address the first challenge. To the end, by composing the outputs of previous modules, Scenario Mutual Module (SMM) is introduced to well adress the second challenge, which considers correlations across scenarios and leverages mutual information dynamically. The contributions of this paper include:

- To the best of our knowledge, we make the first attempt to explicitly learn users' interest discrepancy across scenarios, which is a crucial challenge in multi-scenario learning on real-world E-commerce platforms.
- We propose several novel components in FIMN to address the aforementioned three challenges in the context of multiple scenarios, which significantly improves the effectiveness of multi-scenario learning tasks.
- Extensive offline and online experiments show that the proposed FIMN consistently and significantly outperforms all baselines. Visualization analysis and some real users' show-cases also verify the effectiveness of FIMN coping with above challenges. Also FIMN has already been deployed in our industrial system.

2 RELATED WORK

Multi-scenario learning. Recently, Multi-Task Learning (MTL)[1, 2] has been actively researched in recommendation and retrieval systems. It can learn useful information across different tasks. MoE [11, 24] proposes to select sub-expert based on the shared-bottom input. MMoE [15] adapts the MoE structure while having gating networks trained to optimize each task. PLE [27] is presented to address the seesaw phenomenon (i.e., improvement of one task often leads to performance degeneration of the other tasks). Inspired by the recent success in MTL, numerous similar works have emerged in multi-scenario learning. SAML [6] introduces scenario-aware embedding module to learn feature representations both in global and scenario-specific view. Furthermore, SAR-Net [25] proposes a unified multi-scenario architecture with some scenario-specific experts and scenario-shared experts. Lately, [33, 36] are proposed with combing multi-task and multi-scenario. Besides, several studies (e.g., PEPNet [4], MARIA [28] and M2M [32]) pay attention to employ dynamic weighting operations to model inter-scenario distinctions. M5 [34] studies how to exploit the multi-modal multi-interest multi-scenario characteristics to improve industrial matching.

3 PRELIMINARIES

Let $\mathcal{U} = \{u_1, \ldots, u_{|\mathcal{U}|}\}$ represent a set of $|\mathcal{U}|$ users, $\mathcal{I} = \{i_1, \ldots, i_{|\mathcal{I}|}\}$ be a set of $|\mathcal{I}|$ items, and $\mathcal{S} = \{s_1, \ldots, s_{|\mathcal{S}|}\}$ denote a set of $|\mathcal{S}|$ scenarios. Given each $i \in \mathcal{I}$, $u \in \mathcal{U}$, and $s \in \mathcal{S}$, then $A_i =< a_i^1, a_i^2, \ldots, a_i^{|A_i|} >$, $A_u =< a_u^1, a_u^2, \ldots, a_u^{|A_u|} >$ and $A_s =< a_s^1, a_s^2, \ldots, a_s^{|A_s|} >$ represent item field containing $|A_i|$ attributes, user field containing $|A_u|$ attributes and scenario field containing $|A_s|$ attributes respectively. Specifically, A_i contains categorical features (e.g., *item_id, category_id* and etc.) and numeric statistical features such as the number of clicks to an item in the last month. A_u and A_s are similar to A_i.

Given a query $q \in Q$ initiated by a user u, let $\mathcal{B}^u = \{b_1^u, \ldots, b_{|\mathcal{B}|}^u\}$ denote a chronological sequence of historical behaviors (e.g., clicking or purchasing) of user u and $|\mathcal{B}|$ represents the predefined maximum capacity of sequence, where $b_j^u \in \mathcal{I}$. Given a list of target items for the query q, denoted by $\mathcal{T}_q = \{T_1, \ldots, T_{|\mathcal{T}_q|}\}$ with $T_j \in \mathcal{I}$. With the above knowledge, the objective of the multi-scenario learning problem can be formally defined as to predict the probability $Pr(T_z|q, \mathcal{T}_q, \mathcal{B}^u, s)$ of the z^{th} item in $\mathcal{T}_q$ be interacted by user $u \in \mathcal{U}$ under the scenario s. Notice that the click-through rate (CTR) prediction task is considered in our work.

4 THE FIMN APPROACH

In this section, we present the proposed FIMN model in detail. Figure 3 illustrates the network architecture of our model.

4.1 Embedding Layer

As illustrated by Figure 3, there are five groups of features, i.e., scenario features, user behavior sequence, target item, user profiles and query features. After the computation of the embedding layer [17], the dense embedding vectors of the scenario features, target item, user profiles and query are denoted by $\boldsymbol{e_s} \in \mathbb{R}^{d_s}$, $\boldsymbol{e_t} \in \mathbb{R}^{d_t}$, $\boldsymbol{e_p} \in \mathbb{R}^{d_p}$ and $\boldsymbol{e_q} \in \mathbb{R}^{d}$, respectively, where d is the fixed dimensionality of each field vector and $d_s = |A_s| * d$, $d_t = |A_i| * d$, $d_p = |A_u| * d$. $|A_s|$, $|A_i|$ and $|A_u|$ are the number of fields in scenario, item and user respectively. Similarly, the behavior sequence of a user is denoted as a matrix $\boldsymbol{E_B^u} = [\ldots; \boldsymbol{e}(b_i^u); \ldots]^T \in \mathbb{R}^{|\mathcal{B}| \times d_t}$, where $|\mathcal{B}|$ is the number of user behaviors we select and each $\boldsymbol{e}(b_i^u) \in \mathbb{R}^{d_t}$ represents the embedding vector of the i^{th} item historically interacted by user u.

4.2 Scenario Enhancing Module

Existing unified models may not work well in all scenarios due to the data distribution discrepancy across scenarios. The train process is always dominated by the head scenarios as described in the third challenge. Inspired by the monotonic learning [10, 30], we propose the Scenario Enhancing Module (SEM) to emphasize the importance of long-tail scenarios. We first derive the frequency-based statistical features from the inputs of scenario field. Features include the impression probability of one scenario, the impression number of one scenario, etc., which are all normalized to a small scale. Then features are concatenated as a vector denoted as $\boldsymbol{v_s} \in \mathbb{R}^{d_{v_s}}$. Then the output $\boldsymbol{v_{se}} \in \mathbb{R}^{d_s}$ is computed as follows:

$$\boldsymbol{v_{se}} = \gamma * Sigmoid\left(Monotonic\_MLP\left(-\boldsymbol{v_s}\right)\right) \tag{1}$$

where the $Monotonic\_MLP\left(\cdot\right)$ is a MLP layer whose weights are all nonzero positive values, the output is then transformed with a sigmoid function which limits the scale to $[0, \gamma]$. γ is the scaling factor that is set to 2. By using an element-wise product operation as shown in Equation (2), the embedding vector of scenario features $\boldsymbol{e_s}$ is converted to an enhanced representation:

$$\tilde{\boldsymbol{e}}_{\boldsymbol{s}} = \boldsymbol{e_s} \otimes \boldsymbol{v_{se}} \tag{2}$$

where $\otimes$ denotes element-wise product and $\tilde{\boldsymbol{e}}_{\boldsymbol{s}} \in \mathbb{R}^{d_s}$. It is easy to understand that the value of $\boldsymbol{v_{se}}$ rises monotonously as the frequency-based statistical value declines. Under such circumstances, a smaller statistical value of scenario indicates a longer-tail scenario but derives a higher weight vector, corresponding to the purpose *make longer-tail scenario a bit more important* semantically.

4.3 Multi-aspect Interest Extractor

To explicitly extract multiple interests of a user from several aspects, we modify the widely used Multi-head target attention (MHTA) [5, 18, 19, 29] to field-wise target attention (FTA).

Specifically, we first map the final output $\tilde{\boldsymbol{e}}_{\boldsymbol{s}}$ of SEM to a new vector with the same dimensionality of each field in the target item, i,e, $\hat{\boldsymbol{e}}_{\boldsymbol{s}} = \boldsymbol{W_e}\tilde{\boldsymbol{e}}_{\boldsymbol{s}}$ with $\hat{\boldsymbol{e}}_{\boldsymbol{s}} \in \mathbb{R}^{d}$, $\boldsymbol{W_e} \in \mathbb{R}^{d \times d_s}$ is the projection matrix. Then we split the target item embedding $\boldsymbol{e_t} \in \mathbb{R}^{d_t}$ into $|A_i|$ sub-vectors. $|A_i|$ denotes the fields number of target item or interests number. The split strategy can be defined as:

$$split\left(\boldsymbol{e_t}\right) = \left[\boldsymbol{e_t^{f_1}}, \ldots, \boldsymbol{e_t^{f_{|A_i|}}}\right], \boldsymbol{e_t^{f_j}} \in \mathbb{R}^{d} \tag{3}$$

We then employ Factorization Machines (FM) to get a new 2-order interactive target representation between $\hat{\boldsymbol{e}}_{\boldsymbol{s}}$ and $\boldsymbol{e_t^{f_j}}$. The purpose of FM operation is to effectively distinguish the behaviors more relevant to the target item under the circumstances of the current scenario. The FM interaction is calculated as follows:

$$\hat{\boldsymbol{e}}_{\boldsymbol{t}}^{\boldsymbol{f_j}} = 0.5 * \left(\left(\hat{\boldsymbol{e}}_{\boldsymbol{s}} + \boldsymbol{e_t^{f_j}}\right)^2 - \left(\hat{\boldsymbol{e}}_{\boldsymbol{s}}^2 + \left(\boldsymbol{e_t^{f_j}}\right)^2\right)\right) \tag{4}$$

where $\hat{\boldsymbol{e}}_{\boldsymbol{t}}^{\boldsymbol{f_j}} \in \mathbb{R}^{d}$. Similar to the split strategy of target item, we can also split the behavior sequence matrix into $|A_i|$ sub-matrices, which is computed as:

$$split\left(\boldsymbol{E_B^u}\right) = \left[\boldsymbol{E_{B_1}^u}, \ldots, \boldsymbol{E_{B_{|A_i|}}^u}\right], \boldsymbol{E_{B_j}^u} \in \mathbb{R}^{|\mathcal{B}| \times d} \tag{5}$$

The main part of FTA is dot-product attention. The calculation of FTA is shown in Equation (6). And the calculation process of dot-product attention in detail is described in Equation (7):

$$\begin{aligned} \boldsymbol{I_u^j} &= FTA\left(\hat{\boldsymbol{e}}_{\boldsymbol{t}}^{\boldsymbol{f_j}}, \boldsymbol{E_{B_j}^u}\right) = head_j \boldsymbol{W_j^O}, \\ head_j &= Attention\left(\hat{\boldsymbol{e}}_{\boldsymbol{t}}^{\boldsymbol{f_j}} \boldsymbol{W_j^Q}, \boldsymbol{E_{B_j}^u} \boldsymbol{W_j^K}, \boldsymbol{E_{B_j}^u} \boldsymbol{W_j^V}\right) \end{aligned} \tag{6}$$

$$Attention\left(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}\right) = Softmax\left(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d_k}}\right)\boldsymbol{V} \tag{7}$$

where $\hat{\boldsymbol{e}}_{\boldsymbol{t}}^{\boldsymbol{f_j}} \in \mathbb{R}^{d}$, $\boldsymbol{E_{B_j}^u} \in \mathbb{R}^{|\mathcal{B}| \times d}$ are input embedding matrices of the j^{th} field in target item and behavior sequence respectively. $|\mathcal{B}|$ is sequence length and d is the embedding size of hidden vector for each field of items. Matrices $\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}$ represent queries, keys and values respectively. And d_q, d_k, d_v are embedding sizes for each

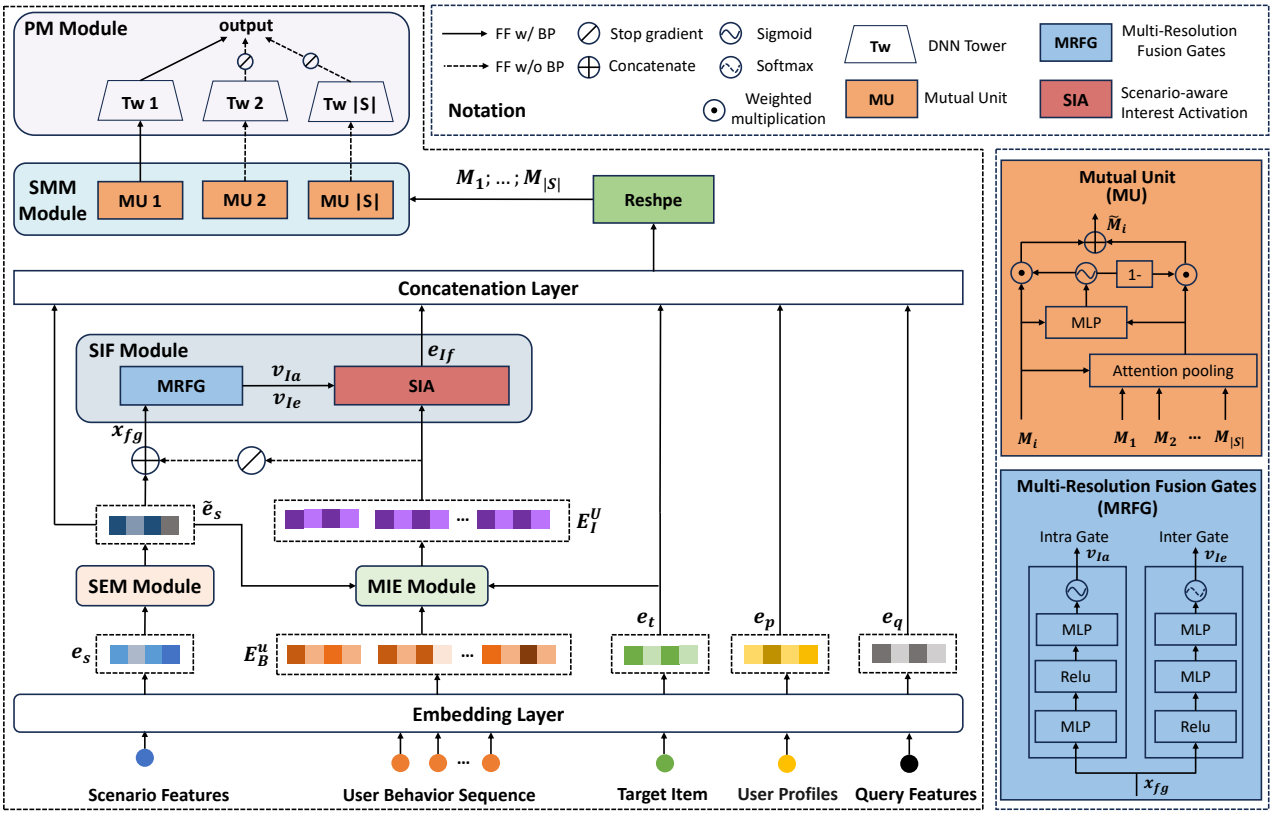


Figure 3: Overall framework of the FIMN model.

row vector of $\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}$. $\sqrt{d_k}$ is used to avoid large value of the inner product. Softmax activation function is used to convert the value of inner-product into the adding weight of the value vector $\boldsymbol{V}$. $\boldsymbol{W}_j^Q \in \mathbb{R}^{d \times d_k}, \boldsymbol{W}_j^K \in \mathbb{R}^{d \times d_k}, \boldsymbol{W}_j^V \in \mathbb{R}^{d \times d_v}, \boldsymbol{W}_j^O \in \mathbb{R}^{d_v \times d}$ are the projection matrices.

Finally, by applying the FTA to all fields, we can obtain a multi-aspect interests matrix $\boldsymbol{E}_I^U = [\boldsymbol{I}_u^1; \ldots; \boldsymbol{I}_u^{|A_i|}]^T \in \mathbb{R}^{|A_i| \times d}$, in which the j^{th} row vector denotes the interest representation corresponding to the j^{th} field. Our proposed FTA is an explicit manner capturing user interests, i.e, each field of the user interests corresponds to a specific aspect (e.g., the category field represents the category preference, the price field represents the price preference, etc.).

4.4 Scenario-aware Interest Fusion

To tackle the first challenge, the Scenario-aware Interest Fusion (SIF) component in FIMN is designed to explicitly refines interests w.r.t. specific scenario after extracting multiple interests in MIE. SIF applies the MRFG to adjust the intra-interest intensity and inter-interest contributions dynamically and refines scenario-specific interests with Scenario-aware Interest Activations (SIA).

Multi-Resolution Fusion Gates. In the MRFG module, the final output $\tilde{\boldsymbol{e}}_s$ of SEM, which denotes the enhanced scenario representation, is concatenated with the multi-aspect interests matrix $\boldsymbol{E}_I^U$. The output is:

$$\boldsymbol{x}_{fg} = \tilde{\boldsymbol{e}}_s \oplus (\oslash(\boldsymbol{E}_I^U)) \tag{8}$$

where $\boldsymbol{x}_{fg} \in \mathbb{R}^{|A_i| * d + d_s}$ and $\oplus$ indicates concatenation. $\oslash(\cdot)$ denotes stop gradient operation. Afterwards, the input of MRFG is generated with a MLP layer as follows:

$$\boldsymbol{x}'_{fg} = Relu(\boldsymbol{x}_{fg}\boldsymbol{W}_{fg} + \boldsymbol{b}) \tag{9}$$

where $\boldsymbol{x}'_{fg} \in \mathbb{R}^{d_g}$, d_g is the length of the input for MRFG. $\boldsymbol{W}_{fg} \in \mathbb{R}^{(|A_i| * d + d_s) \times d_g}$ and $\boldsymbol{b} \in \mathbb{R}^{d_g}$ are learnable weight and bias. $Relu(\cdot)$ is a non-linear activation function. Then the outputs of intra-gate $\boldsymbol{v}_{I_a}$ and inter-gate $\boldsymbol{v}_{I_e}$ can be derived from Equation (10) and Equation (11) respectively:

$$\boldsymbol{v}_{I_a} = \gamma * Sigmoid\left(\boldsymbol{x}'_{fg}\boldsymbol{W}_{I_a} + \boldsymbol{b}_{I_a}\right) \in \mathbb{R}^{|A_i| * d} \tag{10}$$

$$\boldsymbol{v}_{I_e} = Softmax\left(\boldsymbol{x}'_{fg}\boldsymbol{W}_{I_e} + \boldsymbol{b}_{I_e}\right) \in \mathbb{R}^{|A_i|} \tag{11}$$

where $\boldsymbol{W}_{I_a} \in \mathbb{R}^{d_g \times (|A_i| * d)}$ and $\boldsymbol{W}_{I_e} \in \mathbb{R}^{d_g \times |A_i|}$ are projection matrices corresponding to intra-gate and inter-gate. $\boldsymbol{b}_{I_a} \in \mathbb{R}^{|A_i| * d}$ and $\boldsymbol{b}_{I_e} \in \mathbb{R}^{|A_i|}$ are bias vectors similarly. γ is a scaling factor to further squash and double the effective signal and we set it to 2.

Scenario-aware Interest Activation. After obtaining the output of intra-gate $\boldsymbol{v}_{I_a}$, we reshape it to a matrix with the same size

as the multi-aspect interests matrix $\boldsymbol{E}_{\boldsymbol{I}}^{\boldsymbol{U}}$. The reshaping process is:

$$\boldsymbol{E}_{\boldsymbol{I_a}} = split(\boldsymbol{v}_{\boldsymbol{I_a}}) = [\boldsymbol{v}_{\boldsymbol{Ia}}^{\boldsymbol{1}}; ...; \boldsymbol{v}_{\boldsymbol{Ia}}^{|\boldsymbol{A_i}|}]^T \in \mathbb{R}^{|A_i| \times d} \tag{12}$$

where $\boldsymbol{E}_{\boldsymbol{I_a}} \in \mathbb{R}^{|A_i| \times d}$ is the reshaped output. Subsequently, Intra Interest Activation can be formulated as:

$$\hat{\boldsymbol{E}}_{\boldsymbol{I}}^{\boldsymbol{U}} = \boldsymbol{E}_{\boldsymbol{I_a}} \otimes \boldsymbol{E}_{\boldsymbol{I}}^{\boldsymbol{U}} \tag{13}$$

where $\hat{\boldsymbol{E}}_{\boldsymbol{I}}^{\boldsymbol{U}} \in \mathbb{R}^{|A_i| \times d}$ denotes the representation of Intra Interest Activation, which considers the intensities discrepancy of a specific interest in different scenarios by $\boldsymbol{v}_{\boldsymbol{I_a}}$. $\otimes$ is element-wise product.

Finally, by using a weighted sum pooling operation as shown in Equation (14) between inter-gate output $\boldsymbol{v}_{\boldsymbol{I_e}}$ and interest representations $\hat{\boldsymbol{E}}_{\boldsymbol{I}}^{\boldsymbol{U}}$ of Intra Interest Activation, the output of Inter Interest Activation is calculated in:

$$\boldsymbol{e}_{\boldsymbol{If}} = \sum_{j=1}^{|A_i|} \left((\boldsymbol{v}_{\boldsymbol{I_e}})_j \times (\hat{\boldsymbol{E}}_{\boldsymbol{I}}^{\boldsymbol{U}})_j \right) \tag{14}$$

where $(\boldsymbol{v}_{\boldsymbol{I_e}})_j$ is the j^{th} element of $\boldsymbol{v}_{\boldsymbol{I_e}}$ and $(\hat{\boldsymbol{E}}_{\boldsymbol{I}}^{\boldsymbol{U}})_j$ is the j^{th} row vector of $\hat{\boldsymbol{E}}_{\boldsymbol{I}}^{\boldsymbol{U}}$. $\boldsymbol{e}_{\boldsymbol{If}} \in \mathbb{R}^d$ represents the final output representation of the SIF module. Inter Interest Activation performs equipping discriminative contributions to different interests in a specific scenario by $\boldsymbol{v}_{\boldsymbol{I_e}}$ thus achieving scenario-aware interest fusing finally.

4.5 Scenario Mutual Module

It is apparent that different scenarios have distinguishing contributions to the current one. To further consider interrelations and correlations across scenarios in the second challenge, we design a novel Scenario Mutual Module (SMM) to sufficiently leverage the information from other scenarios, thus subsidiary contributions are infused to the current scenario to improve performance adaptively. The input $[\boldsymbol{M}_1; ...; \boldsymbol{M}_{|\mathcal{S}|}] \in \mathbb{R}^{|\mathcal{S}| \times d}$ of SMM is composed of the reshaped outputs of previous modules:

$$[\boldsymbol{M}_1; ...; \boldsymbol{M}_{|\mathcal{S}|}] = Reshape(\tilde{\boldsymbol{e}}_{\boldsymbol{s}} \oplus \boldsymbol{e}_{\boldsymbol{If}} \oplus \boldsymbol{e}_{\boldsymbol{t}} \oplus \boldsymbol{e}_{\boldsymbol{p}} \oplus \boldsymbol{e}_{\boldsymbol{q}}) \tag{15}$$

where $\boldsymbol{M}_{\boldsymbol{i}}$ denotes the i^{th} scenario representation. SMM first uses a vanilla additive target attention mechanism with the the i^{th} scenario representation, i.e., $\boldsymbol{M}_{\boldsymbol{i}}$, as the query to adaptively learn the weight for the representation vector of other scenarios. Thus different weights are assigned to each representation vector of other scenarios according to its relevance to the current one. The mutual weight of the $j^{th}, j \neq i$ scenario representation vector $\boldsymbol{M}_{\boldsymbol{j}}$, i.e., α_j, is computed as follows:

$$\begin{aligned} \alpha_j &= \frac{exp\left(a_j\right)}{\sum_{k \neq i}^{|\mathcal{S}|} exp\left(a_k\right)}, \\ a_j &= \boldsymbol{z}^T tanh\left(\boldsymbol{W}^{\boldsymbol{i}} \boldsymbol{M}_{\boldsymbol{i}} + \boldsymbol{W}^{\boldsymbol{j}} \boldsymbol{M}_{\boldsymbol{j}}\right) \end{aligned} \tag{16}$$

where $\boldsymbol{W}^{\boldsymbol{i}} \in \mathbb{R}^{d_h \times d}, \boldsymbol{W}^{\boldsymbol{j}} \in \mathbb{R}^{d_h \times d}, \boldsymbol{z} \in \mathbb{R}^{d_h}$ are learnable parameters. Then, by using a weighted sum pooling operation as shown in Equation (17), the representations of other scenarios are fused as $\hat{\boldsymbol{M}}_{\boldsymbol{i}} \in \mathbb{R}^d$.

$$\hat{\boldsymbol{M}}_{\boldsymbol{i}} = \sum_{j \neq i}^{|\mathcal{S}|} \alpha_j \boldsymbol{M}_{\boldsymbol{j}} \tag{17}$$

After that, a light dynamic gating network is employed to adaptively control the weights of $\boldsymbol{M}_{\boldsymbol{i}}$ and $\hat{\boldsymbol{M}}_{\boldsymbol{i}}$. The final output $\tilde{\boldsymbol{M}}_{\boldsymbol{i}} \in \mathbb{R}^d$ of SMM for the i^{th} scenario is:

$$\begin{aligned} \tilde{\boldsymbol{M}}_{\boldsymbol{i}} &= \theta \boldsymbol{M}_{\boldsymbol{i}} + (1 - \theta)\, \hat{\boldsymbol{M}}_{\boldsymbol{i}}, \\ \theta &= Sigmoid\left(\boldsymbol{W}^{\boldsymbol{m}} \left(\boldsymbol{M}_{\boldsymbol{i}} \oplus \hat{\boldsymbol{M}}_{\boldsymbol{i}}\right)\right) \end{aligned} \tag{18}$$

where $\boldsymbol{W}^{\boldsymbol{m}} \in \mathbb{R}^{1 \times 2d}$ is learnable parameter, and $\theta \in [0, 1]$ is a scalar.

4.6 Prediction Module

In Prediction Module, several scenario-specific DNN towers are used to map each scenario's output of SMM to the final probability, which represents the CTR prediction score of target item in the current scenario. The probability that user $u \in \mathcal{U}$ will interact with item $T_j \in \mathcal{I}$ in the i^{th} scenario is calculated as follows:

$$p_{T_j}^i = Sigmoid\left(MLP\left(\tilde{\boldsymbol{M}}_{\boldsymbol{i}}\right)\right) \tag{19}$$

Then we used the widely used cross entropy loss as the objective function.

4.7 Model Complexity

We perform model complexity analysis of FIMN to illustrate that our model meets the standards for online deployment. As the SEM and Prediction Module are composed of simple DNNs, We therefore analyze the additional computation cost introduced by MIE, SIF and SMM. The target attention operation in MIE has $O(|\mathcal{B}| \times |A_i| \times d)$ complexity, SIF takes at most $O(d_g \times (|A_i| * d + |A_s| * d))$ to finish the interest refinement and mutual operation in SMM incurs $O(|\mathcal{S}|^2 \times d)$ complexity. $|A_i|$, d_g, $|A_s|$ and $|\mathcal{S}|$ are relatively small values compared to $|\mathcal{B}|$, the complexity can be approximated as $O(|\mathcal{B}| \times d)$. The truncation length $|\mathcal{B}|$ of user behavior is fixed to a small constant, therefore, the complexity of our model is acceptable for online serving.

5 EXPERIMENTS

In this section, we conduct plenty of experiments to validate the efficacy of FIMN.

5.1 Experiment Settings

5.1.1 Datasets. we conduct our experiments over three real-world datasets as follows. Statistics of them are listed in Table 1.

- **AliCCP**[1]**.** AliCCP is a public dataset released by Taobao with prepared training and testing set, which is widely used in the relevant literature [16] for recommendation area. We split the dataset into three scenarios (abbreviated as #C1 to #C3) according to the context feature value as previous work [14]. The splitting method of the dataset follows the official guidance given in [16].
- **Alimama**[2]**.** This dataset is provided by Alimama[8], an online advertising platform in China. It is made up of 8 days of ad records from 2017. We divided the dataset into 3 scenarios based on the feature *pvalue_level_id*.

[1] https://tianchi.aliyun.com/dataset/408
[2] https://tianchi.aliyun.com/dataset/dataDetail?dataId=56

- **Ours.** It contains user search logs covering five scenarios (denoted as #D1 to #D5 for simplicity, corresponding to scenarios in the introduction section), randomly sampled at our APP from the date 20/08/2023 to the date 04/09/2023. We use logs of the last day in the dataset as testing set, and the remaining logs are used as training set.

Table 1: Statistics of three datasets. (M-million)

Dataset	Users	Items	Samples	Scenarios
AliCCP	0.4M	4.3M	1.76M	3
Alimama	0.8M	0.47M	26M	3
Ours	1.2M	3.3M	394.7M	5

5.1.2 Metrics. We adopt widely-used accuracy metric, i.e, AUC, to evaluate model performance. AUC denotes the area under the ROC curve over the testing set. It is worth mentioning that a small improvement of AUC is likely to lead to a significant increase in CTR at our app. Besides, RelaImpr metric is introduced to measure relative improvement over models following [31, 35]:

$$RelaImpr = \left(\frac{AUC(target\ model) - 0.5}{AUC(base\ model) - 0.5} - 1 \right) \times 100\% \tag{20}$$

For each method, we repeat the experiment five times and report the averaged results. The statistical significance test is conducted by using $t - test$.

5.1.3 Baselines. To demonstrate the effectiveness of our proposed model, we compare FIMN with two categories of approaches in previous works. (1) Single-task based: DeepFM [9] and DIN [35]; (2) Multi-task based: MMoE [15], PLE [27], SAR-Net [25] and HiNet [36].

5.1.4 Variants of FIMN.. To evaluate the effectiveness of each module in FIMN, FIMN is also compared with its variants in the experiment. Variants are named with "FIMN-", where "-" represents removing the component from FIMN.

5.1.5 Parameter Setting. We use Adam optimizer with batch size of 4096 and run the experiments with the learning rate of 0.001 for all comparison methods. A Gaussian distribution ($\mu = 0$ and $\sigma = 0.05$) is used to initialize the parameters in DNN. The truncation length of user behavior (if used) is 30. The scaling factor γ is 2 following [4].

5.2 Effectiveness of FIMN

We show the comparison results of FIMN and baselines on both datasets in Table 2. All the performance differences are statistically significant at 0.05 level.

- All multi-task based solutions (i.e., MMoE, PLE, SAR-Net and Hinet) consistently outperform other single-task based methods (i.e, DeepFM and DIN) on two datasets, revealing the power of multi-task learning for improving the ranking results.
- Although MMoE achieves better overall performance than the single-task based baselines, it is obvious to observe the seesaw phenomenon across scenarios in our dataset. As reported in Table 2, MMoE has inferior performance in scenario #D4 and scenario #D5 compared with the best single-task based model DIN. PLE alleviates the phenomenon by splitting experts into two groups, i.e, scenario-shared and scenario-specific partly. Moreover, this issue is slighter on AliCPP dataset as Shared Bottom and MMoE almost have similar performance at low traffic scenario #C3 compared with DIN. This is because scenarios in AliCPP dataset are relatively more similar to each other but our dataset has more different characteristics across scenarios. Note that the overall result of SAR-Net is similar to the PLE in our experiments.
- The proposed FIMN significantly outperforms all baselines across all scenarios on three datasets as shown in Table 2. Specifically, compared with the next-best solution Hinet, FIMN averagely improves the overall AUC by 2.88% RelaImpr on three datasets, which is a big progress made by an industrial recommendation or retrieval system. Also FIMN achieves satisfactory improvement in long-tail scenarios such as #D5 in our app, #C3 in AliCCP and #C3 in Alimama, i,e, FIMN remarkably increases the AUC by 7.83% RelaImpr, 2.25% RelaImpr and 3.05% RelaImpr in above three scenarios respectively, compared with the best baseline Hinet. FIMN achieves superior performance as the three challenges mentioned before are well addressed.

5.3 Ablation Study

As indicated by the Table 3, each component (i.e., SEM, MIE, SIF and SMM) makes a considerable contribution to ensure the quality of the predicted results of FIMN in all scenarios. Comparing FIMN with FIMN-MIE and FIMN-SIF, we can observe that introducing MIE and SIF can significantly improve the prediction accuracy as they explicitly model user interest discrepancy across scenarios. Besides, the performance gap between FIMN and FIMN-SMM indicates that modeling interrelations and correlations among scenarios conduces to better performance. The AUC degradation caused by deleting SEM is more significant in long-tail scenarios (i.e., #D5 in our app, #C3 in AliCCP and #C3 in Alimama) than others, which verifies that SEM can effectively handle distribution discrepancy problem.

5.4 Visualization Analysis

We conduct a visual analysis to intuitively demonstrate the effectiveness of FIMN in coping with the first two main challenges.

Analysis of Intra-gate. We first average the reshaped output of intra-gate $\boldsymbol{E}_{\boldsymbol{I}_a} \in \mathbb{R}^{|A_i| \times d}$ by row over 2000 samples and get a $|A_i|$ dimension vector $\hat{\boldsymbol{o}}_{\boldsymbol{I}_a} \in \mathbb{R}^{|A_i|}$, where $|A_i|$ is the number of interests or the item feature fields. Each value in $\hat{\boldsymbol{o}}_{\boldsymbol{I}_a}$ denotes the intensity of an interest in the current scenario. Figure 4 (a) and Figure 4 (b) illustrate the distribution of interest intensities across different scenarios on the price interest and category interest respectively. The distribution is generated with kernel density estimation based on observed data. As shown in Figure 4, the distribution deviation of price interest and category interest intensities across scenarios is very significant. Specifically, the SIF module can effectively consider the intensities discrepancy of a specific interest in different scenarios, i.e., users in different scenarios have significant discrepancy in price range selection preference and category preference.

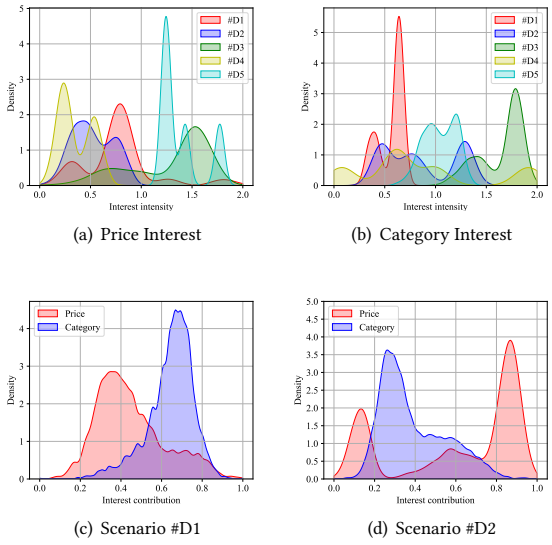
Analysis of Inter-gate. Each element in $\boldsymbol{v}_{\boldsymbol{I}_e} \in \mathbb{R}^{|A_i|}$ represents the contributions of an interest in the current scenario. And the

Table 2: Comparison of different methods on three datasets (Best values are in bold; next-best values are underlined).

Methods	AliCCP				Alimama				Ours					
	#C1	#C2	#C3	Overall	#C1	#C2	#C3	Overall	#D1	#D2	#D3	#D4	#D5	Overall
DeepFM	0.6104	0.6072	0.5916	0.6085	0.5692	0.5738	0.5691	0.5729	0.6952	0.6968	0.7400	0.6931	0.6783	0.7027
DIN	0.6123	0.6091	0.5928	0.6103	0.5721	0.5762	0.5724	0.5751	0.6968	0.6966	0.7420	0.6983	0.6841	0.7041
MMoE	0.6216	0.6157	0.5929	0.6169	0.5784	0.5810	0.5779	0.5793	0.6992	0.6994	0.7452	0.6973	0.6811	0.7065
PLE	0.6214	0.6161	0.5937	0.6172	0.5786	0.5819	0.5798	0.5806	0.7004	0.7001	0.7463	0.6987	0.6839	0.7073
SAR-Net	0.6215	0.6163	0.5979	0.6171	0.5785	0.5816	0.5799	0.5807	0.7002	0.7003	0.7466	0.6985	0.6820	0.7075
Hinet	0.6223	0.6174	0.5978	0.6194	0.5806	0.5834	0.5821	0.5829	0.7010	0.7030	0.7409	0.7042	0.6909	0.7088
FIMN	**0.6246**	**0.6229**	**0.6001**	**0.6231**	**0.5829**	**0.5857**	**0.5846**	**0.5852**	**0.7061**	**0.7069**	**0.7538**	**0.7104**	**0.7059**	**0.7146**
RelaImpr	1.88%	4.68%	2.25%	3.10%	2.85%	2.76%	3.05%	2.77%	2.55%	1.90%	2.91%	3.02%	7.83%	2.78%

Table 3: Comparison of different FIMN variants on three datasets (Best values are in bold; next-best values are underlined).

Models	AliCCP				Alimama				Ours					
	#C1	#C2	#C3	Overall	#C1	#C2	#C3	Overall	#D1	#D2	#D3	#D4	#D5	Overall
FIMN-SEM	0.6238	0.6222	0.5974	0.6225	0.5816	0.5846	0.5838	0.5841	0.7053	0.7062	0.7531	0.7062	0.6979	0.7138
FIMN-MIE	0.6224	0.6215	0.5983	0.6216	0.5796	0.5826	0.5818	0.5822	0.7029	0.7033	0.7501	0.7084	0.7031	0.7115
FIMN-SIF	0.6213	0.6201	0.5976	0.6208	0.5785	0.5812	0.5803	0.5807	0.7015	0.7024	0.7482	0.7063	0.7020	0.7103
FIMN-SMM	0.6231	0.6217	0.5986	0.6220	0.5807	0.5834	0.5841	0.5833	0.7034	0.7046	0.7523	0.7078	0.7039	0.7125
FIMN	**0.6246**	**0.6229**	**0.6001**	**0.6231**	**0.5829**	**0.5857**	**0.5846**	**0.5852**	**0.7061**	**0.7069**	**0.7538**	**0.7104**	**0.7059**	**0.7146**



(a) Price Interest (b) Category Interest

(c) Scenario #D1 (d) Scenario #D2

Figure 4: Distribution of interests intensities and contributions among scenarios

range of them is (0, 1) due to the softmax transformation of intergate in Equation (11). Figure 4 (c) and Figure 4 (d) illustrate the distribution of price and category interest contributions in scenario #D1 and #D2 respectively. We can see that different interests have diverse contributions in scenario #D1 or #D2, which corresponds to the fact that price or category factor has different effect when users make decisions under a specific scenario. For example, users in *Trigger search* are more concerned about price but may consider both price and category when in *Active search*.

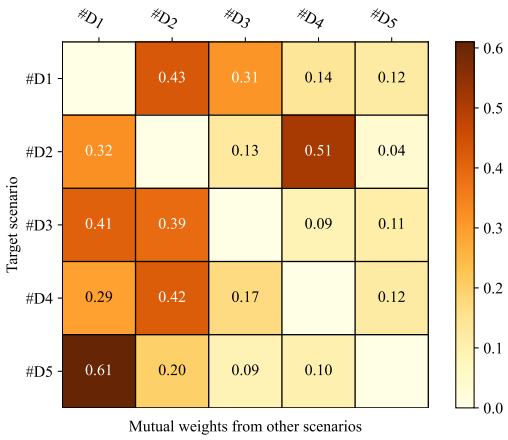


Figure 5: Visualization of mutual weights among different scenarios

Analysis of Mutual weights. The heat map in Figure 5 shows the mutual weights of scenario information output by additive attention mechanism in Equation (16). We found that information contributions have discrepancy among scenarios. For example, to assist in better learning of target scenario #D2, information from #D1 and #D4 have higher weights compared with other scenarios. This phenomenon obviously proves SMM's ability to learn interrelations and correlations between scenarios.

6 ONLINE A/B TEST

We conduct fair online A/B test based on the real traffic in our app. To be specific, we deploy FIMN and comparison methods in online serving system and execute inference tasks on daily requests of users. We take the averaged results from 02/01/2024 to 08/01/2024, FIMN obtains 3.52% overall CTR gains over Hinet (the best baseline in our experiment) online. 3.52% is a significant increase in a mature industrial system. Online test results compared with Hinet of each scenario are illustrated in Figure 6. It shows that FIMN has significant yet consistent improvement across the five scenarios. It is worth noting that FIMN has achieved more significant CTR revenue in #D5 (long tail scenario, *Trigger search*), further proving the ability of the proposed model to address the challenge of data distribution discrepancy. FIMN has already been deployed in industrial system serving for hundreds of millions of people.

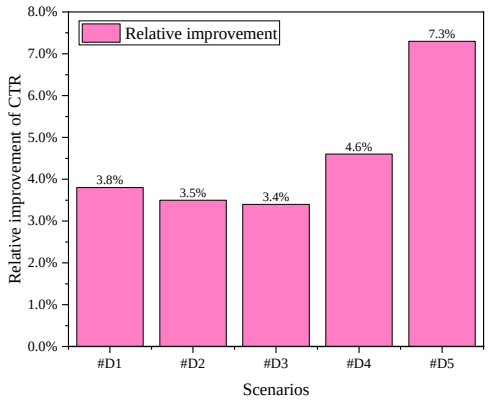


Figure 6: Online A/B test results by each scenario

7 DEPLOYMENT AND EFFICIENCY

7.0.1 Deployment of FIMN.. All experiments are implemented in PAI (Platform of Artificial Intelligence) [3]. Specifically, 6 parameter servers and 30 workers are used in this architecture. Every parameter server owns 3 CPU cores with 8GB RAM, being responsible for storing part of the parameters. Each worker has 6 CPU cores and 16GB memory, which fetches a portion of training samples, computes and delivers the computed results (e.g., gradients of parameters) to parameter servers. Then, the trained FIMN model is uploaded to the real-time prediction (RTP) center to serve online traffic and FIMN performs daily parameter updating process based on the latest collected training data.

7.0.2 Efficiency evaluation. Following the deployment strategy of FIMN, we deploy each comparison method at our app. The training time and online inference time of different methods are shown in Table 4. In Table 4 we can observe that there is significant difference in training and inference time between DIN and DeepFM as target attention mechanism used in DIN consumes more time when calculating the similarity of Q and K. Another observation is that multi-task based methods (i.e., MMoE, PLE, SAR-Net and Hinet) consistently get longer training and inference time compared with other methods, due to more parameters are involved in them. Note that the online inference time of all methods is within 25 milliseconds, which makes them meet the requirements of deployment in industrial applications.

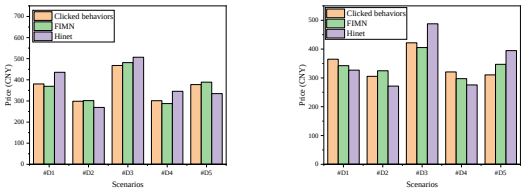
[3]https://help.aliyun.com/product/30347.html

Table 4: Evaluation of efficiency of different methods on our dataset (h-hour; m-minute; s-second; ms-millisecond).

Methods	Training time	Inference time
DeepFM	2h 03m 29s	11ms
DIN	2h 13m 12s	17ms
MMoE	2h 52m 27s	23ms
PLE	3h 2m 25s	23ms
SAR-Net	3h 11m 37s	25ms
Hinet	3h 06m 19s	22ms
FIMN	3h 09m 44s	23ms

8 CASE STUDY

After the successful online deployment of FIMN, We can collect enough user feedback logs to help better explaining the effectiveness of the proposed FIMN model in coping with the beforementioned challenges, representative cases for two real-world users are illustrated in Figure 7 (a) and Figure 7 (b) respectively. Three histograms represent the average price of the items that users clicked in the past seven days by scenarios, the average price of the items that FIMN and Hinet recommend in different scenarios, respectively. It is obvious that the average price of items recommended by FIMN is closer to users' historical preferences than Hinet in all scenarios of the two users, which proves that FIMN can more effectively capture user interest discrepancy across scenarios.



(a) User1 (b) User2

Figure 7: Case study of FIMN for real-world users

9 CONCLUSION

We propose a novel FIMN model to address multi-scenario learning problems. Extensive offline and online experiments demonstrate the superiority and effectiveness of FIMN in tackling three long-lasting challenges of multi-scenario modeling.

REFERENCES

[1] Andreas Argyriou, Massimiliano Pontil, Yiming Ying, and Charles Micchelli. 2007. A spectral regularization framework for multi-task structure learning. *Advances in neural information processing systems* 20 (2007).

[2] Rich Caruana. 1997. Multitask learning. *Machine learning* 28 (1997), 41–75.

[3] Yukuo Cen, Jianwei Zhang, Xu Zou, Chang Zhou, Hongxia Yang, and Jie Tang. 2020. Controllable multi-interest framework for recommendation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2942–2951.

[4] Jianxin Chang, Chenbin Zhang, Yiqun Hui, Dewei Leng, Yanan Niu, Yang Song, and Kun Gai. 2023. Pepnet: Parameter and embedding personalized network for infusing with personalized prior information. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 3795–3804.

[5] Qiwei Chen, Changhua Pei, Shanshan Lv, Chao Li, Junfeng Ge, and Wenwu Ou. 2021. End-to-end user behavior retrieval in click-through rateprediction model. *arXiv preprint arXiv:2108.04468* (2021).

[6] Yuting Chen, Yanshi Wang, Yabo Ni, An-Xiang Zeng, and Lanfen Lin. 2020. Scenario-aware and Mutual-based approach for Multi-scenario Recommendation in E-Commerce. In *2020 International Conference on Data Mining Workshops (ICDMW)*. IEEE, 127–135.

[7] 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.

[8] Kun Gai, Xiaoqiang Zhu, Han Li, Kai Liu, and Zhe Wang. 2017. Learning piece-wise linear models from large scale data for ad click prediction. *arXiv preprint arXiv:1704.05194* (2017).

[9] 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).

[10] Malay Haldar, Prashant Ramanathan, Tyler Sax, Mustafa Abdool, Lanbo Zhang, Aamir Mansawala, Shulin Yang, Bradley Turnbull, and Junshuo Liao. 2020. Improving deep learning for airbnb search. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2822–2830.

[11] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. 1991. Adaptive mixtures of local experts. *Neural computation* 3, 1 (1991), 79–87.

[12] Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*. IEEE, 197–206.

[13] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009), 30–37.

[14] Pengcheng Li, Runze Li, Qing Da, An-Xiang Zeng, and Lijun Zhang. 2020. Improving multi-scenario learning to rank in e-commerce by exploiting task relationships in the label space. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2605–2612.

[15] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H Chi. 2018. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 1930–1939.

[16] Xiao Ma, Liqin Zhao, Guan Huang, Zhi Wang, Zelin Hu, Xiaoqiang Zhu, and Kun Gai. 2018. Entire space multi-task model: An effective approach for estimating post-click conversion rate. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 1137–1140.

[17] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems* 26 (2013).

[18] 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.

[19] 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.

[20] Ahmed Rashed, Shereen Elsayed, and Lars Schmidt-Thieme. 2022. Context and attribute-aware sequential recommendation via cross-attention. In *Proceedings of the 16th ACM Conference on Recommender Systems*. 71–80.

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

[22] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. 2001. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*. 285–295.

[23] J Ben Schafer, Dan Frankowski, Jon Herlocker, and Shilad Sen. 2007. Collaborative filtering recommender systems. In *The adaptive web: methods and strategies of web personalization*. Springer, 291–324.

[24] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538* (2017).

[25] Qijie Shen, Wanjie Tao, Jing Zhang, Hong Wen, Zulong Chen, and Quan Lu. 2021. Sar-net: a scenario-aware ranking network for personalized fair recommendation in hundreds of travel scenarios. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 4094–4103.

[26] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*. 1441–1450.

[27] Hongyan Tang, Junning Liu, Ming Zhao, and Xudong Gong. 2020. Progressive layered extraction (ple): A novel multi-task learning (mtl) model for personalized recommendations. In *Proceedings of the 14th ACM Conference on Recommender Systems*. 269–278.

[28] Yu Tian, Bofang Li, Si Chen, Xubin Li, Hongbo Deng, Jian Xu, Bo Zheng, Qian Wang, and Chenliang Li. 2023. Multi-Scenario Ranking with Adaptive Feature Learning. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 517–526.

[29] 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).

[30] Jia Xu, Wanjie Tao, Zulong Chen, Jin Huang, Huihui Liu, Hong Wen, Shenghua Ni, Qun Dai, and Yu Gu. 2023. PlanRanker: Towards Personalized Ranking of Train Transfer Plans. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 5315–5326.

[31] Ling Yan, Wu-Jun Li, Gui-Rong Xue, and Dingyi Han. 2014. Coupled group lasso for web-scale ctr prediction in display advertising. In *International conference on machine learning*. PMLR, 802–810.

[32] Qianqian Zhang, Xinru Liao, Quan Liu, Jian Xu, and Bo Zheng. 2022. Leaving no one behind: A multi-scenario multi-task meta learning approach for advertiser modeling. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*. 1368–1376.

[33] Yifei Zhang, Hua Hua, Hui Guo, Shuangyang Wang, Chongyu Zhong, and Shijie Zhang. 2023. 3MN: Three Meta Networks for Multi-Scenario and Multi-Task Learning in Online Advertising Recommender Systems. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 4945–4951.

[34] Pengyu Zhao, Xin Gao, Chunxu Xu, and Liang Chen. 2023. M5: Multi-Modal Multi-Interest Multi-Scenario Matching for Over-the-Top Recommendation. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 5650–5659.

[35] 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.

[36] Jie Zhou, Xianshuai Cao, Wenhao Li, Lin Bo, Kun Zhang, Chuan Luo, and Qian Yu. 2023. HiNet: Novel Multi-Scenario & Multi-Task Learning with Hierarchical Information Extraction. In *2023 IEEE 39th International Conference on Data Engineering (ICDE)*. IEEE, 2969–2975.

SUPPLEMENTARY MATERIALS

A IMPACT OF PARAMETERS

Appendix A discusses the impact of different parameters on the performance of the proposed FIMN.

A.1 Impact of Scaling Factor

Figure 8 displays the impact of the scaling factor γ set for SEM and intra-gate in MRFG. As shown in the figure, when the value of factor equals to 2, FIMN achieves its best performance in terms of AUC, while increasing value beyond 2 reduces its performance. Hence, we set the scaling factor in FIMN and its variants to 2 in all experiments.

A.2 Impact of Different Lengths of Behaviors.

Figure 9 illustrates the impact of different lengths of user behaviors. Obviously, setting $|\mathcal{B}|$ to a larger value can improve AUC performance, since more historical useful information is extracted to improve the effect of the model. Figure 9 also shows the impact of different lengths on the training time. Longer user behavior sequences require more computation when calculating target attention. We observe that $|\mathcal{B}| = 30$ is a tradeoff point which provides a relatively significant boost in accuracy and less consumption in training time (minutes). Hence, $|\mathcal{B}|$ is set to 30 in FIMN and its relevant variants in all experiments.

A.3 Impact of Different Dimensions of Vectors.

Figure 10 illustrates the impact of different dimensions d of each field vector. As demonstrated by the figure, when the numerical value of the dimension increases, AUC also shows a trend of improvement. This is mainly due to the use of more parameters and deeper information. However, more parameters means longer training time. Increasing d from 8 to 32 improves the performance of FIMN, while increasing d beyond 32 bringing no remarkable benefit and significantly consuming more training time. This indicates that 32 is a reasonable number of dimension for FIMN. Hence, $d = 32$ is used in all experiments for FIMN and its related variants.

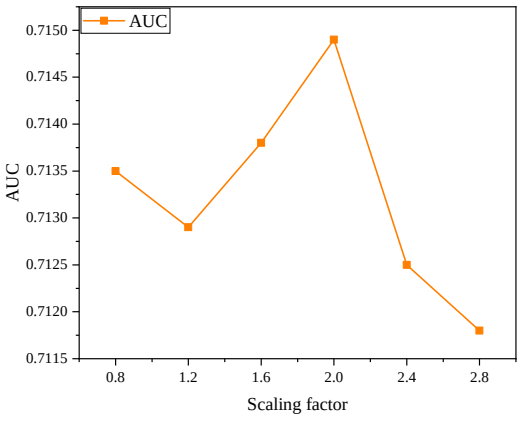


Figure 8: Varying the number of scaling factor.

B FREQUENTLY-USED OF NOTATIONS

Table 5 summarizes the frequently-used notations and their descriptions.

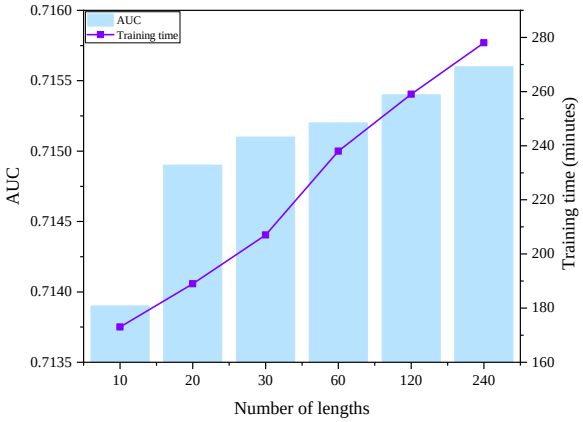


Figure 9: Varying the lengths of behaviors.

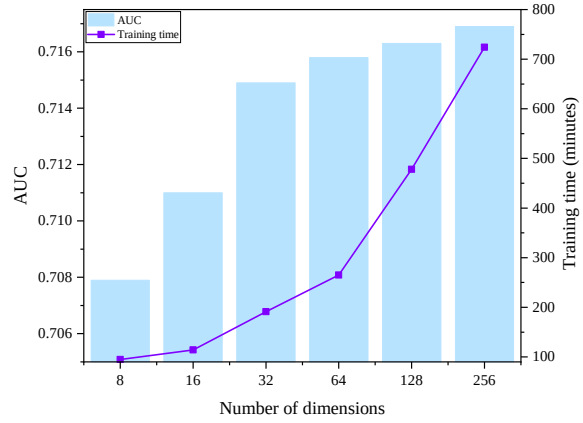


Figure 10: Varying the dimensions of field vectors.

Table 5: Description of Frequently-used Notations

Notations	Descriptions
$\mathcal{U} = \{u_1, \ldots, u_{\lvert\mathcal{U}\rvert}\}$	A set of $\lvert\mathcal{U}\rvert$ users. u_i is the i^{th} user in the set.
$\mathcal{I} = \{i_1, \ldots, \quad i_{\lvert\mathcal{I}\rvert}\}$	A set of $\lvert\mathcal{I}\rvert$ items. i_n is the n^{th} item in the set.
$\mathcal{S} = \{s_1, \ldots, s_{\lvert\mathcal{S}\rvert}\}$	A set of $\lvert\mathcal{S}\rvert$ scenarios. s_k is the k^{th} scenario in the set.
$A_i =< a_i^1, a_i^2, \ldots, a_i^{\lvert A_i\rvert} >$	Item field containing $\lvert A_i\rvert$ attributes w.r.t. item $i \in \mathcal{I}$.
$A_u =< a_u^1, a_u^2, \ldots, a_u^{\lvert A_u\rvert} >$	User field containing $\lvert A_u\rvert$ attributes w.r.t. user $u \in \mathcal{U}$.
$A_s =< a_s^1, a_s^2, \ldots, a_s^{\lvert A_s\rvert} >$	Scenario field containing $\lvert A_s\rvert$ attributes w.r.t. scenario $s \in \mathcal{S}$.
$q \in Q$	A query initiated by a user u.
$\mathcal{B}^u = \{b_1^u, \ldots, b_{\lvert\mathcal{B}\rvert}^u\}$	A sequence of historical behaviors (e.g., clicking or purchasing) of user u.
$\mathcal{T}_q = \{T_1, \ldots, T_{\lvert\mathcal{T}_q\rvert}\}, T_j \in \mathcal{I}$	A list of target items for the query q.
$\boldsymbol{e_s}, \boldsymbol{e_t}, \boldsymbol{e_q}$	Embedding vectors of the scenario features, target item, user profiles and query.
$\boldsymbol{E_B^u} = [\ldots; \boldsymbol{e}(b_i^u); \ldots]^T$	Embedding matrix of the user behavior sequence.
$\tilde{\boldsymbol{e}}_s$	An enhanced representation of scenario s. It is the output of the module SEM.
$\hat{\boldsymbol{e}}_s = \boldsymbol{W_e}\tilde{\boldsymbol{e}}_s$	A new vector w.r.t. scenario s with the same dimensionality of each field in the target item.
$\boldsymbol{E_I^U} = [\boldsymbol{I_u^1}; \ldots; \boldsymbol{I_u^{\lvert A_i\rvert}}]^T$	A multi-aspect interests matrix. Each row vector denotes the specific interest representation.
$\boldsymbol{x_{fg}}$	A concatenation vector of the enhanced scenario representation and multi-aspect interests matrix.
$\boldsymbol{x'_{fg}}$	The input of MRFG. It is generated from $\boldsymbol{x_{fg}}$ with MLP.
$\boldsymbol{v_{I_a}}, \boldsymbol{v_{I_e}}$	The outputs of intra-gate and inter-gate in module MRFG.
$\boldsymbol{E_{I_a}}$	A reshaped matrix of $\boldsymbol{v_{I_a}}$ with the same dimensionality of $\boldsymbol{E_I^U}$.
$\hat{\boldsymbol{E}}_I^U$	The representation of Intra Interest Activation considering the intensities discrepancy of interests.
$\boldsymbol{e_{If}}$	The output of Inter Interest Activation. It is a weighted sum pooling vector of $\boldsymbol{v_{I_e}}$ and $\hat{\boldsymbol{E}}_I^U$.
$[\boldsymbol{M_1}; \ldots; \boldsymbol{M_{\lvert\mathcal{S}\rvert}}]$	The input of SMM. It is composed of the reshaped outputs of previous modules. $\boldsymbol{M_i}$ denotes the i^{th} scenario representation.
$\tilde{\boldsymbol{M}}_i$	The final output of SMM module for the i^{th} scenario.