Applied Intelligence (2022) 52:8651–8663
https://doi.org/10.1007/s10489-021-02931-0

Hierarchical attention and feature projection for click-through rate prediction

Jinjin Zhang[1] · Chengliang Zhong[2] · Shouxiang Fan[1] · Xiaodong Mu[1] · Zhen Ni[3]

Accepted: 16 October 2021 / Published online: 2 November 2021

Abstract

Click-through rate (CTR) prediction plays an important role in many industrial applications, feature engineering directly influences CTR prediction performance because features are normally the multi-field type. However, the existing CTR prediction techniques either neglect the importance of each feature or regard the feature interactions equally for feature learning. In addition, using an inner product or a Hadamard product is too simple to effectively model the feature interactions. These limitations lead to suboptimal performances of existing models. In this paper, we propose a framework called Hierarchical Attention and Feature Projection neural network (HAFP) for CTR prediction, which enables the automatically learning of more representative and efficient feature representation in an end-to-end manner. Towards this end, we employ a feature learning layer with a hierarchical attention mechanism to jointly extract more generalized and dominant features and feature interactions. In addition, a projective bilinear function is designed in meaningful second-order interaction encoder to effectively learn more fine-grained and comprehensive second-order feature interactions. Taking advantages of the hierarchical attention mechanism and the projective bilinear function, our proposed model can not only model feature learning in a flexible fashion, but also provide an interpretable capability of the prediction results. Experimental results on two real-world datasets demonstrate that HAFP outperforms the state-of-the-art in terms of Logloss and AUC for CTR prediction baselines. Further analysis verifies the importance of the proposed hierarchical attention mechanism and the projective bilinear function for modelling the feature representation, showing the rationality and effectiveness of HAFP.

Keywords CTR prediction · Feature interactions · Feature representation · Hierarchical attention mechanism · Projective bilinear function

1 Introduction

Click-through rate (CTR) prediction is a well-known recommendation task that aims to predict the probability of a user clicking on recommended items and ads [26, 30]. Since CTR prediction directly influences the revenues of advertising platforms and the satisfaction of users, it has become an actively investigated topic in both industry and academic communities for recommender system research [5, 12]. Thus, designing an effective and efficient model for improving the accuracy of CTR prediction has received much attention.

The key challenge in CTR prediction is how to effectively model feature engineering. In traditional methods, Logistic Regression (LR) [1] and Factorization Machine (FM) [24] are two popular models for feature learning. LR is a linear model and encodes features through a linear combination. FM utilizes factorized parameters to model second-order feature interactions for feature learning. There are many variants of FM for improving the performance of CTR prediction [12, 22, 34]. However, the limitation of these methods is that they cannot obtain high-order feature interactions. Recently, some deep learning based models are proposed for CTR prediction as these methods show great improvements over traditional methods on recommender systems. Factorization machine supported Neural Network (FNN) [40] and Factorization-Machine based neural network (deepFM) [6] combine FM and multilayer perception

✉ Jinjin Zhang
qqzhang_work@163.com

[1] Xi'an High-Tech Research Institution, Xi'an, 710038, Shannxi, China

[2] Tsinghua University, Beijing, 100084, China

[3] Shaanxi Chang Ling Special Equipment Co.Ltd, Baoji, 721006, Shannxi, China

(MLP) through different aggregations. Additionally, there are some researches focusing on modelling different feature interactions. Attentional Factorization Machine (AFM) [34] introduces the attention mechanism [29] into second-order feature interactions to automatically learn weights. Feature importance and bilinear feature interaction network (FiBiNET) [10] dynamically learns the feature importance and fine-grained second-order feature interactions. These methods achieve remarkable improvements and verify that low- (first-order and second-order) and high-order feature interactions are important for feature learning.

Several works devote to automatic feature engineering with deep neural networks (DNNs) and discover that the quality of low- and high-order feature interactions directly influences the performance of CTR prediction. However, there is not any work jointly focusing on capturing informative feature interactions from both low- and high-order. First, it is widely accepted that different features have different importances for a target task. For example, feature occupation is more important than the feature age when predicting a user's income. Second, not all pair-wise feature interactions are equally useful for prediction. For example, the interaction of feature occupation and home address is more useful than the interaction of feature age and gender when predicting a user's income. Third, high-order feature interactions are built from low-order feature interactions, and the generated feature space would be huge and require extremely heavy computation. Different layer feature interactions represent different semantic information for feature learning, and the less useful interactions should be assigned lower weights since their contributions are limited, which can significantly reduce the computation. Furthermore, the data involved in CTR prediction are typically categorical and very sparse [28], and existing methods usually use an inner product or a Hadamard product for modelling second-order feature interactions. However, it may limit the feature learning and hurt the overall performance since the inner product and Hadamard product are too simple to effectively calculate the interactions of feature interactions in sparse datasets. Although FiBiNET [10] designs three kinds of bilinear function for second-order feature interactions and gains some improvement, it merely considers the projection from the i-th feature to the j-th feature while ignoring the projection from the j-th feature to the i-th feature when building pair-wise feature interactions. We argue that the incomplete projection has bias and crucially limits the performance.

To address the above mentioned problems, we propose an effective framework called Hierarchical Attention and Feature Projection neural network (HAFP) to fully exploit relevant information from different orders of feature interactions and extract more fine-grained second-order feature interactions from the comprehensive projection. Specifically, inspired by the knowledge that the importance of different features differs greatly for the final task, HAFP first retrieves the salient features through the designed attentive global-local contexts module. Next, HAFP computes the weighted score for each second-order feature interaction according to its contribution, which is the second-level attention. Finally, high-order feature interaction of each layer is selectively aggregated for feature learning and a more meaningful and informative feature representation can be learnt. Furthermore, as the quality of second-order feature interactions directly influences the performance of high-order feature interactions, a projective bilinear function is designed to learn more fine-grained feature interactions.

Comparing with existing methods, the proposed HAFP is able to not only encode more informative features and feature interactions but also capture more comprehensive interaction information, thus facilitating the feature representation which is built from our model owns meaningful information and has good explanations. We implement experiments on two public datasets, Criteo and Avazu. The results indicate that HAFP is able to achieve an accurate CTR prediction and outperforms the state-of-the-art methods for CTR prediction on the two datasets in terms of AUC and Logloss. This work provides a new feasible way to improve the accuracy of CTR prediction. Besides, our proposed HAFP can be directly used in practice and helps to increase the revenue of advertising platforms.

The contributions of our HAFP model can be summarized as follows.

- The proposed hierarchical attention mechanism fully exploits relevant contexts for the feature learning, and the weights of new features can be trained in the same way. It improves the extensibility of our model and consistency with practice. To the best of our knowledge, we are the first to jointly capture relevant information from both low- and high-order feature interactions for CTR prediction in an end-to-end manner.
- Inspired by the success of bilinear-interaction layer in [10], we introduce a projective bilinear function which employs an inner product to form a co-projection matrix and a Hadamard product to generate the interaction embedding. It dynamically learns feature interactions in a more fine-grained way.
- An attentive global-local contexts module is designed to adaptively select meaningful features, which can simultaneously emphasize common information that distributes more globally and highlight characterized information that distributes more locally.
- We conduct comprehensive experiments on the two public datasets. The results show that our model can

improve the CTR prediction performance with 0.3% and 0.2% in terms of AUC respectively, compared to the best performance baseline.

The rest of this paper is structured as follows. Section 2 discusses some related work of CTR prediction. Section 3 presents the HAFP model in detail. We conduct comprehensive experiments and present the experimental setups with the corresponding results in Section 4. Finally, we conclude the paper and point out some future work in Section 5.

2 Related work

CTR prediction is usually studied as a binary classification task, and accurate feature engineering is helpful for improving the performance of CTR prediction task [15, 32]. In order to improve the prediction performance, some models pay attention to feature interactions[17, 36, 37], and the other models [8, 27, 31, 35, 38] argue that behavior sequences can benefit model learning and are useful in performance. Since our work focuses on modelling information from features, we briefly review traditional methods and deep learning based methods which are related to feature interactions.

2.1 Traditional methods

In traditional methods, LR is the foundation of many popular models and it is widely used in both industrial and academic areas for CTR prediction, in which the weights corresponding to the features are considered as their importance degree or their influence on the click rate. However, they belong to the linear model and lack the ability to build sophisticated feature interactions. Additionally, FM [24] is another well-known model for CTR prediction. It projects sparse features into low-dimensional dense vectors and builds second-order feature interactions by using an inner product on the dense vectors. Therefore, FM based methods [12, 34] can deal with the problem of data sparsity better than LR based methods. Afterwards, some variants of FM are proposed for improving the performance of the final task. Field-aware Factorization Machine (FFM) [12] introduces field information into the FM model. AFM [34] extends the FFM by adding an attention mechanism to capture the feature interaction importance, and it owns good interpretability. However, these traditional methods only have the capability to model low-order feature interactions, and they have no power to model high-order feature interactions. In addition, a linear combination of feature interactions limits their performances for the final task.

2.2 Deep learning based methods

In deep learning based methods, DNNs are introduced into CTR prediction since they can effectively capture high-order feature interactions for better performances [20, 25]. FNN uses pre-trained embedding from FM and then models high-order feature interactions via MLP. Product-based neural network (PNN) [23] takes an inner product and an outer product for feature embedding instead of FM. Compared to the previous methods which attribute to shallow structures, FNN and PNN obtain better performance. However, the limitation of FNN and PNN is that they focus less on low-order feature interactions, which is insufficient to make accurate feature learning. To jointly encode low- and high-order feature interactions, Wide&Deep [4] and Deep&Cross [33] integrate a wide/cross part and a deep part to individually build low- and high-order feature interactions. DeepFM [6] introduces FM into the wide part of Wide&Deep model and introduces raw features to the deep part. Deep Field Relation Neural Network (DFRNN) [42] takes a 3-dimensional relation tensor to model the feature interactions. The performances of these works verify that jointly modelling low-order and high-order feature interactions is beneficial for extracting comprehensive and representative information. However, such works cannot learn effective interactions since the contributions of different feature interactions to the CTR prediction result may be different. To alleviate this problem, Interpretable CTR prediction model with Hierarchical Attention Mechanism (InterHAt) [16] considers the interaction order and builds second-order feature interactions through a multi-head self-attention based transformer on raw feature embeddings. To further automatically learn indispensable feature interactions, a High-order Attentive Factorization Machine (HoAFM) [28] method introduces a bit-wise attention mechanism to determine the different importance of low- and high-order feature interactions. A Multi-order interactive features aware Factorization Machine (MoFM) [37] approach integrates three different types of prediction models to effectively capture low-order and high-order interactive features. Attentive Capsule Network (ACN) [13] uses transformers to automatically learn the meaningful feature interaction. Cai et al. [2] propose an effective CTR prediction method called CAN, which explicitly exploits the benefits of attention mechanism and DNNs in modelling low-order and high-order feature interactions. Besides, since the residual module is verified to have the capability to retrieve powerful and discriminant representations [21], the research [18] introduces the residual network into the layer of learning high-order feature interactions. It forms a structure of ResNet-CTR which can explore complex feature interactions at different layers. Compared to the previous works, these approaches that

consider the importance of each feature interaction outperform the previous methods. However, they still extract the feature representation from the raw data directly and ignore the impact of different features for CTR prediction. Thus, there is room for improvements because unnecessary features are modelled without considering their importance. Recently, since not all features are equally useful for modelling feature interactions and a better feature representation makes the feature interactions easier, FiBiNET [10] constructs the embedding vectors of multi-field features and feature interactions through the SENET layer and bilinear function. In addition, Yang et al. [39] focus on improving the feature representation and propose an embedding method called operation-aware embedding. It can learn different representations for each feature when taking different operations. Jiang et al. [11] divides the data into different groups according to their important characteristics. However, they enumerate all feature interactions equally for feature learning, which always requires large memory. In addition, useless feature interactions can introduce unnecessary noise and negatively impact the prediction accuracy.

In summary, the key limitations of existing approaches for CTR prediction, which exploit salient features, meaningful second-order feature interactions, or dominant high-order feature interactions in feature engineering, is that they generally have difficulty to effectively build an accurate and representative feature representation. To improve the prediction performance, it is useful to jointly consider the different contributions of features and feature interactions. Here, we introduce a hierarchical attention mechanism to learn the informative features and feature interactions at both low-order and high-order feature interactions, which provide an interpretable capability of the prediction results. Further, we design a projective bilinear function to effectively learn more fine-grained and comprehensive second-order feature interactions, which can enrich the information for modelling high-order feature interactions and further improve the prediction accuracy.

3 The proposed algorithm

We aim to automatically learn the relevant low- and high-order feature interactions in an end-to-end manner. As a result, we propose a Hierarchical Attention and Feature Projection neural network (HAFP) for CTR prediction.

In this section, we mainly describe the framework of HAFP. As shown in Fig. 1, HAFP has three main components: embedding layer, feature learning layer, and prediction layer. First, the embedding layer is used to convert each raw feature into a dense low-dimensional vector. Second, in order to derive the meaningful and representative feature representation, we employ a feature learning layer to encode features and feature interactions based on the output of the embedding layer. The feature learning layer consists of three parts: salient feature

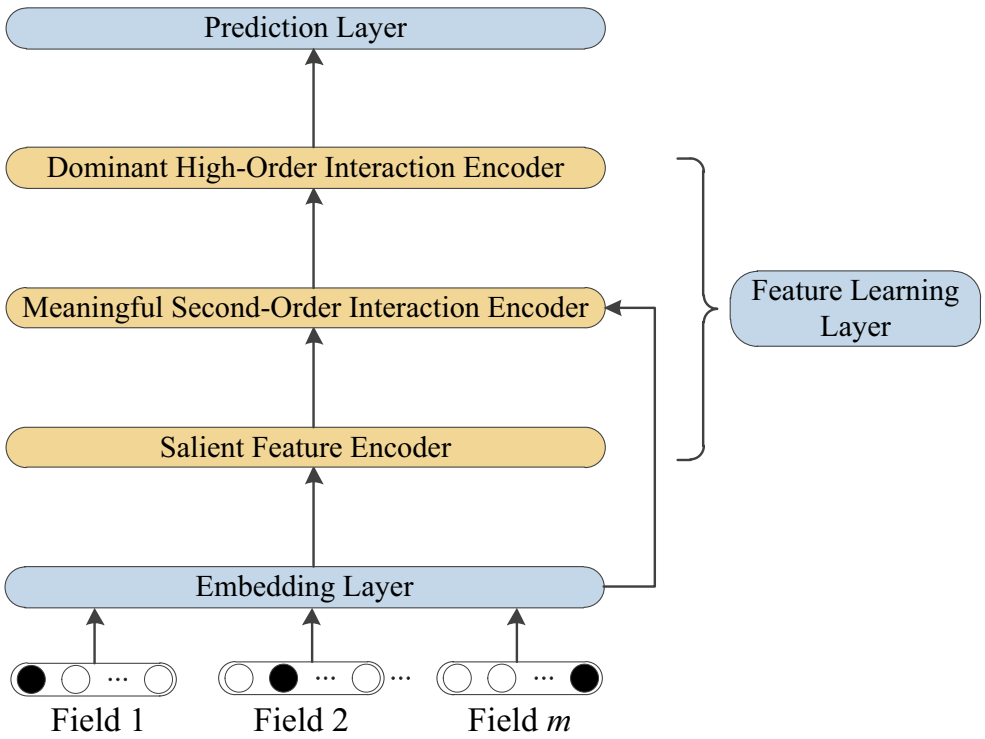

Fig. 1 The framework of Hierarchical Attention and Feature Projection neural network (HAFP)

encoder, meaningful second-order interaction encoder, and dominant high-order interaction encoder. The salient feature encoder transforms the dense low-dimensional feature vector into salient feature embedding with the help of an attentive global-local contexts module. This process pays more attention to the feature importance which dynamically places higher weights on the important features and decreases the weights of uninformative features. The encoder helps to boost reliability of feature embedding. The meaningful second-order interaction encoder transforms the interaction information of salient feature embeddings with the help of a projective bilinear function and a self-attention mechanism. This process not only fully explores the information of second-order feature interactions but also considers their importance for the target task. This encoder has capable to build more fine-grained second-order feature interactions and enhance interaction discriminability. The dominant high-order interaction encoder transforms feature interactions at different layers by using the attention mechanism. It can select dominant and irrelevant ones and assign weights for each layer dynamically. This encoder can build more representative and efficient feature representation. The prediction layer takes the output of the feature learning layer to compute the prediction score which represents the probability of the user clicking on the recommended product. The following sections introduce each part in detail.

3.1 Embedding layer

In the task of CTR prediction, data is always aggregated from different fields, and usually contains categorical and numerical features which cannot be directly used for numerical computations. Table 1 is an example of real-world multi-field data which is used for CTR prediction. To represent these kinds of features, they are often converted into high-dimensional sparse vectors by using one-hot encoding. However, since the embedding generated from one-hot encoder is always sparse and is hard to be processed, a lookup table processing approach is applied to transform each raw feature into a corresponding dense low-dimensional vector and form a field embedding vector. Finally, we use $\boldsymbol{E} = [\boldsymbol{e}_1, \boldsymbol{e}_2, ..., \boldsymbol{e}_m]$ to denote the field embedding vector, where m denotes the number of fields, and $\boldsymbol{e}_i \in R^d$ denotes the embedding of the i-th field feature, and d is the embedding size.

Table 1 An example of multi-field data for CTR prediction. Each of the columns is a field. Gender and Occupation are categorical features, and age is numerical feature

User_id	Gender	Age	Occupation
291245	Male	25	Student
362589	Female	27	Teacher
391027	Male	36	Programmer

3.2 Feature learning layer

3.2.1 Salient feature encoder

Squeeze and excitation network (SENET) [9] is efficient for learning feature importance in CTR prediction as it can effectively learn the relationships between each feature and the global context. However, we argue that it focuses on the common information and ignores the characterized information of each feature. Inspired by the success of attentional feature fusion [14] in computer vision, we design an attentive global-local contexts module in a salient feature encoder, which simultaneously takes global and local contexts into consideration to build salient feature embedding. Thus, it consists two sub-parts to separately retrieve the influences from the global context and the local context. The framework of the attentive global-local contexts module is shown in Fig. 2.

Attentive global-local contexts module. Given field embedding vector $\boldsymbol{E} = [\boldsymbol{e}_1, \boldsymbol{e}_2, ..., \boldsymbol{e}_m]$, the feature embedding learned from global context requires global context information. Therefore, we apply mean pooling on each feature embedding $\boldsymbol{e}_i$ to calculate global information $\boldsymbol{a}_i$, and form a global weight vector $A = [\boldsymbol{a}_1, \boldsymbol{a}_2, ..., \boldsymbol{a}_m]$. Then, we learn the weight of each feature embedding according to the global weight vector by using widely used dimensionality-reduction and dimensionality-increase method. Finally, a global feature embedding $\boldsymbol{V}_g$ is built based on the field embedding vector and weight vector by using a reweight method. The detailed calculations of these steps are shown as follows.

$$\boldsymbol{a}_i = \frac{1}{d}\sum_{j=1}^{d} \boldsymbol{e}_i^j \tag{1}$$

$$\boldsymbol{G} = [\boldsymbol{g}_1, \boldsymbol{g}_2, ..., \boldsymbol{g}_i, ..., \boldsymbol{g}_m] = \sigma_1(\boldsymbol{W}_{g1}\sigma_2(\boldsymbol{W}_{g2}\boldsymbol{A})) \tag{2}$$

$$\begin{aligned}\boldsymbol{V}_g &= [\boldsymbol{v}_{g1}, \boldsymbol{v}_{g2}, ..., \boldsymbol{v}_{gi}, ..., \boldsymbol{v}_{gm}] \\ &= [\boldsymbol{g}_1 \cdot \boldsymbol{e}_1, \boldsymbol{g}_2 \cdot \boldsymbol{e}_2, ..., \boldsymbol{g}_i \cdot \boldsymbol{e}_i, ...\boldsymbol{g}_m \cdot \boldsymbol{e}_m]\end{aligned} \tag{3}$$

where $\boldsymbol{e}_i^j$ denotes j-th value of the embedding of the i-th field feature. $\boldsymbol{G}$ is the global gate, and $\boldsymbol{g}_i$ denotes the global gate of the i-th field feature. $\boldsymbol{W}_{g1} \in R^{\frac{m}{r} \times m}$ and $\boldsymbol{W}_{g2} \in R^{m \times \frac{m}{r}}$ are learning parameters, in which r is the scaling factor and it is used to control the reduction and increases degree in computing weight vector. σ_1 and σ_2 are nonlinear activation functions. $\boldsymbol{v}_{gi}$ denotes the embedding of the i-th global feature embedding.

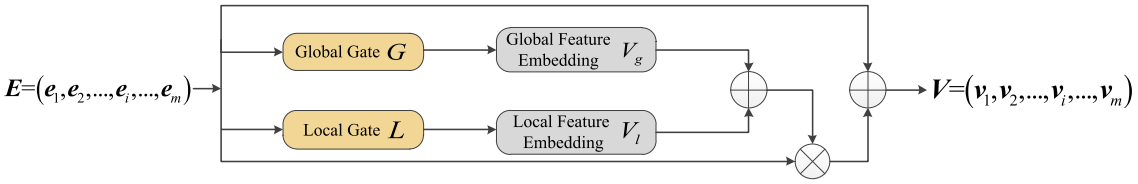

Fig. 2 The attentive global-local contexts module

Additionally, in order to capture the characterized information of each feature in modelling feature importance, we also take the local context into consideration. Specifically, given a field embedding vector $\boldsymbol{E} = [\boldsymbol{e}_1, \boldsymbol{e}_2, ..., \boldsymbol{e}_m]$, we directly employ dimensionality-reduction and dimensionality-increase mechanism in the field embedding to compute the contribution of individual feature for the target task.

$$\boldsymbol{L} = [\boldsymbol{l}_1, \boldsymbol{l}_2, ..., \boldsymbol{l}_i, ..., \boldsymbol{l}_m] = \sigma_1(\boldsymbol{W}_{l1}\sigma_2(\boldsymbol{W}_{l2}\boldsymbol{E})) \tag{4}$$

where $\boldsymbol{L}$ is the local gate, and $\boldsymbol{l}_i$ denotes the local gate of the i-th field feature. The function of $\boldsymbol{W}_{l1} \in R^{\frac{m}{r} \times m}$ and $\boldsymbol{W}_{l2} \in R^{m \times \frac{m}{r}}$ is similar to $\boldsymbol{W}_{g1} \in R^{\frac{m}{r} \times m}$ and $\boldsymbol{W}_{g2} \in R^{m \times \frac{m}{r}}$, and they are also used for dimension reduction and dimension increase. It is noteworthy that $\boldsymbol{L}$ has the same shape as the input field embedding vector $\boldsymbol{E}$, which preserves and highlights the subtle details of the local information. Then, we form a local feature embedding $\boldsymbol{V}_l$ by assigning a local weight on the field embedding vector.

$$\boldsymbol{V}_l = [\boldsymbol{v}_{l1}, \boldsymbol{v}_{l2}, ..., \boldsymbol{v}_{li}, ..., \boldsymbol{v}_{lm}] = [\boldsymbol{l}_1 \cdot \boldsymbol{e}_1, \boldsymbol{l}_2 \cdot \boldsymbol{e}_2, ..., \boldsymbol{l}_i \cdot \boldsymbol{e}_i, ..., \boldsymbol{l}_m \cdot \boldsymbol{e}_m] \tag{5}$$

where $\boldsymbol{v}_{li}$ denotes the embedding of the i-th local feature embedding $\boldsymbol{V}_l$.

Given the global feature embedding $\boldsymbol{V}_g$ and the local feature embedding $\boldsymbol{V}_l$, the salient feature embedding $\boldsymbol{V}$ can be obtained as follows:

$$\boldsymbol{V} = [\boldsymbol{v}_1, \boldsymbol{v}_2, ..., \boldsymbol{v}_i, ..., \boldsymbol{v}_m] = (\boldsymbol{E} \otimes \sigma(\boldsymbol{V}_l \oplus \boldsymbol{V}_g)) \oplus \boldsymbol{E} \tag{6}$$

where $\boldsymbol{v}_i$ denotes the i-th salient feature embedding. $\otimes$ and $\oplus$ denote the element-wise multiplication and addition. σ is nonlinear activation function. Comparing Eq. (2) with Eq. (4), we can observe that the global context can emphasize common information that distributes more globally, and the local context can highlight characterized information that distributes more locally. Thus, with the help of the attentive global-local contexts module, the salient feature encoder comprehensively emphasizes the features that distribute globally and locally, and the weight of each feature is dynamically adjusted according to its contribution.

3.2.2 Meaningful second-order interaction encoder

The meaningful second-order interaction encoder in our manuscript models the second-order feature interactions in a precise and effective way. An inner product and a Hadamard product are commonly used in existing works for modelling the second-order feature interactions. However, they are too simple to effectively calculate the feature interactions in sparse datasets [10]. To alleviate this limitation, FiBiNET proposes a field-interaction type for modelling second-order feature interactions by integrating an inner product and a Hadamard product, and it achieves good performance. However, we argue that it does not fully consider the relationships between pair-wise features. Specifically, the field-interaction type transforms the i-th feature into the j-th feature through an inner product and then models the interaction via a Hadamard product, which ignores the mapping relation from the j-th feature to the i-th feature. Therefore, we propose a more fine-grained approach called projective bilinear function which takes the overall mapping relations between two features through two inner products and obtains interaction relations on mapping features via the Hadamard product. Compared to the widely used inner product and Hadamard product, the projective bilinear function can encode more informative and inherent relations between different features. In addition, it facilitates the following encoder to learn meaningful information. The structure of the projective bilinear function is shown in Fig. 3, taking the i-th salient feature embedding $\boldsymbol{v}_i$ and the j-th salient feature embedding $\boldsymbol{v}_j$ as an example, their feature interaction $\boldsymbol{p}'_{ij}$ is calculated by:

$$\boldsymbol{p}'_{ij} = (\boldsymbol{v}_i \cdot \boldsymbol{W}_{pi}) \odot (\boldsymbol{v}_j \cdot \boldsymbol{W}_{pj}) \tag{7}$$

where $\odot$ is the element-wise product of vectors. $\boldsymbol{W}_{pi} \in R^{d \times d}$ and $\boldsymbol{W}_{pj} \in R^{d \times d}$ are learning parameters. The ranges of i and j are $1 \leq i \leq m$ and $i < j \leq m$. Compared to the field-interaction type, it has a stronger expression in modelling second-order feature interactions and forms a more fine-grained interaction.

Generally, not all of the feature interactions are relevant to the final task. Irrelevant feature interactions are considered as noise and may deteriorate the model generalization performance. Therefore, we introduce the attention mechanism to compute corresponding attention score with an MLP. The input is the vector of feature

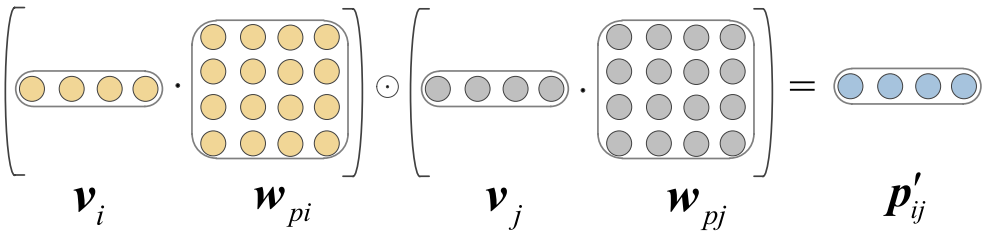

Fig. 3 The projective bilinear function for feature interaction

interaction. Formally, the attention score α_{ij}^p and weighted feature interaction $\boldsymbol{p}_{ij}$ are defined as:

$$\alpha_{ij}^p = \frac{\exp(\boldsymbol{h}_p^T \mathrm{ReLU}(\boldsymbol{W}_{ij}^p \boldsymbol{p}'_{ij} + b_p))}{\sum_{i,j} \boldsymbol{h}_p^T \mathrm{ReLU}(\boldsymbol{W}_{ij}^p \boldsymbol{p}'_{ij} + b_p)} \tag{8}$$

$$\boldsymbol{p}_{ij} = \alpha_{ij}^p \boldsymbol{p}'_{ij} \tag{9}$$

where $\boldsymbol{W}_{ij}^p \in R^{d \times d}$, $\boldsymbol{h}_p, b_p \in R^d$ are learning parameters, and $\boldsymbol{h}_p^T$ is the transposition of $\boldsymbol{h}_p$. Inspired by the success of Densely connected convolutional Networks (DenseNet) [19], the residual network [41] has the capability to provide more comprehensive deep features and to alleviate vanishing-gradient problems. Thus, we also implement our proposed second-order feature interaction method on the field embedding vector to strengthen and enrich interactions. Besides, we also compute attention scores for each feature interaction with an MLP to distinguish its importance. Namely, taking $\boldsymbol{e}_i$ and $\boldsymbol{e}_j$ as an example, the result of weighted field feature interaction $\boldsymbol{q}_{ij}$ can be computed as follows:

$$\boldsymbol{q}'_{ij} = (\boldsymbol{e}_i \cdot \boldsymbol{W}_{qi}) \odot (\boldsymbol{e}_j \cdot \boldsymbol{W}_{qj}) \tag{10}$$

$$\alpha_{ij}^q = \frac{\exp(\boldsymbol{h}_q^T \mathrm{ReLU}(\boldsymbol{W}_{ij}^q \boldsymbol{q}'_{ij} + b_q))}{\sum_{i,j} \boldsymbol{h}_q^T \mathrm{ReLU}(\boldsymbol{W}_{ij}^q \boldsymbol{q}'_{ij} + b_q)} \tag{11}$$

$$\boldsymbol{q}_{ij} = \alpha_{ij}^q \boldsymbol{q}'_{ij} \tag{12}$$

where $\boldsymbol{q}'_{ij}$ is the feature interaction of the i-th feature $\boldsymbol{e}_i$ and the j-th feature $\boldsymbol{e}_j$. α_{ij}^q is attention score of $\boldsymbol{e}_i$ and $\boldsymbol{e}_j$. $\boldsymbol{W}_{qi}$, $\boldsymbol{W}_{qj}$, $\boldsymbol{W}_{ij}^q \in R^{d \times d}$, $h_q, b_q \in R^d$ are learning parameters, and $\boldsymbol{h}_q^T$ is the transposition of $\boldsymbol{h}_q$. Finally, we also employ concatenation and fully-connected layers to learn the comprehensive second-order feature interactions $\boldsymbol{h}_s$ and endow them with a richer expressive ability.

$$\begin{aligned} \boldsymbol{h}_s &= [\boldsymbol{h}^1, \boldsymbol{h}^2, ..., \boldsymbol{h}^i, ..., \boldsymbol{h}^n] \\ &= FC(Concat(\boldsymbol{p}_1, \boldsymbol{p}_2, ..., \boldsymbol{p}_i, ..., \boldsymbol{p}_n, \boldsymbol{q}_1, \boldsymbol{q}_2, ..., \boldsymbol{q}_i, ..., \boldsymbol{q}_n)\boldsymbol{W}_{fc}) \end{aligned} \tag{13}$$

where $\boldsymbol{h}^i$ is the i-th embedding of the $\boldsymbol{h}_s$. $\boldsymbol{p}_i$ and $\boldsymbol{q}_i$ are vectors. $FC(\cdot)$ is fully-connected layers, and $\boldsymbol{W}_{fc}$ is linear matrix, and n is the number of feature interactions and is equal to $\frac{m(m-1)}{2}$. Figure 4 shows the processing of meaningful second-order interaction encoder.

3.2.3 Dominant high-order interaction encoder

Deep learning networks with several fully-connected layers are widely used in various works to extract high-order feature interactions, and a hierarchical structure has the capability to build more representative and efficient features. Normally, existing works merely take the output of the last layer as a dense real-value feature vector to make predictions. However, we argue that these methods ignore the relations among layers. Intuitively, features from different layers have different information for feature learning. Thus, we introduce an attention mechanism into the layers, and try to select dominant and irrelevant ones and assign weights for each layer dynamically, which can improve the feature extraction and is less costly.

The structure of the dominant high-order interaction encoder is shown in Fig. 5. Firstly, the comprehensive second-order feature interactions $\boldsymbol{h}_s$ is fed into a feed-

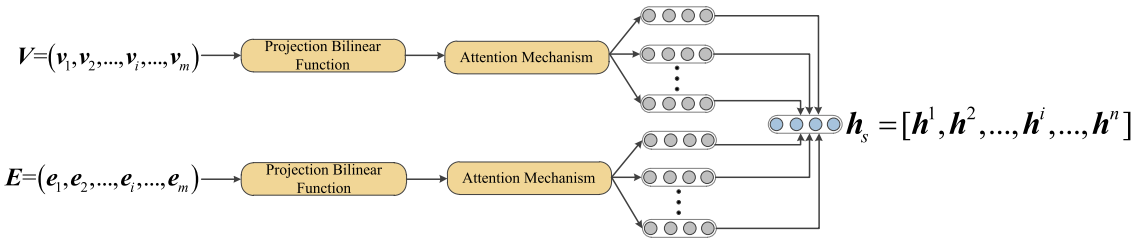

Fig. 4 The meaningful second-order interaction encoder

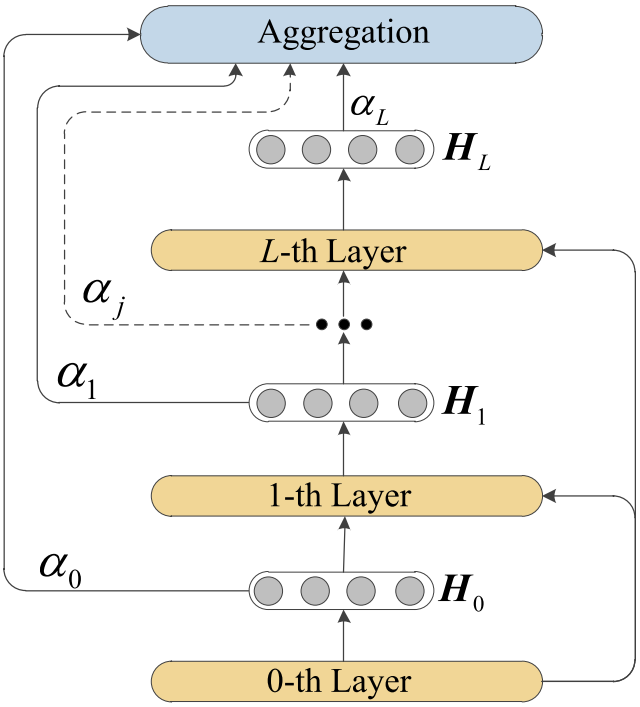

Fig. 5 The dominant high-order interaction encoder

forward neural network in which hidden layers have the same units. It is denoted as:

$$\boldsymbol{H}_1 = \sigma(\boldsymbol{W}_0\boldsymbol{H}_0 + b_0) \tag{14}$$

$$\boldsymbol{H}_L = \sigma(\boldsymbol{W}_{L-1}\boldsymbol{H}_{L-1} + b_{L-1}) \tag{15}$$

where $\boldsymbol{W}_0$ and b_0 are the weight matrix and bias vector of the 0-th layer. Similarly, $\boldsymbol{W}_{L-1}$ and b_{L-1} are the weight matrix and bias vector of the L-1th layer. $\boldsymbol{H}_0 = \boldsymbol{h}_s$, and the L denotes the hidden layer depth and σ is the activation function. $\boldsymbol{H}_1$ is the output of the 1-th layer. $\boldsymbol{H}_L$ is the output of the L-th layer. Thus, compared to the 0-th layer, the output of the L-th layer contains more comprehensive information. Then, in order to build more detailed and useful information, we take attention mechanism to aggregate these high-order features into a dense real-value feature vector $\boldsymbol{h}$.

$$\alpha_k = \frac{\exp(\boldsymbol{h}_k \mathrm{Re}LU(\boldsymbol{W}_k\boldsymbol{H}_k + b_k))}{\sum_{k=0}^{L} \exp(\boldsymbol{h}_k \mathrm{Re}LU(\boldsymbol{W}_k\boldsymbol{H}_k + b_k))} \tag{16}$$

$$\boldsymbol{h} = [\alpha_0\boldsymbol{H}_0, \alpha_1\boldsymbol{H}_1, ..., \alpha_L\boldsymbol{H}_L] \tag{17}$$

where α_k is the weight of k-th layer, which represents the different relations among hierarchical layers. $\boldsymbol{H}_k$ is the output of the k-th layer. $\boldsymbol{W}_k$, $\boldsymbol{h}_k$, and b_k are learning parameters. Through Eq. (16) and Eq. (17), hierarchical features are not equally aggregated for feature learning.

3.3 Prediction layer

Finally, $\boldsymbol{h}$ is fed into the sigmoid function for CTR prediction. It is formulated as:

$$\hat{y} = \sigma(\boldsymbol{W}_{L+1}\boldsymbol{h} + b_{L+1}) \tag{18}$$

where $\boldsymbol{W}_{L+1}$ and b_{L+1} are the model weight and the bias vector respectively. $\hat{y} \in (0, 1)$ is the predicted value. Furthermore, we use the widely used cross-entropy loss function to train our proposed HAFP model:

$$loss = \sum_{j \in N} (y_j \log(\hat{y}_j) + (1 - y_j) \log(1 - \hat{y}_j)) + \lambda \|\theta\|_2 \tag{19}$$

where N and θ are the total number of samples and parameter set of the model, respectively. y_j is the ground truth of the j-th instance. Additionally, we introduce L_2 regularization weighted by λ to prevent overfitting, and we use Adam gradient descent optimizer to optimize Eq. (19).

4 Experiments and analysis

4.1 Research questions

We conduct experiments which aim to answer the following research questions:

(RQ1) What is the performance of HAFP in CTR prediction? Does it outperform the state-of-the-art models in terms of Logloss and AUC? (See Section 4.3)

(RQ2) How well does HAFP perform with the hierarchical attention mechanism? (See Section 4.4)

(RQ3) How well does HAFP perform with different types of bilinear interactions? (See Section 4.5)

(RQ4) How well does HAFP perform with single context instead of global-local contexts in the salient feature encoder? (See Section 4.6)

Before implementing extensive experiments, we first present the experimental settings including datasets, evaluation metrics, baselines, and parameter settings.

4.2 Experiment settings

4.2.1 Datasets

We use two datasets which are commonly adopted in CTR prediction, Criteo[1] and Avazu[2], to evaluate the efficiency of the proposed model. The first dataset is released by

[1] http://labs.criteo.com/2014/02/download-dataset/

[2] http://www.kaggle.com/c/avazu-ctr-prediction

the Display Advertising Challenge 2014. It contains 39 anonymous fields about displayed ads which consist of 26 categorical fields and 13 continuous fields. The second dataset is released by the Feature Prediction Competition 2014. Its data relates to users' click behaviors on displayed mobile ads, and it has 24 fields about user/device features and ad attributes. In experiments, we split each dataset into three parts: 80% for training, 10% for validation, and 10% for testing.

4.2.2 Evaluation metrics

To evaluate the performance of the HAFP in CTR prediction, we take Area Under Curve (AUC) and Logloss as evaluation strategies which have been widely adopted in related works. Please note that an improvement of 1% in AUC or Logloss brings a large increase of revenue for advertising platforms [3, 4].

AUC: AUC is the primary evaluation, and is used to reflect the ranking performance between clicked and non-click instances. The upper bound of AUC is 1, and a higher value of AUC represents a better performance.

Logloss: Logloss measures the overall likelihood of test data. It has been widely used in the classification tasks, and a lower value of Logloss represents a better performance.

4.2.3 Baselines

We compare our model HAFP with different baseline methods for CTR prediction. The baselines include:

LR: LR employs linear combination with each feature to compute CTR prediction.

FM: [24] FM uses inner products on first-order and second-order feature interactions to compute CTR prediction.

AFM: [34] AFM extends FM by introducing an attention mechanism which distinguishes different weights of second-order feature interactions.

NFM: [7] Neural Factorization Machine(NFM) builds the feature interactions via a bi-interaction pooling layer before DNNs.

DeepFM: [6] DeepFM extracts feature interactions by combining an FM part and a deep MLP part.

InterHAt: [16] InterHAt models feature interactions by using a multi-head transformer and a hierarchical attention layers.

FiBiNET: [10] FiBiNET employs a squeeze-and-excitation network layer and a bilinear-interaction layer to explore salient features and feature interactions for CTR prediction.

4.2.4 Implementation details

We implement HAFP and baselines with Tensorflow on a GPU Tesla T4. In the embedding layer, the dimension of each feature in our work is set to 8 for the Avazu dataset and to 10 for the Criteo dataset. The scaling factor r in the salient feature encoder is set to 3, and the activation functions in the attentive global-local contexts module is RELU. In the dominant high-order interaction encoder, the hidden layer depth L is set to 4, and the activation functions in this encoder are RELU. Additionally, for optimizing the HAFP model, we use Adam to update parameters in the training stage with a mini-batch size of 512 for Criteo and Avazu datasets, and we set the learning rate to 0.0001 on the two datasets. Additionally, to ensure the reliability of model performances, reported results are the average value of 5 iterations of the experiments. Moreover, the parameters of all of the baseline models follow the experimental settings reported in their works for fair comparisons.

4.3 Performance comparison

The results for CTR prediction between HAFP and baselines on Criteo and Avazu datasets are shown in Table 2. From the table, we can find the following observations:

LR performs worse than other methods, which indeed shows the power of feature interactions in feature learning. AFM achieves a significantly better performance than FM, which demonstrates the benefits of an attention mechanism in learning the weights of feature interactions. Furthermore, LR and FM based methods perform worse than the methods which incorporate a deep learning network in CTR prediction. It demonstrates the effectiveness of non-linear transformation and deep neural network in modelling feature interactions. InterHAt benefits from second-order feature interactions more than NFM and DeepFM, and it

Table 2 Performance comparison on two datasets for Logloss and AUC, respectively. The statistical significance between each pair of our proposed HAFP and the best baseline at $p<0.05$ level

Method	Criteo		Avazu	
	Logloss	AUC	Logloss	AUC
LR	0.5133	0.7308	0.4378	0.7276
FM	0.5040	0.7397	0.4352	0.7355
AFM	0.4964	0.7441	0.4337	0.7362
NFM	0.4897	0.7483	0.4310	0.7387
DeepFM	0.4892	0.7491	0.4283	0.7398
InterHAt	0.4842	0.7580	0.3931	0.7549
FiBiNET	0.4843	0.7585	0.3979	0.7583
HAFP	0.4824	0.7608	0.3903	0.7600

achieves better performance. Capturing feature importance is important for feature learning because different features have various contributions for the final task. InterHAt represents second-order feature interactions with a multi-head transformer, but it neglects to consider the feature importance in feature embedding. FiBiNET performs better than InterHAt. That is mainly due to the salient features with a SENET structure which captures more accurate embeddings of each feature.

Compared to InterHAt and FiBiNET, HAFP further builds feature learning from the hierarchical attention mechanism, which captures the real-value feature vector from both low- and high-order feature interactions simultaneously. HAFP takes full account of relevant information and outperforms the baselines. Generally, HAFP obtains improvements of 0.3% and 0.2% in AUC on two datasets over the best baseline FiBiNET, respectively.

Furthermore, to verify whether the relative improvement rates of HAFP are statistically significant, we conduct a paired t-test here and results of the p-values are shown in Table 2 with different markers. In this table, the p-value refers to the comparison between HAFP and the best baseline. The p-values in Table 2 are all less than 0.05, which validates the improvements of HAFP are statistically significant.

4.4 Influence of hierarchical attention mechanism

To illustrate the influence of the hierarchical attention mechanism for CTR prediction, we compare the performance of HAFP and three variants of HAFP. FP-0 refers to HAFP without the salient feature encoder and without the attention mechanism in the meaningful second-order interaction encoder and the dominant high-order interaction encoder. FP-1 refers to HAFP without the attention mechanism in the meaningful second-order interaction encoder and the dominant high-order interaction encoder. FP-12 refers to HAFP without the attention mechanism in the dominant high-order interaction encoder. The results on Criteo and Avazu are summarized in Table 3.

First, although FP-0 is the worst method for CTR prediction in the variants of HAFP, compared to the similar structure of FiBiNET, FP-0 gains a better performance. Considering the difference between FP-0 and FiBiNET, the results show that the projective bilinear function plays an important role for feature learning, and the performance verifies the effectiveness of the designed projective bilinear function. Second, compared to FP-0, FP-1 obtains some improvements, which indicates the effect of considering feature importance in feature learning, and the performance verifies that building accurate feature embedding has the capability to improve CTR prediction. Third, FP-12 achieves a better performance than FP-0 and FP-1 in terms of Logloss and AUC on the two datasets. This confirms that using the attention mechanism in the second-order interaction encoder can capture the relevant contexts for the feature learning. In addition, it means that not all pair-wise feature interactions are equally useful for CTR prediction tasks. Finally, compared to the above three variants of HAFP, HAFP receives the best performance, which indicates that the hierarchical attention mechanism contributes to our model. Additionally, HAFP outperforms FP-0 with 0.2% and 0.2% respectively for AUC on the two datasets. The results demonstrate our hierarchical attention mechanism can capture the positive features and feature interactions for CTR prediction tasks, which verify the rationality and feasibility of our contribution in this work. In summary, the performance comparison of HAFP with different encoders demonstrates that with the designed strategy, better feature representation can be achieved and further help CTR prediction.

Table 3 Performance comparison of HAFP with different encoders

Models	Criteo		Avazu	
	Logloss	AUC	Logloss	AUC
FP-0	0.4837	0.7592	0.3922	0.7584
FP-1	0.4835	0.7593	0.3918	0.7588
FP-12	0.4832	0.7597	0.3912	0.7592
HAFP	0.4824	0.7608	0.3903	0.7600

4.5 Influence of bilinear function

To study the effectiveness of our projective bilinear function for learning the second-order feature interactions, we conduct the ablation experiments in this section to study the impact of the projective bilinear function in HAFP. HAFI refers to that field-interaction type is used to encode the feature interaction in the meaningful second-order interaction encoder. HAT refers to that multi-head transformer is used to encode the feature interaction in the meaningful second-order interaction encoder. As Fig. 6 shows, HAFI obtains a better performance in both AUC and Logloss than HAT on the two datasets. Additionally, the computation of HAT is heavier than HAFI since HAT owns more parameters and requires more time for training. Thus, bilinear interaction function with attention mechanism is beneficial and suitable for modelling second-order feature interactions. Furthermore, HAFP outperforms HAFI and HAT in both the datasets, which verifies the effectiveness of the projective bilinear function. The main reason is that it fully considers mapping relation between two features and the interaction results are more comprehensive.

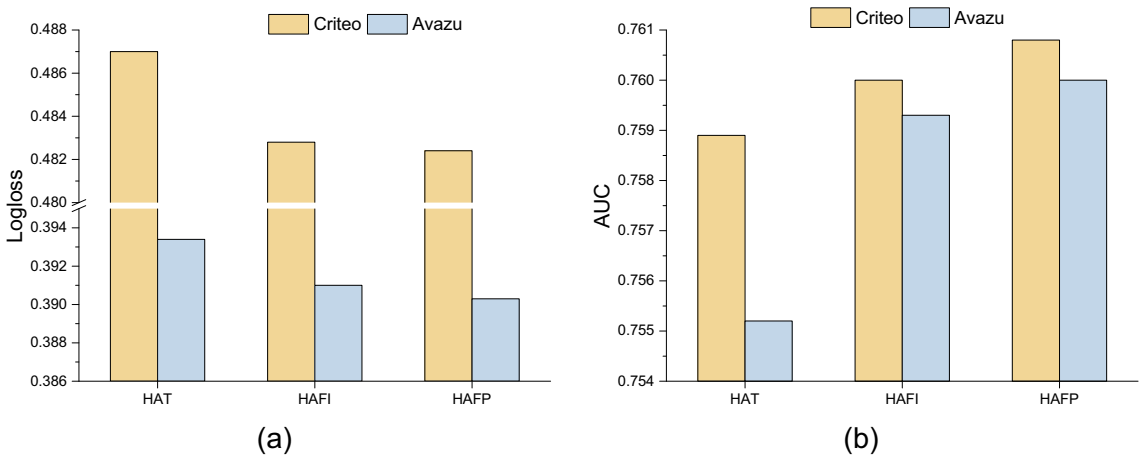

Fig. 6 Performance comparison of HAFP with different bilinear functions

4.6 Influence of attentive global-local context module

To study the impact of our attentive global-local module (AGLM), we construct its two ablation modules, attentive global module (AGM) and attentive local module (ALM), in which the other parts of HAFP are set the same. The only difference in these three modules is the contextual information used in building feature importance. The comparison results are shown in Fig. 7. It can be seen that:

(1) HAFP with AGM perform slightly better than HAFP with ALM in Logloss, while the latter gains better performance in terms of AUC on Avazu. Since the global context is used to extract common information and the local context is used to retrieve characterized information, we argue that both global information and local information play an important role for building feature importance.

(2) HAFP with AGLM achieves better performance than AGM and ALM in all settings. It suggests that merely focusing on global or local information is too biased and weakens the effect of feature learning. In summary, integrating global and local information is vital for dynamically assigning feature weight and is an effective way for improving CTR prediction.

5 Conclusion

In this paper, we highlight the relevant information in different order feature interactions for CTR prediction. We propose a novel Hierarchical Attention and Feature Projection (HAFP) neural network. There are two major parts in HAFP: 1) It employs a three-level attention mechanism to strengthen the weight of relevant features and decrease the weight of irrelevant features. 2) It designs a projective bilinear function to learn more

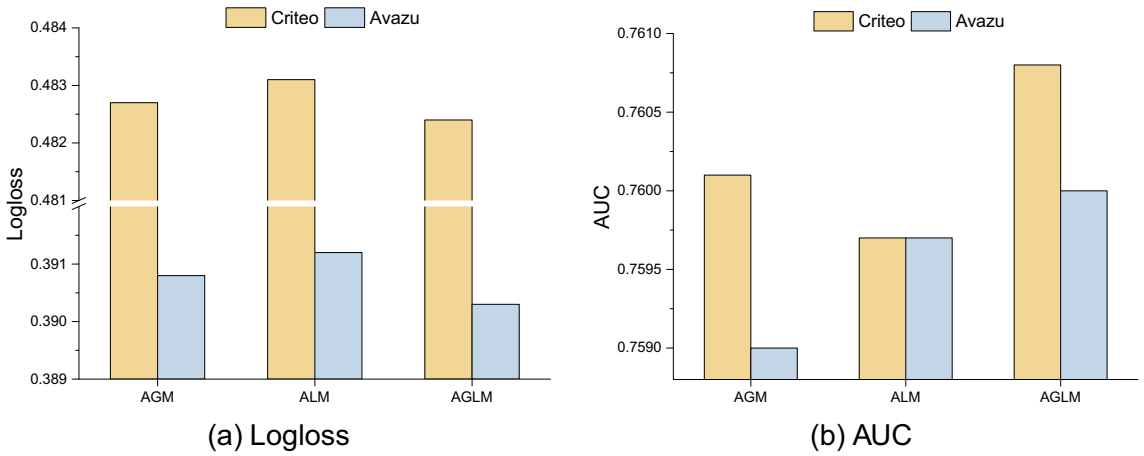

Fig. 7 Performance comparison of HAFP with different contexts in attentive global-local module

fine-grained second-order feature interactions. Compared to the existing methods, our model fully utilizes the interactions of different field pairs and automatically selects dominant features and feature interactions for feature learning. We conduct extensive experiments on two public datasets. The results show that HAFP outperforms state-of-the-art baselines for CTR prediction, and ablation experiments which analyze the effect of hierarchical attention, the bilinear function, and the attentive global-local context module demonstrate the rationality and effectiveness of our contributions.

Although the prediction performance is improved in our proposed method, it models the feature interactions in an implicit way, and unstructured combination of features will inevitably limit the capability to model sophisticated interactions among different features in a sufficiently flexible fashion. This limitation opens up new research possibilities. In future work, we plan to build sophisticated interactions among different features in an explicit manner. Moreover, inspired by the power of graph neural network (GNN), we are going to attempt to extend our work with GNN to further improve the performance.

References

1. Avila Clemenshia P, Vijaya MS (2016) Click through rate prediction for display advertisement. International Journal of Computer Applications 975:8887
2. Cai W, Wang Y, Ma J, Jin Q (2021) Can: Effective cross features by global attention mechanism and neural network for ad click prediction. Tsinghua Sci Technol 27(1):186–195
3. Bo C, Ding Y, Xin X, Li Y, Wang Y, Wang D (2021) Airec: Attentive intersection model for tag-aware recommendation. Neurocomputing 421:105–114
4. Cheng H-T, Koc L, Harmsen J, Shaked T, Chandra T, Aradhye H, Anderson G, Corrado G, Chai W, Ispir M, Anil R, Haque Z, Hong L, Jain V, Liu X, Shah H (2016) Wide & deep learning for recommender systems. In: Proceedings of the 1st Workshop on Deep Learning for Recommender Systems, pp 7–10
5. Frey RM, Xu R, Ammendola C, Moling O, Giglio G, Ilic A (2017) Mobile recommendations based on interest prediction from consumer's installed apps-insights from a large-scale field study. Inf Syst 71:152–163
6. Guo H, Tang R, Ye Y, Li Z, He X (2017) Deepfm: a factorization-machine based neural network for CTR prediction. In: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, pp 1725–1731
7. He X, Chua T-S (2017) Neural factorization machines for sparse predictive analytics. In: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp 355–364
8. Hong W, Xiong Z, You J, Wu X, Xia M (2021) CPIN: Comprehensive present-interest network for CTR prediction. Expert System Application 168:114469
9. Hu J, Li S, Sun G (2018) Squeeze-and-excitation networks. In: Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition, pp 7132–7141
10. Huang T, Zhang Z, Zhang J (2019) Fibinet: combining feature importance and bilinear feature interaction for click-through rate prediction. In: Proceedings of the 13th ACM Conference on Recommender Systems, pp 169–177
11. Jiang D, Xu R, Xu X, Xie Y (2021) Multi-view feature transfer for click-through rate prediction. Inf Sci 546:961–976
12. Juan Y-C, Zhuang Y, Chin W-S, Lin C-J (2016) Field-aware factorization machines for CTR prediction. In: Proceedings of the 10th ACM Conference on Recommender Systems, pp 43–50
13. Li D, Hu B, Chen Q, Wang X, Qi Q, Wang L, Liu H (2021) Attentive capsule network for click-through rate and conversion rate prediction in online advertising. Knowl-Based Syst 211:106522
14. Li G, Gan Y, Wu H, Xiao N, Lin L (2019) Cross-modal attentional context learning for rgb-d object detection. IEEE Trans Image Process 28(4):1591–1601
15. Li H, Duan H, Zheng Y, Wang Q, Wang Yu (2020) A CTR prediction model based on user interest via attention mechanism. Appl Intell 50(4):1192–1203
16. Li Z, Cheng W, Chen Y, Chen H, Wang W (2020) Interpretable click-through rate prediction through hierarchical attention. In: Proceedings of the Thirteenth ACM International Conference on Web Search and Data Mining, pp 313–321
17. Liu B, Zhu C, Li G, Zhang W, Lai J, Tang R, He X, Li Z, Yu Y (2020) Autofis: Automatic feature interaction selection in factorization models for click-through rate prediction. In: Proceedings of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp 2636–2645
18. Liu M, Cai S, Lai Z, Qiu L, Hu Z, Yi D (2021) A joint learning model for click-through prediction in display advertising. Neurocomputing 445:206–219
19. Lodhi B, Kang J (2019) Multipath-densenet: a supervised ensemble architecture of densely connected convolutional networks. Inf Sci 482:63–72
20. Luo Y, Wang M, Zhou H, Yao Q, Tu W-W, Chen Y, Dai W, Yang Q (2019) Autocross: Automatic feature crossing for tabular data in real-world applications. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp 1936–1945
21. Ma C, Mu X, Lin R, Wang S (2021) Multilayer feature fusion with weight adjustment based on a convolutional neural network for remote sensing scene classification. IEEE Geoscience and Remote Sensing Letters 18:241–245
22. Pan J, Xu J, Ruiz AL, Zhao W, Pan S, Sun Yu, Lu Q (2018) Field-weighted factorization machines for click-through rate prediction in display advertising. In: Proceedings of the 2018 World Wide Web Conference on World Wide Web, pp 1349–1357
23. Qu Y, Cai H, Ren K, Zhang W, Yu Y, Wen Y, Wang J (2016) Product-based neural networks for user response prediction. In: Proceedings of the IEEE 16th International Conference on Data Mining, pp 1149–1154
24. Steffen R (2012) Factorization machines with libfm. ACM Transactions on Intelligent Systems and Technology 3(3):1–22
25. Shan Y, Ryan Hoens T, Jiao J, Wang H, Yu D, Mao JC (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, pp 255–262
26. Silveira T, Zhang M, Liu Y, Ma S (2019) How good your recommender system is? a survey on evaluations in recommendation. International Journal of Machine Learning and Cybernetics 10:813–831
27. Song K, Huang Q, Zhang F, Lu J (2021) Coarse-to-fine: a dual-view attention network for click-through rate prediction. Knowledge Based Systems 216:106767

28. Tao Z, Wang X, He X, Huang X, Chua T-S (2020) Hoafm: A high-order attentive factorization machine for CTR prediction. Inf Process Manag 57:102076
29. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I (2017) Attention is all you need. In: Advances in neural information processing systems 30: annual conference on neural information processing systems 2017, pp 5998–6008
30. Wang Q, Fang'ai Liu PH, Xing S, Zhao X (2020) A hierarchical attention model for ctr prediction based on user interest. IEEE Syst J 14(3):4015–4024
31. Wang Q, Liu F, Huang Pu, Xing S, Zhao X (2020) A hierarchical attention model for CTR prediction based on user interest. IEEE Syst J 14(3):4015–4024
32. Wang Q, Fang'ai Liu SX, Zhao X (2019) Research on CTR prediction based on stacked autoencoder. Appl Intell 49(8):2970–2981
33. Wang R, Fu B, Fu G, Wang M (2017) Deep & cross network for ad click predictions. In: Proceedings of the ADKDD'17, pp 1–7
34. Xiao J, Ye H, He X, Zhang H, Wu F, Chua T-S (2017) Attentional factorization machines: Learning the weight of feature interactions via attention networks. In: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, pp 3119–3125
35. En Xu, Yu Z, Guo B, Cui H (2021) Core interest network for click-through rate prediction. ACM Transactions Knowledge Discovery Data 15(2):1–16
36. Xue N, Liu B, Guo H, Tang R, Zhou F, Zafeiriou SP, Zhang Y, Wang J, Li Z (2020) Autohash: Learning higher-order feature interactions for deep ctr prediction. IEEE Trans Knowl Data Eng, pp 1–1
37. Yan C, Chen Y, Wan Y, Wang P (2021) Modeling low- and high-order feature interactions with FM and self-attention network. Appl Intell 51(6):3189–3201
38. Yan C, Li X, Chen Y, Zhang Y (2021) Jointctr: a joint ctr prediction framework combining feature interaction and sequential behavior learning. Appl Intell, pp 1–14
39. Yi Y, Xu B, Shen S, Shen F, Zhao J (2020) Operation-aware neural networks for user response prediction. Neural Netw 121:161–168
40. Zhang W, Du T, Wang J (2016) Deep learning over multi-field categorical data - - a case study on user response prediction. In: Advances in information retrieval - 38th european conference on IR research, vol 9626, pp 45–57
41. Zhong Z, Li J, Luo Z, Chapman M (2018) Spectral–spatial residual network for hyperspectral image classification: a 3-d deep learning framework. IEEE Trans Geosci Remote Sens 56(2):847–858
42. Zou D, Wang Z, Zhang L, Zou J, Qi Li, Chen Y, Sheng W (2021) Deep field relation neural network for click-through rate prediction. Inf Sci 577:128–139