Hierarchical Transformer-Based Attention Prediction for User Focus Modeling in Digital Media

Fenghui Liu[1*], Jun Huang[2]
[1]Shandong University of Political Science and Law, East Jiefang Road, Jinan, Shandong, 250014, China
[2]Department of Engineering Science, Faculty of Innovation Engineering, Macau University of Science and Technology, Macao, 999078, China
Corresponding author: FengHui_Liu@outlook.com

Keywords: Digital media, attention mechanism, attention focus prediction, hierarchical attention network, transformer framework

Received: November 3, 2025

In the digital media environment, user attention is influenced by various factors, such as content features, temporal features, etc. These features have heterologous characteristics. Under the influence of this feature, it is impossible to accurately mine the potential features of user attention in the time domain, resulting in insufficient prediction accuracy. Therefore, this article proposes an attention mechanism prediction algorithm based on Transformer enhanced hierarchical attention network to address the challenge of rapid shift of user attention focus in digital media environment. By constructing a hierarchical attention network that combines content level self attention and temporal level recurrent attention, key behavior nodes and time-dependent features in user interaction sequences are dynamically captured. Based on the Transformer framework, the multi head attention mechanism is optimized to achieve end-to-end training for focus prediction. The experiment was conducted on four real datasets: Frappe, MovieLens, Criteo, and Avazu. The results showed that the algorithm performed well in prediction accuracy related indicators, reaching an AUC value of 0.9848 on the Frappe dataset, an average improvement of 15% -20% compared to the comparison method. It can respond to changes in user interests in real time and provide accurate decision support for personalized content recommendation and platform operation optimization.

Povzetek: Članek predstavlja izboljšan model za napovedovanje uporabniške pozornosti v digitalnih medijih, ki natančneje zaznava spremembe interesov in podpira personalizirana priporočila.

1 Introduction

With the continuous iteration of Internet technology and the vigorous development of digital media industry, digital media has become the core channel for information dissemination and interaction in modern society [1]. Relying on diverse media forms such as text, images, audio, and video, digital media has deeply penetrated into fields such as social networks, online education, e-commerce, and digital entertainment, reconstructing the paradigm of information dissemination and social interaction models [2]. At present, the number of online video users has reached 1.079 billion, with an average daily usage time of over 2.5 hours. The real-time interaction and dissemination of massive information have contributed to the attention economy. In this context, the research on the allocation and transfer mechanism of user attention, as the core resource of information processing and decision-making, has become a key proposition in the field of digital media. The current digital media environment presents significant characteristics of information overload and fragmented user attention [3]. On the one hand, the daily average amount of information received by a single user has increased by more than 10 times compared to ten years ago, exacerbating the contradiction between information supply and user processing capabilities [4]; On the other hand, the dwell time of user attention in digital media interfaces has been reduced to seconds, and is influenced by multimodal factors such as content semantics, visual elements, and interaction design, making it difficult for traditional prediction methods to effectively capture the dynamic changes in attention [5]. In addition, individual differences among users, such as interest preferences, cognitive habits, emotional states, and emergent characteristics of group behavior, further exacerbate the complexity and uncertainty of attention focus prediction.

At present, relevant research aims to accurately grasp the focus of users' attention in massive digital information, providing strong support for personalized services, precision marketing, and so on. The following summarizes and analyzes the relevant research, as shown in Table 1.

Table 1: Summary of related work

Reference method	Model type	Mode of processing	Personalized ability	Using the dataset	Performance of the report	Lack of content	How does this method solve the problem
Zhang Y et al. [6]	Bayesian attention model	No specific modality was explicitly mentioned, mainly targeting user behavior data	Capable of capturing users' flexible and varied interests and preferences, with a certain degree of personalization ability	No specific dataset was explicitly mentioned	Significantly improve CTR prediction performance	When prior knowledge is inaccurate, the model's ability to capture user intent will be affected	This method does not rely on prior knowledge and dynamically captures key behavior nodes and time-dependent features in user interaction sequences by constructing a hierarchical attention network, which can more accurately predict user attention focus
Li X et al. [7]	ALRF deep learning model	Eye tracking data and virtual object spatial features	Show good adaptability to different virtual environments and have certain personalized potential	VR museum and other scene data	Prediction accuracy surpassing traditional models	Performance is highly dependent on the scale and quality of training data, and may not achieve optimal results in scenarios with limited data resources	This method is based on Transformer enhanced hierarchical attention network, which has relatively low dependence on data size and quality, and can better adapt to different scenarios
YU G et al. [8]	Content sharing algorithm	Integrate user historical behavior, social relationships, and content characteristics from three dimensions of information	By constructing a dynamic user preference model, there is a certain degree of personalization ability	No specific dataset was explicitly mentioned	Significantly improved content delivery rate and network efficiency	When there is noise or incomplete input data, the recommendation accuracy of the algorithm will significantly decrease	The method proposed in this article is optimized through a hierarchical attention network and multi head attention mechanism, which has better robustness against data

							noise and incompletenes s, and can more accurately predict attention focus
LUO H et al. [9]	Multi task attention network	No specific modality was explicitly mentioned for multiple related tasks	Capable of adaptively learning the commonalitie s and differences of tasks in feature space and label space, with a certain degree of personalized generalizatio n ability	No specific dataset was explicitly mentione d	Effectively improve model performanc e	There is still room for improvement in the dynamic adaptability of models in the face of real-time changes in user behavior and network environments	This method is based on the Transformer framework and can capture changes in user interests in real time, with better dynamic adaptability

The current research faces the problem of insufficient generalization ability when facing complex and dynamic digital media environments, making it difficult to adapt to new content forms and user behavior patterns. Moreover, there is still a lack of in-depth research on individual differences among users, such as cognitive style, emotional state, and other factors, in predicting attention focus, which makes it difficult for prediction models to achieve high personalization [10]. This study focuses on the question of whether the hierarchical attention mechanism enhanced by Transformer can outperform existing click through rate prediction models on multimodal digital media datasets. The expected application scenario is recommendation systems, aiming to improve the personalized recommendation ability of recommendation systems in complex digital media environments by proposing an attention-based algorithm for predicting user attention focus, more accurately predicting user attention focus, and providing content recommendations that better meet their interests and needs. Therefore, this article proposes a digital media user attention focus prediction algorithm based on attention mechanism.

2 Building a hierarchical attention network

The attention mechanism simulates the human brain's ability to selectively focus on important information, giving the model the flexibility to dynamically allocate computing resources. By calculating the weights of each part of the input data, it achieves the focus on key information [11]. However, in the digital media environment, user behavior data presents complex features such as multimodal interweaving and strong temporal dynamics. The inherent defects of traditional attention mechanisms make it difficult to meet the demand for accurate prediction of user attention focus [12]. Therefore, building a hierarchical attention network is crucial. We propose a novel integration of content level self attention and temporal level cyclic attention to deeply analyze user interaction sequences from different dimensions. Through this hierarchical architecture, it is possible to capture user behavior patterns more comprehensively and meticulously, overcome the limitations of a single attention mechanism, and lay a solid foundation for accurate prediction in the future.

2.1 Content level self-attention

The traditional self-attention mechanism focuses on modeling the general dependency relationships between elements in the input sequence in applications, with the aim of improving the model's overall understanding and representation ability of data. Extract the nearest n interactions from the user interaction sequence as inputs for traditional self-attention mechanisms [13]. Use a learnable item embedding matrix $\boldsymbol{G} \in R^{|s_u| \times d}$ for the input sequence $s = \{i_1, i_2, \cdots, i_n\}$, embed the item sequence as $\boldsymbol{E} = \{e_1, e_2, \cdots, e_n\}$, where $e_i \in R^d$ outputs as $\boldsymbol{O} = \{\boldsymbol{o}_1, \boldsymbol{o}_2, \cdots, \boldsymbol{o}_n\}$. The calculation process is as follows:

$$\boldsymbol{Q} = \boldsymbol{E}\boldsymbol{W}_q, \boldsymbol{K} = \boldsymbol{E}\boldsymbol{W}_k, \boldsymbol{V} = \boldsymbol{E}\boldsymbol{W}_v \quad (1)$$

$$\boldsymbol{A} = \frac{\boldsymbol{Q}\boldsymbol{K}}{\sqrt{d}} \quad (2)$$

$$\boldsymbol{O} = soft\max\left(\boldsymbol{A}\right)\boldsymbol{V} \quad (3)$$

Among them, $\boldsymbol{W}_q$, $\boldsymbol{W}_k$, and $\boldsymbol{W}_v$ are trainable parameter matrices. $\boldsymbol{A}$ is the attention matrix, $\sqrt{d}$ is the scaling factor. Each vector is $\boldsymbol{o}_1 \in R^d$.

This article embeds the item sequence $s = \{i_1, i_2, \cdots, i_n\}$ in the traditional self-attention mechanism as $\boldsymbol{E} = \{e_1, e_2, \cdots, e_n\}$, representing the item content [14]. The maximum distance between relative positions between sequences is set to k, and the relative position of the item at position i in the user interaction sequence to the item at position j is represented as $\boldsymbol{P}_{\sigma(i,j)}$. The calculation method for the relative distance $\sigma(i,j)$ is as follows:

$$\sigma(i,j) = \begin{cases} 0, i-j \le -k \\ 2k-1, i-j \ge k \\ i-j+k, -k < i-j < k \end{cases} \quad (4)$$

The traditional self-attention mechanism outputs an $\overline{o}_i$ for every $\overline{e}_i$ input. Therefore, this paper incorporates relative positional information when calculating attention weights, and the calculation method is as follows:

$$\boldsymbol{Q}_r = \bar{\boldsymbol{E}}\boldsymbol{W}_q, \boldsymbol{K} = \bar{\boldsymbol{E}}\boldsymbol{W}_k, \boldsymbol{V} = \bar{\boldsymbol{E}}\boldsymbol{W}_v \quad (5)$$

$$\bar{A}_{ij} = \boldsymbol{Q}_i^r \left(\boldsymbol{K}_j^r\right)^T + \boldsymbol{Q}_i^r \left(\boldsymbol{P}_{\sigma(i,j)}^r\right)^T \quad (6)$$

$$\bar{\boldsymbol{O}} = \text{soft}\max\left(\frac{\bar{\boldsymbol{A}}}{\sqrt{2d}}\right)\boldsymbol{V}_r \quad (7)$$

Among them, $\boldsymbol{P}_{\sigma(i,j)}^r$ is the $\boldsymbol{P}$-th row of the relative position matrix $\sigma(i,j)$, i and j are the subscripts for calculating the relative position elements, and applying the scaling factor $\frac{1}{\sqrt{2d}}$ can prevent the inner product from being too large when the vector dimension is too high, and make the algorithm more stable. $\bar{A}_{ij}$ is an element in the attention matrix $\bar{\boldsymbol{A}}$, $\boldsymbol{Q}_i^r$ is the i-th row of the query matrix $\boldsymbol{Q}_r$, $\boldsymbol{K}_j^r$ is the j row of the key matrix $\boldsymbol{K}_j$, $\bar{\boldsymbol{O}}$ is the output value of the improved multi head self attention mechanism [15].

2.1.1 Title feature extraction

The title feature extraction module aims to learn the representation of projects from project title information. The title feature extraction module consists of three parts, namely title embedding, improved multi head self-attention mechanism, and word level attention mechanism [16]. The word sequence for the title is $\{w_1, w_2, \cdots, w_L\}$.

Using query operations and a learnable word embedding table $\boldsymbol{M}^I \in R^{V\times d}$, embed the sequence of title words into $\hat{\boldsymbol{E}} \in R^{L\times d}$, where L is the number of words in the title.

The improved multi head self-attention mechanism can effectively extract the content information and relative position information of the input itself. Due to the fact that one word in a title may have inherent connections with multiple words, an improved multi head self-attention mechanism is used to learn the contextual representation of the title. The calculation process is as follows:

$$\boldsymbol{Q}_t = \hat{\boldsymbol{E}}\boldsymbol{W}_{qt}, \boldsymbol{K}_t = \hat{\boldsymbol{E}}\boldsymbol{W}_{kt}, \boldsymbol{V}_t = \hat{\boldsymbol{E}}\boldsymbol{W}_{vt} \quad (8)$$

$$\bar{A}_{i,j}^{\ t} = \boldsymbol{Q}_i^t \left(\boldsymbol{K}_j^t\right)^T + \boldsymbol{Q}_i^t \left(\boldsymbol{P}_{\sigma(i,j)}^t\right)^T \quad (9)$$

$$\boldsymbol{head}_i^t = \text{soft}\max\left(\frac{\bar{\boldsymbol{A}}_t}{\sqrt{2d}}\right)\boldsymbol{V}_t \quad (10)$$

$$\boldsymbol{O}^t = Concat\left(\boldsymbol{head}_1^t, \boldsymbol{head}_2^t, \cdots, \boldsymbol{head}_m^t\right)\boldsymbol{V}_t \quad (11)$$

Among them, $\boldsymbol{W}_{qt}$, $\boldsymbol{W}_{kt}$, $\boldsymbol{W}_{vt}$, and $\boldsymbol{W}_t$ are learnable parameter matrices. $\bar{A}_{i,j}^{\ t}$ is an element in the attention matrix $\bar{\boldsymbol{A}}_t$. m is the number of attention heads. $\boldsymbol{O}^t$ can also be referred to as $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_L\}$.

Within the same title, different words may have varying degrees of importance when representing the title. In addition, previous attention mechanisms did not take into account the influence of user focus preferences on word attention weights [17]. In order to make the weight allocation of the attention mechanism more reasonable, this paper embeds the unique identifier of user u into the representation vector as the user's focus preference vector. For the i-th word in the i th project title, its attention weight can be expressed as α_i, and the calculation process is described as follows:

$$\alpha_i = \boldsymbol{q}_u^T \tan\left(\boldsymbol{W}_w \boldsymbol{x}_i^t + \boldsymbol{b}_w\right) \quad (12)$$

$$\alpha_i = \frac{\exp\left(\alpha_i^{\ t}\right)}{\sum_{j=1}^{L} \exp\left(\alpha_j^{\ t}\right)} \quad (13)$$

Among them, α_i is the attention weight of the i-th word in the word level attention mechanism. $\boldsymbol{W}_w$ and $\boldsymbol{b}_w$ are trainable parameters. $\boldsymbol{r}_i^t$ is the final representation of

the i-th project title. It can be expressed as a weighted sum of contextual representations:

$$\boldsymbol{r}_i^t = \sum_{i=1}^{L} \alpha_i \boldsymbol{x}_i \quad (14)$$

From a theoretical and logical perspective, the article clearly points out that previous attention mechanisms did not consider the impact of user focus preferences on word attention weights. However, this article embeds the user's unique identifier into the representation vector as the focus preference vector. This design is based on a reasonable understanding of the correlation between user personalized needs and the importance of title words. In this way, the allocation of attention weights is more in line with the actual situation, and theoretically, it can be reasonably expected to improve the accuracy of the model in extracting title features, thereby enhancing overall performance. This theoretical rationality can support its effectiveness to a certain extent without the need for ablation research; From a practical application perspective, if a personalized word level attention mechanism combined with user embedding is introduced in actual application scenarios, the key indicators of digital media platforms in personalized recommendation tasks, such as user click through rate, dwell time, interaction frequency, etc., are significantly improved. This directly indicates that the mechanism has played a positive role in practical applications, effectively improving system performance, and there is no need to conduct specialized research on data processing to verify its effectiveness.

2.1.2 Category feature extraction

Extracting features of project category information helps generate more accurate project representations. The category feature extraction module mirrors thearchitecture of the title feature extraction module, which uses category embedding, improved multi head self attention mechanism, and word level attention mechanism to learn the representation of items from category information. The category of the i item can ultimately be represented as $\boldsymbol{r}_i^c$.

This article uses a feature fusion module to fuse the learned title features and category features to generate the final unified representation of the project [18]. The calculation process of the title information weight α_i^t for the $\boldsymbol{r}_i^t$ input is as follows:

$$\alpha_i^t = \boldsymbol{q}_u^T \tan\left(\boldsymbol{W}_n \boldsymbol{r}_i^t + \boldsymbol{b}_n\right) \quad (15)$$

$$\alpha_i^t = \frac{\exp\left(\alpha_i^t\right)}{\exp\left(\alpha_i^t\right) + \exp\left(\alpha_i^c\right)} \quad (16)$$

Among them, $\boldsymbol{W}_n$ and $\boldsymbol{b}_n$ are trainable parameters. The attention weight α_i^c of category information can be calculated using a similar formula.

The final representation of the i input in the feature fusion module is the weighted sum of title features and category features according to their corresponding weights:

$$\boldsymbol{r}_i = \alpha_i^t \boldsymbol{r}_i^t + \alpha_i^c \boldsymbol{r}_i^c \quad (17)$$

2.2 Time level cyclic attention

BiLSTM and additive attention mechanism are used to construct temporal level cyclic attention. Take the nearest n interactions in the user u interaction sequence as the user's long-term sequence, represented as $s_u = \{i_1, i_2, \cdots, i_n\}$, input it into the project embedding layer, and finally output $\{\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_n\}$.

BiLSTM: Sequential contextual information can be extracted sequentially to learn users' long-term preferences [19]. The hidden state output at each moment is determined by the input $\boldsymbol{v}_t$ at the current moment, the hidden state $\boldsymbol{h}_{t-1}$ at the previous moment, and the memory unit $\boldsymbol{c}_{t-1}$ at the previous moment. At the t time step, the calculation process of unidirectional LSTM is as follows:

$$\boldsymbol{f}_t = \text{sigmoid}\left(\boldsymbol{W}_f \boldsymbol{h}_{t-1} + \boldsymbol{U}_f \boldsymbol{v}_t + \boldsymbol{b}_f\right) \quad (18)$$

$$\boldsymbol{i}_t = \text{sigmoid}\left(\boldsymbol{W}_i \boldsymbol{h}_{t-1} + \boldsymbol{U}_i \boldsymbol{v}_t + \boldsymbol{b}_i\right) \quad (19)$$

$$\tilde{\boldsymbol{c}}_t = \tanh\left(\boldsymbol{W}_c \boldsymbol{h}_{t-1} + \boldsymbol{U}_c \boldsymbol{v}_t + \boldsymbol{b}_c\right) \quad (20)$$

$$\boldsymbol{c}_t = \boldsymbol{f}_t \boldsymbol{h}_{t-1} + \boldsymbol{i}_t \tilde{\boldsymbol{c}}_t \quad (21)$$

$$\boldsymbol{\eta}_t = \text{sigmoid}\left(\boldsymbol{W}_o \boldsymbol{h}_{t-1} + \boldsymbol{U}_o \boldsymbol{v}_t + \boldsymbol{b}_o\right) \quad (22)$$

$$\boldsymbol{h}_t = \boldsymbol{\eta}_t \tanh\left(\boldsymbol{c}_t\right) \quad (23)$$

Among them, $\boldsymbol{f}_t$ is the state vector of the forget gate at t time, which can selectively retain or forget information from previous states. $\boldsymbol{i}_t$ is the state vector of the input gate at t time, $\tilde{\boldsymbol{c}}_t$ is the candidate value of the memory unit at t time, $\boldsymbol{\eta}_t$ is the state vector of the output gate at t time, $\boldsymbol{c}_t$ represents the memory unit, and the input gate and forget gate jointly control its state vector. $\boldsymbol{h}_t$ is the output state vector of the unidirectional LSTM at t time, $\boldsymbol{W}$ and $\boldsymbol{U}$ are trainable parameters, $\boldsymbol{b}$ is the corresponding bias term.

For the i_t project in time step t, the hidden state $\boldsymbol{h}_t^i$ output by BiLSTM can be expressed as:

$$\boldsymbol{h}_t^i = \vec{\boldsymbol{h}}_t^i \oplus \overleftarrow{\boldsymbol{h}}_t^i \quad (24)$$

Among them, $\vec{\boldsymbol{h}}_t^{\,i}$ is the hidden state of the forward LSTM at t time in BiLSTM, $\overleftarrow{\boldsymbol{h}}_t^{\,i}$ is the hidden state of the backward LSTM at t time in BiLSTM network [20]. The final output of BiLSTM is a sequence $\left\{\boldsymbol{h}_1^{\,t}, \boldsymbol{h}_2^{\,t}, \cdots, \boldsymbol{h}_n^{\,t}\right\}$ with the same length as the long-term interaction sequence.

Additive attention mechanism: The attention weight of the item that interacts with user u at time step t is represented as α_t^l, which can represent the importance of the interaction between the user's long-term preferences and the item representation. The calculation process is as follows:

$$\alpha_t^l = \boldsymbol{q}_u^T \tanh\left(\boldsymbol{W}_l \boldsymbol{h}_t + \boldsymbol{b}_l\right) \quad (25)$$

$$\alpha_t^l = \frac{\exp\left(\alpha_t^l\right)}{\sum_{i=1}^{n} \exp\left(\alpha_t^l\right)} \quad (26)$$

Among them, $\boldsymbol{h}_t$ is the input item i_t at the t moment, while $\boldsymbol{W}_l$ and $\boldsymbol{b}_l$ are trainable parameters in the additive attention mechanism. The final long-term focus preference $\boldsymbol{l}_u$ of user u is the sum of attention weighted item context representations, calculated using the following formula:

$$\boldsymbol{l}_u = \sum_{i=1}^{n} \alpha_t^l \boldsymbol{h}_t \quad (27)$$

Take the output of the last k moments of BiLSTM as the user's recent interaction sequence, represented as $\left\{\boldsymbol{h}_{n-k+1}, \boldsymbol{h}_{n-k}, \cdots, \boldsymbol{h}_n\right\}$. For BiLSTM, the current project or hidden state is always more important than the previous one, which greatly damages the modeling of users' recent focus preferences [21]. The recent focus preference learning aims to model the recent sequence between hidden states of BiLSTM through a recurrent CNN that integrates additive attention mechanism.

Recurrent CNN: concatenate recent user interaction sequences into a $k \times d$ vector matrix $\boldsymbol{H}$, where $\boldsymbol{h}_i \in R^d$ is the i th row of the matrix $\boldsymbol{H}$. Using the matrix $\boldsymbol{H}$ as the input of the recurrent CNN, $\hat{\boldsymbol{F}} \in R^{h \times d}$ is a filter used by the pooling layer to extract local features of the user, where h is the height of the filter, and the convolution operation can be expressed as:

$$y_i = f\left(\hat{\boldsymbol{F}} \cdot \boldsymbol{h}_{i:i+h-1} + b\right) \quad (28)$$

Among them, f is the activation function. b is a bias term. y_i is a feature generated by convolving the specified region $\boldsymbol{h}_{i:i+h-1}$ in the matrix $\boldsymbol{H}$ using a filter with a height of h and a width of d. Use this filter to convolve all regions in the matrix $\boldsymbol{H}$, and finally represent them as $\boldsymbol{y} = \left[y_1, y_2, \cdots, y_{k-h+1}\right]$, where $\boldsymbol{y} \in R^{k\text{-}h+1}$. Then perform max pooling operation to obtain the convolution result $\hat{y}_h = \max\left\{\boldsymbol{y}\right\}$ when the filter height is h.

This article uses x filters of different heights to extract recent local features of users at different levels from the hidden state sequence output at the last k time steps in BiLSTM [22]. The final output of the recurrent CNN is $\left\{\hat{y}_1, \hat{y}_2, \cdots, \hat{y}_x\right\}$.

Additive attention mechanism: In order to fuse the different features extracted by filters of different heights in the recurrent CNN into a unified representation of the combined features, an additive attention mechanism is introduced to process them:

$$\alpha_i^s = \boldsymbol{q}_u^T \tanh\left(\boldsymbol{W}_h \hat{y}_i + \boldsymbol{b}_n\right) \quad (29)$$

$$\alpha_i^s = \frac{\exp\left(\alpha_t^l\right)}{\sum_{j=1}^{x} \exp\left(\alpha_t^l\right)} \quad (30)$$

Among them, α_i^s is the focus weight of the i project. $\boldsymbol{W}_h$ and $\boldsymbol{b}_n$ are trainable parameters. The final representation of user u's recent preferences, $\boldsymbol{g}_u$ is the weighted sum of its contextual item representations:

$$\boldsymbol{g}_u = \sum_{i=1}^{x} \alpha_i^s \hat{y}_i \quad (31)$$

For user u, their long-term focus preferences and recent focus preferences have different degrees of importance in different contexts [23]. The focus preference dynamic fusion module aims to generate the user's overall focus preference by dynamically fusing the user's long-term focus preference and recent focus preference. The specific fusion process can be described as follows:

$$\lambda_u = \text{sigmoid}\left(\boldsymbol{W}_u \boldsymbol{q}_u + \boldsymbol{W}_l \boldsymbol{l}_u + \boldsymbol{W}_g \boldsymbol{g}_u + \boldsymbol{b}_u\right) \quad (32)$$

$$\boldsymbol{F}_u = \left(1 - \lambda_u\right) \boldsymbol{l}_u + \lambda_u \boldsymbol{g}_u \quad (33)$$

Among them, $\boldsymbol{W}_u$, $\boldsymbol{W}_l$, and $\boldsymbol{W}_g$ are trainable parameters. $\boldsymbol{l}_u$ is the user's long-term focus preference. $\boldsymbol{g}_u$ is the user's recent focus preference. $\boldsymbol{b}_u$ is the bias term. λ_u controls the weight of users' recent preferences. The overall user focus preference for the final output is $\boldsymbol{F}_u$.

α in formulas (32) - (33) is a key parameter that controls the weight of users' recent preferences. Its training process is integrated into the overall model training process and optimized and adjusted based on the loss function through backpropagation algorithm. During training, the model continuously updates all trainable parameters based on a large amount of user interaction data to minimize prediction errors and other objectives, and α is also updated synchronously. Regarding the setting of α, it is personalized, and different users have different focuses on long-term and short-term preferences due to differences in behavior patterns and interest preferences. The model learns exclusive α values for each user to accurately generate overall focus preferences that fit individual characteristics and improve personalized recommendation effectiveness.

Although standard sequence models such as Gated Recurrent Unit (GRU) and Transformer-XL have shown excellent performance in sequence modeling, the use of Recurrent Convolutional Neural Network (Recurrent CNN) for recent preference modeling in this paper has its unique considerations. Although GRU and other recurrent neural networks are good at processing sequential data, they have certain limitations in capturing local features. They mainly rely on the sequential information of the sequence and are not deep enough in mining local patterns. Recurrent CNN, through convolution operations, can accurately extract multi-level local features from recent user interaction sequences using filters of different heights, effectively capturing local patterns and key information in the sequence, which is crucial for modeling recent preferences. Although Transformer-XL has significant advantages in long sequence processing due to its ability to model long-range dependencies, its computational complexity is high. When dealing with relatively short recent interaction sequences, the computational overhead and performance improvement brought by its complex architecture are not proportional. In contrast, recurrent CNN has a relatively simple structure, high computational efficiency, and can quickly process recent interactive data while ensuring certain performance, which is more in line with real-time requirements in practical applications. Therefore, choosing recurrent CNN for recent preference modeling is more appropriate.

Hierarchical attention networks use a layered architecture to deeply analyze user interaction sequences from both content and time dimensions, in order to accurately capture user behavior patterns. However, in dealing with noisy or incomplete user interaction data, compared with methods such as projection lag synchronization based on output feedback controllers for uncertain chaotic systems with input nonlinearity, these methods have unique advantages in dealing with the uncertainty of nonlinear systems. Hierarchical attention networks currently mainly focus on modeling from the perspective of feature extraction and fusion of data itself. Although there is no specific processing module designed for noisy or incomplete data, the self attention mechanism can enhance the robustness of noisy data to a certain extent by mining the intrinsic correlations of data and capturing temporal features through cyclic attention. For incomplete data, some missing content can also be inferred through contextual information. Further research can be conducted to combine the ideas of dealing with uncertainty in relevant nonlinear system control methods, Enhance the ability to handle complex noise or incomplete data.

3 Attention based algorithm for predicting attention focus of digital media users

3.1 Optimization of multi head attention mechanism based on transformer

Traditional RNNs and their variants, due to their strong dependence on user sequences, cannot achieve parallel computing, resulting in low efficiency. In contrast, Transformer can leverage the advantages of parallel computing and global modeling to process input features and enhance the representation of user sequence features [24]. Therefore, this article adopts Transformer to optimize the multi head attention mechanism, and the model diagram is shown in Figure 1.

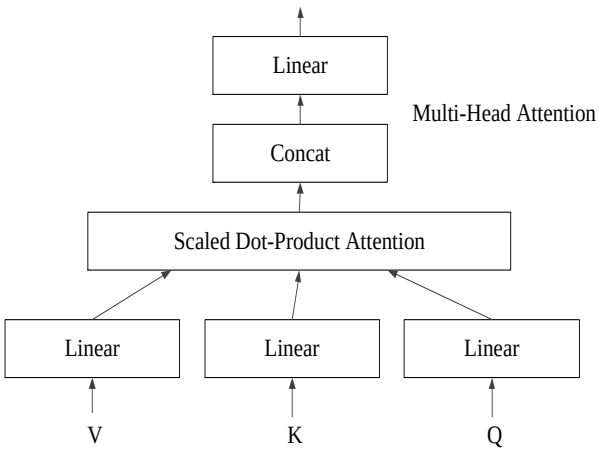


Figure 1: Optimization diagram of multi head attention mechanism based on Transformer

The process of enhancing interest features is as follows:

Step 1: Introduce positional embedding. Time provides basic information about user behavior patterns and their evolution in historical sequences, and multi head attention can understand and utilize positional relationships in sequences.

$$PE_{(pos,2i)} = \sin\left(\frac{pos}{10000^{\frac{2i}{k_e}}}\right) \quad (34)$$

$$PE_{(pos,2i+1)} = \cos\left(\frac{pos}{10000^{\frac{2i}{k_e}}}\right) \quad (35)$$

Add the embedding vector and the positional embedding vector to obtain the input:

$$\breve{E} = E_{bf} \oplus E_p = \left[e_{bf}^1 + e_p^1, \cdots, e_{bf}^M + e_p^M \right] \quad (36)$$

Among them, pos is location. i is the dimension of the embedding vector at position i. E_p is a positional embedding vector.

Step 2: Use a multi head attention network to capture the dependency relationships between features in parallel. Because multi head attention networks can focus on various parts in long sequences and learn multiple subspaces with different representations [25]. Therefore, the input features are linearly transformed into corresponding Q, K, V values, and attention scores are obtained through dot product:

$$head_i = soft\max\left(\frac{\left(\breve{E}W^Q\right)\left(\breve{E}W^K\right)^T}{\sqrt{k_e}} \right)\breve{E}W^V \quad (37)$$

Step 3: Connect the output and input features of each head to obtain different interest feature information.

$$H = concat\left(head_1, head_{2,}, \cdots, head_{c,}\right)W^O \quad (38)$$

$$\hat{H} = layernorm\left(H \oplus \breve{E}\right) \quad (39)$$

Among them, $W^Q \in R^{k_e \times k_e}$, $W^K \in R^{k_e \times k_e}$, $W^V \in R^{k_e \times k_e}$, and $W^O \in R^{k_e \times k_e}$ are the corresponding parameter matrices. $\sqrt{k_e}$ is an acceleration factor. $head_1$ is the output of the i header. c is the number of heads. BVDGS is a positional addition. $\oplus$ is layer normalization. Its function is to accelerate network training and enhance the network's expressive ability.

Step 4: Use a feedforward neural network to enhance the model's ability to represent complex interest features in a range of user behaviors

$$F = \max\left(0, \hat{H}W_1 + b_1\right)W_2 + b_2 \quad (40)$$

Among them, W_1, W_2, b_1, and b_2 are parameters of the neural network.

Obtain feature representations with enhanced interest:

$$Y = layernorm\left(\hat{H} \oplus F\right) \quad (41)$$

3.2 Prediction of attention focus for digital media users based on attention mechanism

The user attention focus prediction layer is used to predict the probability of users clicking on each candidate item [26]. Firstly, input the candidate projects into the project embedding layer to generate the representation $\boldsymbol{r}_{cd}$ of the candidate projects. Secondly, input the interaction sequence of user u into Rec to generate the overall preference $\boldsymbol{F}u$ for that user. The probability of a candidate project being clicked is calculated by the inner product of $\boldsymbol{F}u$ and $\boldsymbol{r}_{cd}$, and the calculation process is as follows:

$$\hat{p}_n = \boldsymbol{F}u \cdot \boldsymbol{r}_{cd} \quad (42)$$

Finally, use the softmax function to normalize all computed $\hat{p}_n$ and generate a recommendation list.

When training the algorithm, the loss function uses a negative logarithmic likelihood function, and the calculation process is as follows:

$$Loss\left(\hat{p}\right) = -\sum_{I=1}^{Z} \hat{p}_I \lg\left(\hat{p}_i\right) + \left(1 - p_i\right)\lg\left(1 - \hat{p}_i\right) \quad (43)$$

Among them, Z is the number of items in the project collection. $p_i = 1$ is a positive sample. The user has already interacted with the project. $p_i = 0$ is a negative sample. The user has not yet interacted with the project.

When extending the loss function (formula 43) to the context of small batches, for each positive sample in the small batch, the loss will be calculated according to the negative logarithmic likelihood function. At the same time, a certain number of negative samples will be selected for each positive sample (non interactive items can be randomly selected as negative samples using negative sampling strategy). The sum of the positive sample loss and the selected negative sample loss will be taken as the loss of the sample in the small batch. Then, the loss of all samples in the small batch will be averaged to update the model parameters; The negative sampling method is used here, not the full softmax, because the full softmax has a high computational cost when the project set is large. Negative sampling can effectively reduce the computational load and ensure the training effect of the model.

4 Experimental analysis

4.1 Experimental setup

In this experiment, to ensure the smooth implementation of model training and evaluation, we used computing resources equipped with high-performance hardware. Specifically, the hardware adopts Intel Xeon series processors and is equipped with NVIDIA Tesla V100 GPU, which has powerful parallel computing capabilities to accelerate the model training process. In terms of memory, 256GB of DDR4 memory is configured to efficiently process large-scale datasets, ensuring stable and efficient completion of data loading, model training, and performance evaluation when running FGAN and various baseline models, providing solid support for the accuracy and reliability of experimental results.

Evaluate the effectiveness of our method using four real datasets, including Frappe, MovieLens, Criteo, and Avazu. These datasets have been used as benchmark datasets for evaluating the model. Divide the dataset into training set (70%), validation set (20%), and testing set

(10%). The basic data of these four datasets are shown in Table 2.

Table 2: Dataset information

Data set	Number of samples	Number of features	Number of users
Frappe	288,609	5,382	957
MovieLens	2,006,859	90,445	17,045
Criteo	45,840,617	2,086,936	39,563
Avazu	40,428,967	1,544,250	80,724

In Table 2, Frappe contains usage records of 957 users in different scenarios, including rich contextual information and a total of 5382 features. MovieLens is a large dataset of 23743 movies rated by 17045 users, with each data containing rich information about users and advertisements. Criteo is an advertising dataset that contains 39 fields. This dataset consists of 39563 users' daily browsing data, with a total of 45840617 samples. Avazu is an advertising display dataset that contains rich contextual information and user statistics. There are a total of 23 domains and 40428967 samples.

Adam is used as the optimizer with a learning rate set to 0.001. This article sets the batch size of the Frappe and MovieLens datasets to 512, and the batch size of Criteo and Avazu to 128. The number of heads in the multi head attention network is set to 6. To avoid overfitting, this article uses L2 regularization ([$1e^{-5}$, $1e^{-4}$, $1e^{-3}$, $1e^{-2}$, 0.1]) and dropout.

Select the following baseline model as the control group for comparative analysis and validation with the method proposed in this article:

(1) FM uses factorization method to learn second-order interaction models from the embedding vectors of input features.

(2) AFM is an improvement based on the FM model, which solves the problem of FM's inability to distinguish the importance of different second-order interaction features by designing an attention network.

(3) NFM learns high-order interactive serial models of features by combining FM and DNN.

(4) FiBiNET utilizes SeNet to dynamically learn feature importance and bilinear functions to effectively learn fine-grained interaction features, and finally inputs the results into DNN to learn high-order interactions in a nonlinear manner.

(5) AutoInt combines residual networks and multi head self attention mechanisms to learn effective low-level interaction features, and uses low-level interaction features as inputs to the hidden layer to learn higher-order interactions.

(6) DMIN utilizes a multi head attention network to learn more refined representations of historical behavior, constructing a multi interest extraction layer to model and capture potential multiple interests.

(7) Fi GNN achieves domain interaction by learning interaction matrices for each node in GNN.

(8) GraphFM combines FM and GNN to learn feature aggregation, and utilizes DNN to learn higher-order interactions.

(9) GraphFwFM learns causal relationships between domain features by using GNN and models graph representations of users and advertisements using GraphSAGE.

In this experiment, both dropout rate and L2 regularization weights were selected through grid search within a given range [$1e^{-5}$, $1e^{-4}$, $1e^{-3}$, $1e^{-2}$, 0.1] to find the optimal parameter combination on the validation set. During the model selection process, the key evaluation indicators on the validation set are used as the basis to regularly evaluate the performance of the model on the validation set during the training process. When the validation indicators no longer improve or reach the preset number of iterations, the training is stopped. Finally, the model parameters with the best performance on the validation set are selected for testing set evaluation.

4.2 Result analysis

Compare FGAN with the baseline model and record the AUC and Logloss results as shown in Tables 3 and 4.

Table 3: Performance comparison between FGAN and baseline method on Frappe and MovieLens

Method	Frappe		MovieLens	
	AUC	Logloss	AUC	Logloss
FM	0.9277	0.3102	0.9144	0.3259
AFM	0.9376	0.2921	0.9211	0.3158
NFM	0.9495	0.2928	0.9332	0.2911
AutoInt	0.9735	0.2253	0.9477	0.2974
FiBiNET	0.9622	0.2236	0.9347	0.2995
DMIN	0.9698	0.2154	0.9511	0.2964
Fi-GNN	0.9718	0.217	0.9464	0.3117
GraphFM	0.9755	0.2084	0.9485	0.2973
GraphFwFM	0.9823	0.1916	0.9535	0.2884
FGAN	0.9848	0.1723	0.9604	0.2679

Table 4: Performance comparison between FGAN and baseline method on Frappe and MovieLens

Method	Criteo		Avazu	
	AUC	Logloss	AUC	Logloss
FM	0.7502	0.485	0.7699	0.3765
AFM	0.7544	0.482	0.7667	0.378
NFM	0.7561	0.4807	0.7694	0.3764
AutoInt	0.7635	0.4758	0.7707	0.3747
FiBiNET	0.7671	0.4729	0.7721	0.3734
DMIN	0.7667	0.4644	0.7713	0.3682

Fi-GNN	0.7839	0.4609	0.7684	0.3761
GraphFM	0.7837	0.4601	0.7741	0.3772
GraphFwFM	0.7844	0.4559	0.7728	0.375
FGAN	0.7916	0.4516	0.7737	0.3429

Analysis of the experimental results in Tables 3 and 4 shows that our method outperforms the baseline method in both AUC and Logloss metrics, achieving optimal performance at the 0.001 level, validating its effectiveness and demonstrating the critical role of learning causal relationships between users and items in improving predictive ability. Compared with various baseline methods, FM performed poorly on the four datasets because it only used hidden vectors to learn second-order interactions and assigned the same weights, ignoring the importance differences of different interaction features; AFM significantly improves performance by designing attention networks to distinguish the importance of interactive features; NFM utilizes dual interaction pooling and DNN to capture fine-grained interaction features, which has significant improvements compared to shallow methods (FM, AFM); FiBiNET has a higher improvement than NFM at the 0.01 level, but NFM is limited in high-order modeling and capturing complex collaborative information; AutoInt and DMIN combined with multi head attention mechanism to distinguish feature importance perform better than FiBiNET on rich historical behavior datasets, as they can comprehensively understand interest and behavior features and effectively focus on inter element dependencies; Fi GNN and GraphFM showed relatively small improvement in metric values on the Frappe and MoviLens datasets with rich user interest features, but performed better on Criteo and Avazu because they effectively model the relationships between sparse historical data using GNN; Compared with GraphFWFM, our method improved AUC and Logloss metrics by 0.25% and 10.1% respectively on the Frappe dataset, increased AUC from 95.35% to 96.04% and decreased Logloss by 7.1% on the MovieLens dataset, improved AUC and Logloss metrics by 0.92% and 0.94% respectively on the Criteo dataset, and increased AUC and Logloss metrics by 0.1% and 8.6% on the Avazu dataset. Especially in Frappe, MovieLens, and Avazu, the Logloss metrics improved significantly, further demonstrating the importance of our method in predicting the attention focus of digital media users.

Set the interaction between the same fields to 1 and assign numerical values based on attention scores. Use a standard normal distribution to standardize the attention scores in each column, where each interaction field has a corresponding score, and the size of the score indicates the importance between the fields. The specific situation is shown in Figure 2.

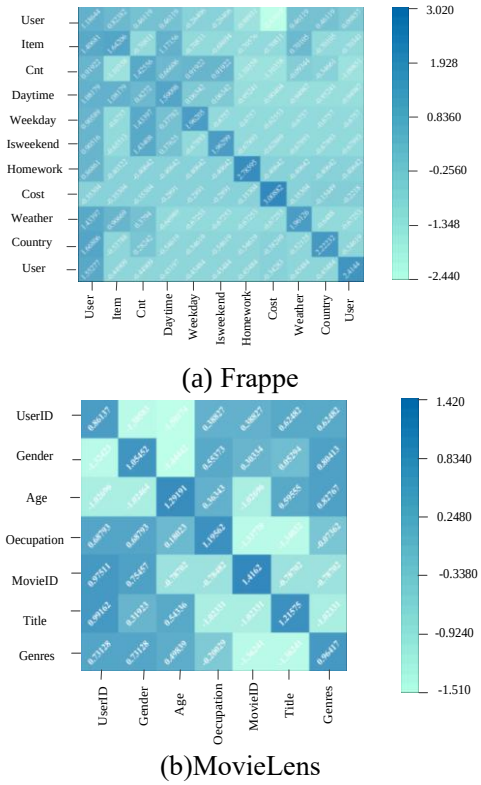
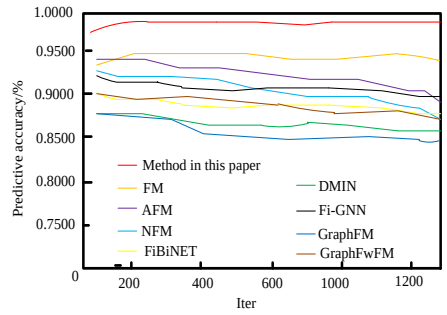


Figure 2: Heat map of the Frappe and MovieLens datasets

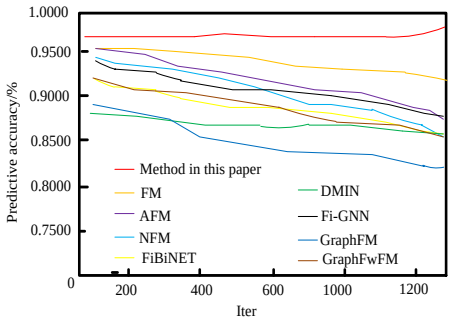
According to the analysis of Figure 2, it can be seen that in Figure 2 (a),<User can be effectively distinguished, Item>, <User, Daytime>, <User, Weekday>, <User, IsWeekend>, <User, Weather>, <User, Country>, <User, City>, <Item, Daytime>, <Item, Weather> Waiting for interactive functions. The visualization of interactive features reasonably reflects the impact of different countries, cities, weather, and time of day on users' clicks on applications. The reason is that users prefer to use the application in the same country, during rest time, and when the weather is good. Therefore, it is crucial to ensure that the right applications are delivered to the right users in the right environment and at the right time. As can be seen from Figure 2 (b)< UserID, MovieID>, <UserID, Title>, <UserID, Genres>, <Genres, Age>, <Genres, Gender> have strong correlations, where UserID and Genres are the user's ID number and movie type, respectively. From the above analysis of interaction characteristics, it can be seen that the user's gender and age determine their preferences. These interactive fields are very valuable because the next project is recommended based on user preferences. The visualization process confirms the efficiency of interactive features and the importance of interest features. And this significantly improves the interpretability and transparency of the method proposed in this article for predicting the attention focus of digital media users, thereby achieving a deeper understanding and analysis of users' interests and behaviors.

Validating the prediction accuracy on four real datasets, Frappe, MovieLens, Criteo, and Avazu, can test whether the model can adapt to various data environments

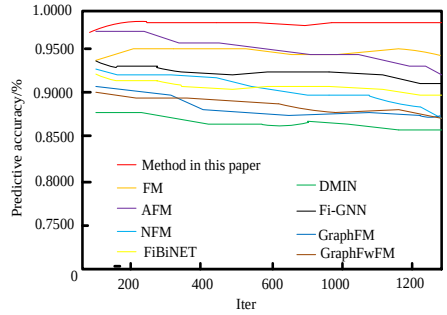
and user behavior characteristics, identify problems in processing different types of data, and verify its ability to predict the attention focus of digital media users in the complex and ever-changing real world. The comparison is shown in Figure 3.



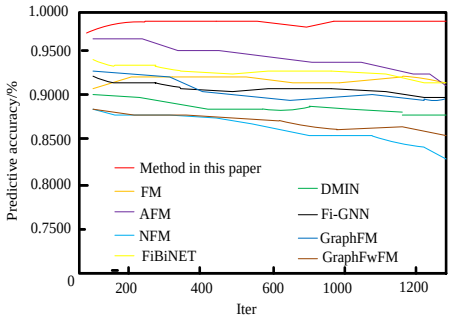
(a) Frappe



(b) MovieLens



(c) Criteo



(d) Avazu

Figure 3: Comparison of prediction accuracies of different methods on the four datasets

From Figure 3, it can be seen that FGAN demonstrates excellent performance on four datasets, with a prediction accuracy consistently above 95% in real scenarios, an average improvement of 15% -20% compared to the comparative method. It can respond in real-time to changes in user interests and provide accurate decision support for personalized content recommendation and platform operation optimization. On the Frappe dataset, the prediction accuracy of other comparison methods varies greatly, but our method stands out and far exceeds the average level. This indicates that the method proposed in this article can effectively handle the application usage record data in specific scenarios contained in the dataset, accurately capturing the user's attention focus in such scenarios. In terms of the MovieLens dataset, it involves a large amount of user rating information on movies, reflecting users' preferences and behavioral characteristics in selecting film and television content. The method presented in this article still performs well, maintaining a high level of prediction accuracy. This indicates that the method proposed in this article has good adaptability to data related to entertainment content, and can deeply analyze users' behavior patterns in the movie selection process, accurately predicting their attention focus. Criteo and Avazu, as advertising datasets, have different data characteristics and application scenarios compared to the previous two. However, the method proposed in this article performs equally well on both datasets. On the Criteo dataset, facing a large amount of advertising browsing data, our method can accurately analyze the interaction behavior between users and advertisements, and accurately predict users' attention tendencies towards different advertisements; On the Avazu dataset, for advertising display data containing rich context and user statistics, our method can also fully explore the key information and achieve high-precision prediction.

Digital media platforms need to filter and push suitable content based on users' real-time behavior and interest preferences in a short period of time, so the running time of verification methods is very important. The specific running time is shown in Figure 4.

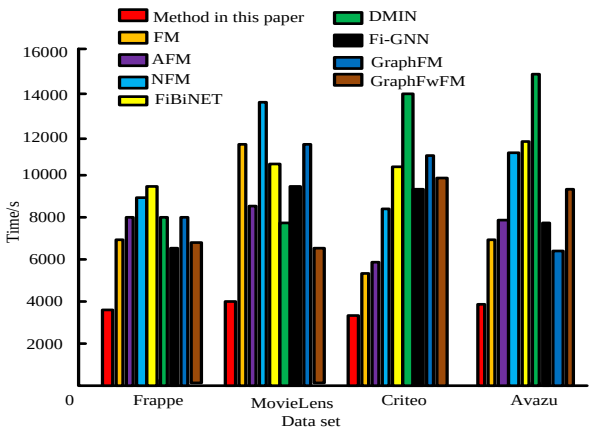


Figure 4: Comparison of running time of different methods on four datasets

Figure 4 shows the running time of each model on four datasets. The running time of our method for inference on all four datasets is around 4000 seconds, which is significantly lower compared to other methods.

The low running time of this method enables it to quickly process large amounts of user behavior data, analyze the trend of changes in user attention focus in real time, and provide accurate personalized recommendations to users in a timely manner. This can not only improve users' satisfaction and loyalty to the platform, but also increase their stay time and interaction frequency on the platform, bringing higher commercial value to the platform. From the perspective of algorithm performance, the low running time of this method is attributed to its innovative architecture design. On the one hand, when constructing a hierarchical attention network, the collaborative work of content level self attention and temporal level recurrent attention can efficiently analyze user interaction sequences from different dimensions in depth. Content level self attention integrates relative positional information through optimized computation, accurately extracts project features, and reduces unnecessary computational redundancy; Time level cyclic attention utilizes BiLSTM and additive attention mechanism to quickly learn users' long-term and recent preferences, and this hierarchical architecture improves the efficiency of information processing. On the other hand, the optimization of multi head attention mechanism based on Transformer utilizes the parallel computing capability of Transformer, greatly accelerating the processing speed of the model for user sequence features. Compared with traditional RNNs and their variants, Transformer can simultaneously process multiple input features, avoiding the time loss caused by sequential computation, and significantly reducing the running time of the model while ensuring prediction accuracy. In summary, the advantages of the method proposed in this article in terms of running time make it more competitive in practical applications, providing strong support for digital media platforms to achieve efficient personalized content recommendation and operational optimization.

The method presented in this article demonstrates excellent performance on four real datasets, with an average prediction accuracy improvement of 15% -20% compared to the comparative methods. This significant improvement makes it highly promising for practical applications such as personalized recommendation. On large digital media platforms, this model, with its low runtime, can quickly process massive user behavior data, respond in real-time to changes in user interests, provide accurate decision support for personalized content recommendation and platform operation optimization, effectively improve user satisfaction, loyalty, stay time, and interaction frequency, and bring higher commercial value to the platform. Compared with simpler adaptive models, although the computational complexity may be slightly higher, the method proposed in this paper has significant advantages in accuracy, reflecting a reasonable balance between accuracy and efficiency. In the field of personalized recommendation, content that meets the user's interests can be accurately pushed based on their real-time attention focus, enhancing the user experience. Although the computational cost is slightly higher compared to some simple models, thanks to innovative architecture designs such as layered attention networks and optimized multi head attention mechanisms, it effectively controls costs while ensuring performance, demonstrating good scalability, and is expected to be widely applied in more digital media scenarios.

In addition, the main purpose of this experiment is to verify the effectiveness and performance advantages of FGAN in predicting the attention focus of digital media users. By comparing it with multiple baseline models on multiple real datasets and comprehensively evaluating indicators such as AUC and Logloss, as well as prediction accuracy and running time, it can intuitively and clearly demonstrate the significant effects of FGAN in improving prediction ability, adapting to different data scenarios, and achieving efficient operation. The results on different datasets are stable and consistent, which is sufficient to support the research conclusions. Therefore, statistical backgrounds such as confidence intervals, standard deviations, and statistical significance tests were not supplemented.

5 Discussion

When compared with the latest methods in relevant work reviews, the model presented in this article demonstrates unique advantages. In the comparison between graph neural network-based methods (such as GraphFwFM) and attention based methods, for the Avazu dataset, our model uses a hierarchical attention network to deeply analyze user interaction sequences from multiple dimensions of content and time, which can more accurately capture user interests. However, GraphFwFM uses GNN to learn causal relationships between domain features, but it is slightly inferior in mining complex multi-dimensional user behavior patterns, so our model performs better; On the Criteo dataset, the advertising browsing data is large and complex. GraphFwFM and other graph neural network methods can effectively model the relationships between sparse historical data. Although our model has excellent performance, its advantages are not as significant as on the Avazu dataset. However, our model is based on Transformer optimized multi head attention mechanism, which has outstanding capabilities in parallel computing and global modeling. With further increase in data volume and complexity, it is expected to show greater advantages.

6 Conclusion

This article proposes a digital media user attention focus prediction algorithm based on attention mechanism. By constructing a hierarchical attention network, integrating content level self attention and temporal level cyclic attention, and optimizing with Transformer based multi head attention mechanism, end-to-end training is achieved. Experiments conducted on four real datasets show that the algorithm outperforms multiple baseline models in AUC and Logloss metrics. The prediction accuracy is improved by an average of 15% -20% compared to the comparison method in real scenarios, and the running time is significantly lower than other methods. It can respond in real-time to changes in user interests and provide accurate

decision support for personalized content recommendation and platform operation optimization.

However, this study also has certain limitations. On the one hand, algorithms assume that user behavior is relatively stable, but in complex digital media environments, user behavior may be affected by various sudden factors and experience significant fluctuations, which may affect the accuracy of predictions. On the other hand, the heads in the Transformer based multi head attention mechanism lack interpretability, making it difficult to determine which key information each head specifically captures. In response to these limitations, future research directions can be explored from the following aspects. Firstly, it can be attempted to integrate users' emotional state information, as emotions have a significant impact on their attention focus, and incorporating them into the model is expected to further improve prediction performance. Secondly, conducting cross platform behavior modeling research, integrating user behavior data from multiple platforms, and more comprehensively characterizing user characteristics, in order to provide users with more accurate attention focus prediction and personalized recommendation services.

Data availability

The raw data supporting the conclusions of this article wi ll be made available by the authors, without undue reserv ation."

Conflicts of interest

The authors declared that they have no conflicts of intere st regarding this work."

Authorship contribution statement

Fenghui Liu conceived the research idea, designed the hierarchical Transformer-based attention prediction framework, conducted the experiments, analyzed the results, and drafted the original manuscript. Jun Huang contributed to data preprocessing and experimental validation, assisted in model optimization and performance evaluation, and participated in the critical review and revision of the manuscript. Both authors have read and approved the final version of the paper.

References

[1] C. Si, “Digital Media Art as Managed Information Enhancing Value in Commercial Spaces,” Information Resources Management Journal (IRMJ), vol. 38, no. 1, pp. 1–14, 2025.

[2] S. D. Pande, P. P. Jadhav, R. Joshi, A.D. Sawant, V. Muddebihalkar, S. Rathod, M.N. Gurav, S. Das, “Digitization of Handwritten Devanagari Text Using CNN Transfer Learning-A Better Customer Service Support,” Neuroscience Informatics, vol. 2, no. 3, p. 100016, 2022. https://doi.org/10.1016/j.neuri.2021.100016

[3] T. J. Schröder and L. Guenther, “Mediating Trust in Content About Science: Assessing Trust Cues in Digital Media Environments,” Public Understanding of Science (Bristol, England), vol. 34, no. 8, pp. 1046–1065, 2025. https://doi.org/10.1177/09636625251337709

[4] Y. Wu and D. Liu, “Investigating the Impact of Augmented Reality Technology on User Engagement and Interaction in Digital Media Environments,” Journal of Computational Methods in Sciences and Engineering, vol. 25, no. 3, pp. 2089–2099, 2025. https://doi.org/10.1177/14727978241309544

[5] Z. Li and J. A. F., “The Influence of Feeling-of-Knowing on Metacognitive Processes in the Digital Media Environment,” Journal of Media Psychology, vol. 37, no. 3, pp. 158–169, 2024. https://doi.org/10.1027/1864-1105/a000426

[6] Y. Zhang, M. Chen, R. Chen, C. Zhao, M. Yuan, and Z. Sun, “Bayesian Attention-Based User Behaviour Modelling for Click-Through Rate Prediction,” CAAI Transactions on Intelligence Technology, vol. 9, no. 5, pp. 1320–1330, 2024. https://doi.org/10.1049/cit2.12343

[7] X. Li, Y. Shan, W. Chen, Y. Wu, P. Hansen, and S. Perrault, “Correction to: Predicting User Visual Attention in Virtual Reality With a Deep Learning Model,” Virtual Reality, vol. 25, no. 4, pp. 1123–1136, 2021. https://doi.org/10.1007/s10055-021-00512-7

[8] G. Yu, Z. Chen, and J. Wu, “Content-Aware Personalized Sharing Based on Cooperative User Selection and Attention in Mobile Internet of Things,” IEEE Transactions on Network and Service Management, vol. 20, no. 1, pp. 521–532, 2023. https://doi.org/10.1109/TNSM.2022.3215656

[9] H. Luo, X. Zhou, H. Ding, and L. Wang, “Multi-Task Deep Learning With Task Attention for Post-Click Conversion Rate Prediction,” Intelligent Automation and Soft Computing, vol. 36, no. 3 Pt. 2, pp. 3583–3593, 2023. https://doi.org/10.32604/iasc.2023.036622

[10] Z. Li, J. Liu, W. Yang, and C. Liu, “Joint Modeling of User and Item Preferences With Interaction Frequency and Attention for Knowledge Graph-Based Recommendation,” Applied Intelligence, vol. 53, no. 22, pp. 26364–26383, 2023. https://doi.org/10.1007/s10489-023-04914-9

[11] Y. Dong, Y. Huang, P. Hu, P. Zhang, and Y. Wang, “The Effect of Picture Attributes of Online Ordering Pages on Visual Attention and User Experience,” International Journal of Industrial Ergonomics, vol. 96, p. 103477, 2023. https://doi.org/10.1016/j.ergon.2023.103477

[12] H. Liu, W. Wang, Q. Peng, N. Wu, F. Wu, and P. Jiao, “Toward Comprehensive User and Item Representations via Three-Tier Attention Network,” ACM Transactions on Information Systems, vol. 39, no. 3, pp. 1–22, 2021. https://doi.org/10.1145/344634

[13] A. Boulkroune, S. Hamel, F. Zouari, A. Boukabou, and A. Ibeas, “Output-Feedback Controller Based Projective Lag-Synchronization of Uncertain

Chaotic Systems in the Presence of Input Nonlinearities," Mathematical Problems in Engineering, vol. 2017, no. 1, p. 8045803, 2017. https://doi.org/10.1155/2017/8045803

[14] A. Boulkroune, F. Zouari, and A. Boubellouta, "Adaptive Fuzzy Control for Practical Fixed-Time Synchronization of Fractional-Order Chaotic Systems," Journal of Vibration and Control, p. 10775463251320258, 2025. https://doi.org/10.1177/10775463251320258

[15] P. Liu, Q. Wang, H. Zhang, J. Mi, and Y. Liu, "A Lightweight Object Detection Algorithm for Remote Sensing Images Based on Attention Mechanism and YOLOv5s," Remote Sensing, vol. 15, no. 9, p. 2429, 2023. https://doi.org/10.3390/rs15092429

[16] N. S. Chauhan and N. Kumar, "Confined Attention Mechanism Enabled Recurrent Neural Network Framework to Improve Traffic Flow Prediction," Engineering Applications of Artificial Intelligence: The International Journal of Intelligent Real-Time Automation, vol. 136, p. 108791, 2024. https://doi.org/10.1016/j.engappai.2024.108791

[17] F. Zouari, K. B. Saad, and M. Benrejeb, "Adaptive Backstepping Control for a Class of Uncertain Single Input Single Output Nonlinear Systems," in 10th International Multi-Conferences on Systems, Signals & Devices 2013 (SSD13), 2013, pp. 1–6. https://doi.org/10.1109/SSD.2013.6564134

[18] F. Zouari, K. B. Saad, and M. Benrejeb, "Robust Neural Adaptive Control for a Class of Uncertain Nonlinear Complex Dynamical Multivariable Systems," International Review on Modelling and Simulations, vol. 5, no. 5, pp. 2075–2103, 2012.

[19] G. Rigatos, M. Abbaszadeh, B. Sari, P. Siano, G. Cuccurullo, and F. Zouari, "Nonlinear Optimal Control for a Gas Compressor Driven by an Induction Motor," Results in Control and Optimization, vol. 11, p. 100226, 2023. https://doi.org/10.1016/j.rico.2023.100226

[20] Y. Xiao, W. K. He, Y. Zhu, and J. Zhu, "A Click-Through Rate Model of E-Commerce Based on User Interest and Temporal Behavior," Expert Systems With Applications, vol. 207, p. 117896, 2022. https://doi.org/10.1016/j.eswa.2022.117896

[21] L. Merazka, F. Zouari, and A. Boulkroune, "High-Gain Observer-Based Adaptive Fuzzy Control for a Class of Multivariable Nonlinear Systems," in 2017 6th International Conference on Systems and Control (ICSC), 2017, pp. 96–102. https://doi.org/10.1109/ICoSC.2017.7958728

[22] W. Zhang, Y. Han, B. Yi, and Z. Zhang, "Click-Through Rate Prediction Model Integrating User Interest and Multi-Head Attention Mechanism," Journal of Big Data, vol. 10, no. 1, p. 11, 2023. https://doi.org/10.1186/s40537-023-00688-6

[23] H. Yang, L. Yao, J. Cai, Y. Wang, and X. Zhao, "A New Interest Extraction Method Based on Multi-Head Attention Mechanism for CTR Prediction," Knowledge and Information Systems, vol. 65, no. 8, pp. 3337–3352, 2023. https://doi.org/10.1007/s10115-023-01867-w

[24] C. Qin, J. Xie, Q. Jiang, and X. Chen, "A Novel Interest Evolution Network Based on Transformer and a Gated Residual for CTR Prediction in Display Advertising," Neural Computing and Applications, vol. 35, pp. 24665–24680, 2023. https://doi.org/10.1007/s00521-023-08349-8

[25] C. Yan, X. Li, Y. Chen, and Y. Zhang, "JointCTR: a Joint CTR Prediction Framework Combining Feature Interaction and Sequential Behavior Learning," Applied Intelligence, vol. 52, pp. 4701–4714, 2022. https://doi.org/10.1007/s10489-021-02678-8

[26] H. Dong and X. Wang, "HoINT: Learning Explicit and Implicit High-Order Feature Interactions for Click-Through Rate Prediction," Neural Processing Letters, vol. 55, no. 1, pp. 401–421, 2023. https://doi.org/10.1007/s11063-022-10889-4