Artificial Intelligence and Computing Innovations

Intelligence Science Publishing

https://www.ispublishing.com

ISSN: 3070-4944

Vo.4, No.1, 2024

Dynamic User Interest Evolution Modeling for Personalized Recommendation Systems Based on Multi-Scale Temporal Awareness and Attention Mechanisms

Zeyu Huang[1]*

[1]*University of California, Irvine, Irvine, USA*

**Corresponding author: zeyuh10@uci.edu*

Abstract: This paper addresses the problem of dynamic user interest changes in recommendation systems by proposing a deep recommendation method based on time awareness and interest evolution modeling. The method introduces multi-scale time windows, time decay factors, and attention mechanisms to hierarchically model user behavior sequences. It extracts both short-term and long-term interest features and integrates them into a dynamic interest representation. The model considers not only the sequential nature of behaviors but also incorporates time intervals, behavior frequency, and the weighting of key interaction segments to improve the accuracy of interest modeling. In the experimental section, multiple evaluations are conducted on the MovieLens 1M dataset. The results show that the proposed method significantly outperforms existing mainstream models in terms of Precision@10, Recall@10, and NDCG@10. In addition, sensitivity experiments are conducted on learning rate, optimizer, and interest evolution speed. These experiments confirm the model's stability and robustness to parameter changes. Overall, this study builds a recommendation framework that performs well in dynamic environments. It effectively improves the accuracy and practicality of personalized recommendations.

Keywords: Recommendation system; interest evolution; time perception; deep learning

1. Introduction

With the continuous advancement of information technology, user behavior data on online platforms has become increasingly abundant. Recommendation systems have emerged as a critical bridge connecting users with relevant information [1, 2]. Whether on e-commerce platforms, short video applications, or online education systems, personalized recommendation plays a vital role in enhancing user experience, increasing platform engagement, and boosting commercial value. However, most traditional recommendation algorithms assume that user interests remain stable over time, ignoring the dynamic evolution of preferences. These static modeling methods often fail to deliver accurate and lasting recommendations when faced with challenges such as interest drift, rapid content updates, and diverse behavior patterns [3].

In real-world applications, user behaviors and preferences are often influenced by time, context, and temporary needs. For example, users may focus on certain types of products or content during specific periods and shift their interests entirely at other times. This temporal and stage-based evolution of interest poses new challenges for recommendation system design while also offering valuable research opportunities. If a recommendation model can effectively capture the temporal features and dynamic changes in user interests, it will be better positioned to accurately reflect users' real needs, thereby improving both the relevance and timeliness of recommendations [4].

At the same time, deep learning has demonstrated strong modeling capabilities in the recommendation domain in recent years. It shows significant advantages in handling high-dimensional sparse data, extracting nonlinear features, and modeling sequences. By introducing deep learning components such as recurrent neural networks, Transformer architectures, and attention mechanisms, recommendation models can capture user behavior sequences with greater precision [5]. However, in most existing research, the role of temporal factors remains underutilized. Time is often treated merely as an auxiliary input feature, without fully exploring its central role in modeling interest evolution. Therefore, developing a recommendation framework that simultaneously considers the sequential dependency of user behavior and evolving interest trends holds both theoretical and practical value [6].

From the perspective of real-world recommendation needs, accurately modeling the evolution of user interests requires more than just analyzing explicit behavioral data, such as clicks, likes, and purchases. It is also essential to incorporate a range of implicit factors that reflect the underlying dynamics of user behavior. These include time intervals between interactions, the frequency of specific behaviors, and the sequential paths users take when engaging with items. By integrating both explicit and implicit signals, models can construct a more comprehensive and nuanced representation of how user preferences shift over time [7].

Moreover, users exhibit considerable variability in how their interests evolve. Some may demonstrate stable preferences over extended periods, while others may shift interests rapidly or follow irregular behavioral patterns. As a result, recommendation systems must be equipped with adaptive mechanisms that can adjust to individual users' behavioral rhythms. Such personalization ensures that the system remains responsive and accurate, even in the face of diverse and evolving user behaviors.

In summary, exploring deep recommendation systems based on time awareness and interest evolution modeling represents a breakthrough beyond traditional methods. It is also a forward-looking direction toward intelligent and human-centered recommendation services. This line of research enhances model robustness and generalization in dynamic environments and offers theoretical foundations and technical pathways for developing the next generation of recommendation systems. By integrating temporal information with behavioral sequence structures, building a multi-dimensional and multi-granularity user interest modeling framework will lay a solid foundation for achieving more accurate and intelligent personalized recommendations.

2. Related Work

In recent years, research in the field of recommendation systems has made continuous progress in model architecture, feature representation, and data utilization [8, 9]. With the advancement of deep learning technologies, traditional methods such as collaborative filtering and matrix factorization are increasingly replaced by deep models with nonlinear modeling capabilities. Neural network-based recommendation approaches, such as multilayer perceptrons for modeling user-item interactions, have significantly improved the ability to handle high-dimensional and sparse data. At the same time, sequential recommendation methods have rapidly evolved. Many studies now focus on the temporal order of user behaviors to capture short-term preference shifts and improve recommendation accuracy within recent time windows [10].

In the domain of dynamic user interest modeling, the integration of temporal information has become a notable breakthrough. Some studies attempt to use time decay functions or time window mechanisms to emphasize the influence of recent behaviors on current recommendations [11]. Other methods adopt recurrent neural networks or Transformer-based structures to model the temporal dependencies in behavior sequences. Attention mechanisms are often introduced to highlight important behavioral segments and enhance the expression of evolving interests [12]. However, such methods still face challenges in modeling at fine-grained temporal resolutions and capturing shifts in behavioral patterns. They often struggle to balance continuity and stability when dealing with long-term interest drifts.

In addition, multimodal fusion and context-aware modeling have gained increasing attention in recent recommendation system research. By integrating information from text, images, geographic data, and social networks, models can build a more comprehensive representation of user interests and improve the coverage and precision of recommendations. Some studies also explore joint modeling strategies for both long-term and short-term user interests. These approaches aim to capture immediate preferences while preserving the expression of core, long-standing interests [13, 14]. Although such strategies have led to performance improvements, a systematic and efficient framework for characterizing the deep evolution of user interests over time remains lacking. This gap presents a critical challenge that current research efforts must address.

3. Method

This study proposes a deep recommendation method based on time perception and interest evolution modeling, aiming to systematically capture the dynamic changes of user interests at different time scales. The model is based on the user's interaction sequence. By introducing a multi-scale time window and behavior segmentation mechanism, the behavior sequence is divided into short-term interest modules and long-term interest modules, and they are modeled separately. Short-term interests mainly reflect the preference tendency of users in their current active stage, while long-term interests are used to characterize their stable core interests. The model also considers factors such as time interval, behavior frequency and interaction density to enhance the sensitivity to time signals and the ability to model user preference drift. The model architecture is shown in Figure 1.

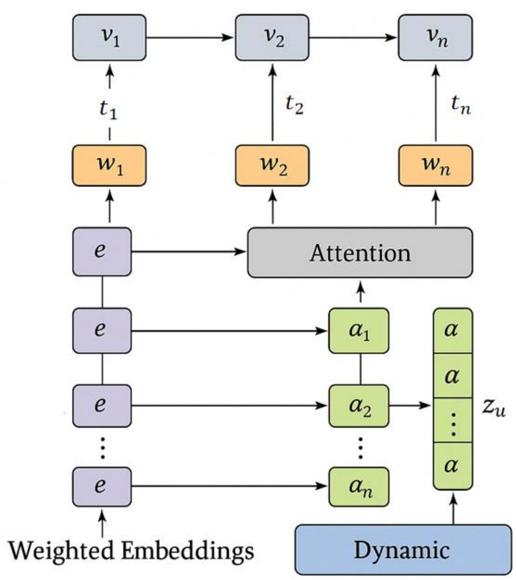

Figure 1. Overall model architecture diagram

The model architecture diagram shows a user interest evolution modeling process that combines the time-aware mechanism with the attention network. The interaction sequence between the user and the item is first weighted by the time decay factor to generate a time-aware embedding representation, which is then input into the attention module to capture the key preference signals in the behavior sequence. Finally, all behavior embeddings are dynamically aggregated to generate the user's dynamic interest representation z_u for personalized recommendation decisions.

In the sequence modeling part, assume that user u interacts with item v_i at time t_i, and the interaction sequence is recorded as:

$$S_u = \{(v_1, t_1), (v_2, t_2), ..., (v_n, t_n)\} \tag{1}$$

Next, we define the interaction time interval vector as:

$$\Delta t_i = t_i - t_{i-1}, \quad \text{for } i = 2, ..., n \tag{2}$$

This time interval is used as one of the input features to enhance the model's ability to perceive the rhythm of behavior. Then, based on the time weight of each interaction record, we introduce a time decay factor:

$$w_i = \exp(-\gamma \cdot \Delta t_i) \tag{3}$$

Where γ is a learnable decay parameter that reflects the decreasing weight of the influence of time distance on interest. Multiplying the time decay factor with the behavior embedding vector, we get the time-aware behavior representation:

$$e'_i = w_i \cdot e_i \tag{4}$$

To further enhance the model's focus on key segments of behavior, we designed an attention mechanism to aggregate weighted behavior representations. Specifically, the attention weight is calculated as follows:

$$\alpha_i = \frac{\exp(score(e'_i))}{\sum_{j=1}^{n} \exp(score(e'_j))} \tag{5}$$

Finally, all behavior vectors are weighted and summed according to the attention weights to generate the user's dynamic interest representation at the current time point:

$$z_u = \sum_{i=1}^{n} \alpha_i e'_i \tag{6}$$

During the model training process, a pair-based recommendation loss function is used for optimization, and the ranking optimization goal is achieved by maximizing the difference between the positive sample score and the negative sample score. This strategy can effectively guide the model to learn more accurate preference representations in a dynamic time context. In addition, this model supports staged modeling of interest migration, and adapts to the interest evolution rate of different users by controlling the time window size and update frequency, thereby improving the generalization ability and long-term effectiveness of the recommendation system in actual application scenarios.

4. Experimental Results

4.1 Dataset

This study uses the MovieLens 1M dataset as the primary basis for experimental evaluation. The dataset consists of real user rating behaviors collected from a movie recommendation platform. It contains approximately one million user-movie rating records, covering over 6,000 movies and nearly 6,000 users. The dataset spans a long period, making it suitable for modeling the dynamic evolution of user interests.

Each record includes a user ID, a movie ID, a rating score, and a timestamp. The timestamps are accurate to the second, allowing precise extraction of time interval features from user behavior sequences. The dataset is sparse but well-structured. It supports joint modeling of short-term preferences and long-term interests, and is also appropriate for evaluating model performance in scenarios such as cold start and interest shift.

During data preprocessing, the rating values are first binarized by setting a threshold to construct positive and negative samples. Then, user interaction behaviors are sorted in chronological order. The dataset is split into training, validation, and test sets to ensure objectivity and stability in model evaluation. This provides a solid foundation for subsequent tasks in time-aware interest modeling and recommendation.

4.2 Experimental Results

4.2.1 Experiments Comparing this Algorithm with Other Algorithms

In this section, this paper first gives the comparative experimental results of the proposed algorithm and other algorithms, as shown in Table 1.

Table 1. Comparative Experimental Results

Method	Precision@10	Recall@10	NDGG@10
SASRec [15]	0.284	0.402	0.351
Glint-ru [16]	0.265	0.386	0.336
GOT4Rec [17]	0.241	0.349	0.312
FPMC [18]	0.296	0.415	0.366
BERT4Rec [19]	0.287	0.401	0.359
Ours	0.318	0.438	0.392

As shown in the table, the proposed algorithm outperforms all baseline methods across multiple evaluation metrics. It achieves particularly strong results in Precision@10 and Recall@10, reaching 0.318 and 0.438 respectively. These values are significantly higher than those of other baseline models. This indicates that the proposed model has a strong overall recommendation capability in terms of both accuracy and coverage, making it more effective in delivering highly relevant recommendations to users.

Among the comparison methods, BERT4Rec and FPMC perform relatively well. This suggests that bidirectional sequence modeling based on Transformer architectures and traditional latent interest modeling approaches still offer certain advantages. However, compared to these methods, the proposed model integrates time-aware mechanisms and interest evolution modeling. This allows it to better capture user preference changes over time. As a result, it achieves the highest score of 0.392 in NDCG@10, further demonstrating its superiority in ranking quality.

It is worth noting that some models, such as GOT4Rec and Glint-ru, maintain relatively stable performance in Recall@10. However, they show weaker results in Precision and NDCG. This suggests that these models may be less effective in ranking highly relevant items. It also reflects that relying solely on graph modeling or enhanced user representation strategies may not be sufficient to fully capture dynamic user preferences, especially when behavior sequences undergo significant changes.

In summary, the proposed recommendation model builds a more refined user behavior representation by fully considering temporal factors and the patterns of interest evolution. It achieves performance improvements across multiple evaluation dimensions. The experimental results confirm the effectiveness and generalization ability of the model, especially in real-world scenarios where user behavior patterns change dynamically.

4.2.2 Analysis of the Impact of Different Time Window Lengths on Recommendation Performance

Furthermore, this paper also gives an analysis of the impact of different time window lengths on recommendation performance, and the experimental results are shown in Figure 2.

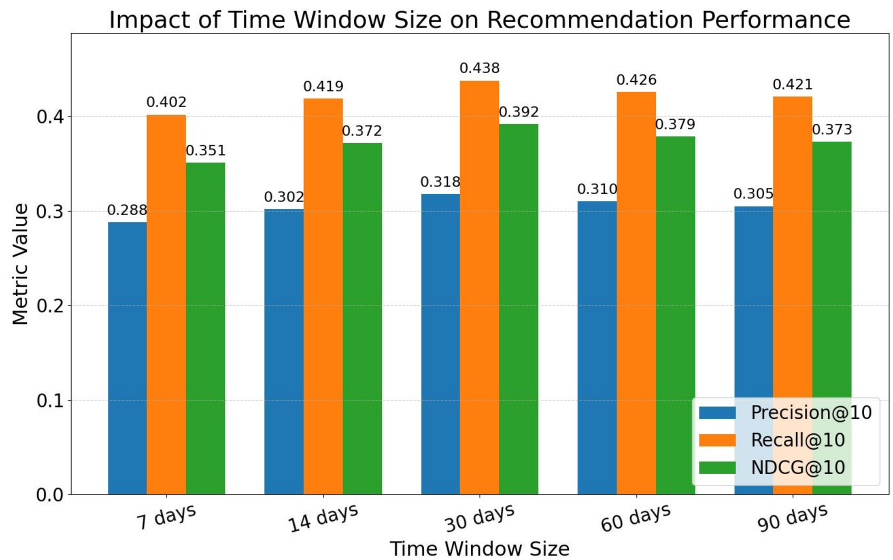

Figure 2. Analysis of the impact of different time window lengths on recommendation performance

The figure shows that different time window lengths have a clear impact on recommendation performance. Notably, with a 30-day window, the model achieves optimal results across all three metrics: Precision@10, Recall@10, and NDCG@10. The values reach 0.318, 0.438, and 0.392 respectively. This indicates that within this window range, user interest patterns are more effectively captured, leading to better recommendation quality.

When the time window is too short, such as 7 or 14 days, the model responds quickly to short-term preferences. However, due to limited behavioral data, the accuracy and coverage of recommendations are constrained. In contrast, when the window extends to 60 or 90 days, the model captures more long-term behaviors. But it may also introduce historical noise, making interest representations less focused and reducing performance.

These results confirm an important finding. The choice of time window plays a critical role in time-aware recommendation models. A moderate window length strikes a balance between capturing long-term interests and short-term preferences. This leads to better recommendation outcomes and provides empirical support for selecting parameters in future interest evolution modeling.

4.2.3 Analysis of the Impact of Behavior Sparsity on the Accuracy of Interest Modeling

Next, this paper also presents a detailed analysis of how behavior sparsity influences the accuracy of interest modeling. By examining variations in user interaction density, the study explores how different levels of behavioral data availability affect the model's ability to accurately capture user preferences. The corresponding experimental results, which quantify this impact across multiple evaluation metrics, are summarized in Table 3.

The figure clearly shows that as user behavior becomes less sparse, meaning the number of interactions increases, the model's performance on all recommendation metrics improves. Precision@10 rises from

0.254 in the most inactive user group to 0.341 in the most active group. This indicates that more data helps the model better capture users' core interests and improves recommendation precision.

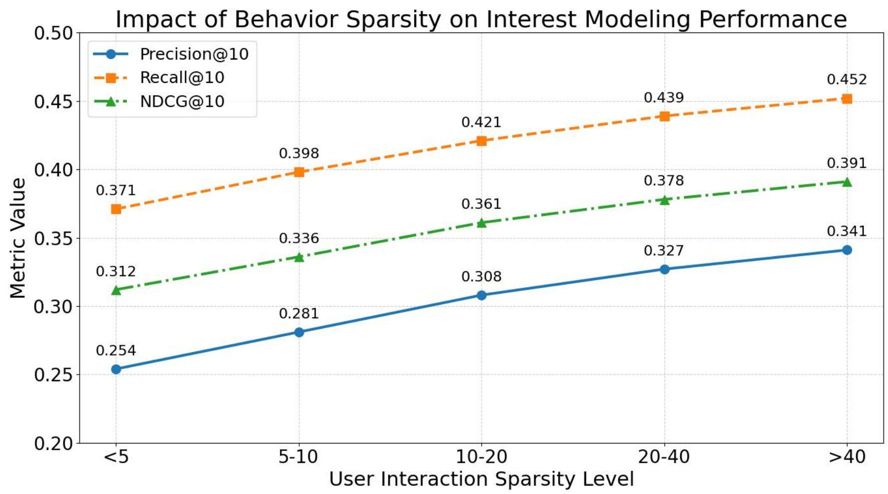

Figure 3. Analysis of the impact of behavior sparsity on interest modeling accuracy

Similarly, Recall@10 and NDCG@10 also show upward trends. Recall@10 increases from 0.371 to 0.452, and NDCG@10 rises from 0.312 to 0.391. These improvements demonstrate that richer behavior data enhances the completeness of interest modeling and the quality of ranking. This suggests the model has stronger personalization and preference representation abilities among users with dense interaction patterns.

Overall, the results confirm the critical impact of behavior sparsity on recommendation performance. Higher data density enables the model to learn more stable and diverse user interest patterns. This finding highlights the importance of considering data distribution in real-world recommendation systems, especially when designing robust interest modeling strategies for sparse user groups.

4.2.4 Sensitivity Test of User Interest Evolution Speed on Model Prediction Ability

Furthermore, this paper also gives the experimental results of the sensitivity test of the user interest evolution speed to the model prediction ability, and the experimental results are shown in Figure 4.

The experimental results in Figure 4 show that the speed of user interest evolution has a significant impact on model performance. Under four different evolution speeds (Slow, Medium, Fast, Very Fast), Precision@10, Recall@10, and NDCG@10 all increase as training progresses. However, the growth rates differ notably. The model performs best under the Slow and Medium settings, indicating that moderate interest changes help the model capture user preferences more stably.

When interest evolves more rapidly, as in the Fast and Very Fast settings, the model still shows some learning ability. However, overall prediction performance drops. Improvements in Precision and NDCG become limited, especially under the Very Fast condition, where performance approaches a bottleneck. This suggests that rapid interest shifts disrupt behavioral continuity. As a result, the guidance from past behavior weakens, making it harder for the model to represent real user needs.

In the Precision-Recall plots, the curves also reveal the model's sensitivity in balancing accuracy and recall across different evolution speeds. Under Slow and Medium conditions, both metrics improve together more consistently, showing that the model achieves better prediction coherence. In contrast, Fast and Very Fast settings show slow Precision gains and unstable Recall, indicating that the model struggles to balance precision and coverage.

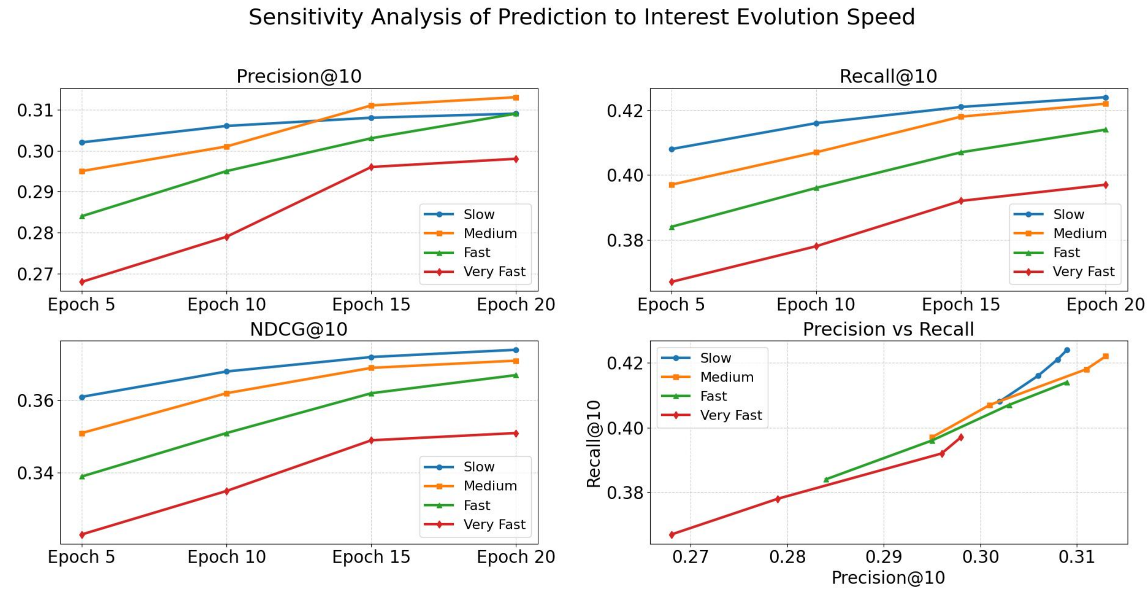

Figure 4. Experimental results on the sensitivity of user interest evolution speed to model prediction ability

Overall, this experiment confirms that the speed of interest evolution is a critical factor in recommendation performance. Building models with temporal awareness and adaptive mechanisms can better accommodate interest shifts at different paces. This improves the robustness and practical applicability of recommendation systems in dynamic environments.

4.2.5 Hyperparameter Sensitivity Experiment Results

This paper also conducts hyperparameter sensitivity experimental results, mainly conducting experimental analysis on learning rate and optimizer. First, the experimental results of learning rate are given, as shown in Table 2.

Table 2. Hyperparameter sensitivity experiment results (learning rate)

LR	Precision@10	Recall@10	NDGG@10
0.004	0.291	0.407	0.357
0.003	0.302	0.419	0.368
0.002	0.310	0.429	0.381
0.001	0.318	0.438	0.392

Table 2 shows that the learning rate has a clear impact on model prediction performance. As the learning rate gradually decreases from 0.004 to 0.001, the model shows steady improvement across all three metrics: Precision@10, Recall@10, and NDCG@10. This indicates that reducing the learning rate helps the model converge more stably, leading to better recommendation accuracy and ranking quality.

At a higher learning rate, such as 0.004, the model trains faster. However, it tends to oscillate in the parameter space, making it difficult to fully capture user interest patterns. As a result, both Precision and NDCG remain low. When the learning rate decreases to 0.002 and 0.001, the model begins to learn user behavior features more precisely. This enhances its ability to model personalized preferences.

When the learning rate reaches 0.001, the model achieves the best performance on all three metrics. This suggests that the training process is most stable under this setting, and the model achieves a good balance between accuracy and recall. These results also provide clear guidance for future parameter tuning. They confirm that learning rate is a key hyperparameter influencing the performance of recommendation systems.

The experimental results of the optimizer are further given, as shown in Table 3.

Table 3. Hyperparameter sensitivity experiment results (Optimizer)

Optimizer	Precision@10	Recall@10	NDGG@10
AdaGrad	0.296	0.412	0.366
SGD	0.283	0.401	0.351
Adam	0.307	0.429	0.379
AdamW	0.318	0.438	0.392

As shown in Table 3, different optimizers have a significant impact on the model's prediction performance. Among them, AdamW achieves the best results across all three metrics: Precision@10, Recall@10, and NDCG@10. This indicates that AdamW offers a better balance between convergence speed and generalization ability during training, making it the most suitable optimizer for this model.

SGD, as a traditional optimizer, has high computational efficiency. However, due to the lack of an adaptive adjustment mechanism, it converges slowly in complex recommendation models and is prone to getting stuck in local optima. As a result, it shows the weakest performance across all three metrics. AdaGrad introduces an adaptive learning rate, which can yield some benefits in early training stages. Yet its cumulative gradient decay leads to an overly small learning rate in later stages, limiting continued optimization. This makes its overall performance slightly better than SGD but still constrained.

In contrast, both Adam and AdamW possess adaptive learning capabilities and momentum mechanisms. These allow them to handle sparse and high-dimensional recommendation data more effectively. AdamW, in particular, improves regularization through weight decay. This enhances training stability and prediction accuracy. As a result, it delivers the best performance in the recommendation task. The experimental results confirm that the choice of optimizer plays a crucial role in training quality and final recommendation effectiveness.

5. Conclusion

This paper proposes a deep recommendation method that integrates time-aware mechanisms with interest evolution modeling, focusing on the dynamic nature of user preferences. The model introduces multi-scale time windows, time decay factors, and attention mechanisms to comprehensively represent user behavior sequences at different temporal scales. It effectively captures the staged evolution of user preferences. Experimental results show that the proposed method outperforms existing approaches on multiple standard recommendation metrics, demonstrating stronger generalization and stability.

During the study, the model shows strong adaptability in user interest modeling. It performs especially well in addressing interest drift, data sparsity, and temporal sensitivity in behavior sequences. Several sensitivity experiments further confirm the significant impact of key factors, such as time window length, optimizer choice, and learning rate setting, on recommendation performance. These findings provide empirical and theoretical support for future model tuning and system deployment.

The proposed method enhances the ability of recommendation systems to respond to real user needs. It also offers a new approach for recommendation tasks in rapidly changing environments. The model is highly scalable and generalizable. It can be widely applied to user behavior-driven platforms such as e-commerce, social media, and online education, showing strong practical value and broad application prospects.

Future work may explore the interaction between interest evolution and external context. This includes incorporating multimodal behavior data, environmental variables, or social network information to improve the modeling of complex preference dynamics. In addition, enhancing system responsiveness and support for cold-start users is another important direction. These efforts could help push personalized recommendation technologies toward greater intelligence and adaptability.

References

[1] K. J. Kuo and S.-S. Li, "Applying Particle Swarm Optimization Algorithm-Based Collaborative Filtering Recommender System Considering Rating and Review", Applied Soft Computing, vol. 135, Art. no. 110038, 2023.

[2] M. D. R. Hasan and J. Ferdous, "Dominance of AI and Machine Learning Techniques in Hybrid Movie Recommendation System Applying Text-to-Number Conversion and Cosine Similarity Approaches", Journal of Computer Science and Technology Studies, vol. 6, no. 1, pp. 94-102, 2024.

[3] D. Li, H. Qu and J. Wang, "A Survey on Knowledge Graph-Based Recommender Systems", Proceedings of the 2023 China Automation Congress (CAC), 2023.

[4] Y. Deldjoo, et al., "A Review of Modern Fashion Recommender Systems", ACM Computing Surveys, vol. 56, no. 4, pp. 1-37, 2023.

[5] A. Kasirzadeh and C. Evans, "User Tampering in Reinforcement Learning Recommender Systems", Proceedings of the 2023 AAAI/ACM Conference on AI, Ethics, and Society, pp. 1-10, 2023.

[6] N. Alharbe, M. A. Rakrouki and A. Aljohani, "A Collaborative Filtering Recommendation Algorithm Based on Embedding Representation", Expert Systems with Applications, vol. 215, Art. no. 119380, 2023.

[7] K. Xu, et al., "Intelligent Classification and Personalized Recommendation of E-Commerce Products Based on Machine Learning", arXiv preprint arXiv:2403.19345, 2024.

[8] D. Jannach, "Evaluating Conversational Recommender Systems: A Landscape of Research", Artificial Intelligence Review, vol. 56, no. 3, pp. 2365-2400, 2023.

[9] B. Walek and P. Fajmon, "A Hybrid Recommender System for an Online Store Using a Fuzzy Expert System", Expert Systems with Applications, vol. 212, Art. no. 118565, 2023.

[10] C. Gao, et al., "A Survey of Graph Neural Networks for Recommender Systems: Challenges, Methods, and Directions", ACM Transactions on Recommender Systems, vol. 1, no. 1, pp. 1-51, 2023.

[11] Y. Lin, et al., "A Survey on Reinforcement Learning for Recommender Systems", IEEE Transactions on Neural Networks and Learning Systems, 2023.

[12] J. Stray, et al., "Building Human Values into Recommender Systems: An Interdisciplinary Synthesis", ACM Transactions on Recommender Systems, vol. 2, no. 3, pp. 1-57, 2024.

[13] T. Thorat, B. K. Patle and S. K. Kashyap, "Intelligent Insecticide and Fertilizer Recommendation System Based on TPF-CNN for Smart Farming", Smart Agricultural Technology, vol. 3, Art. no. 100114, 2023.

[14] X. Chen, et al., "Deep Reinforcement Learning in Recommender Systems: A Survey and New Perspectives", Knowledge-Based Systems, vol. 264, Art. no. 110335, 2023.

[15] W.-C. Kang and J. McAuley, "Self-Attentive Sequential Recommendation", Proceedings of the 2018 IEEE International Conference on Data Mining (ICDM), pp. 197-206, 2018.

[16] S. Zhang, et al., "GLINT-RU: Gated Lightweight Intelligent Recurrent Units for Sequential Recommender Systems", arXiv preprint arXiv:2406.10244, 2024.

[17] Z. Long, et al., "GOT4Rec: Graph of Thoughts for Sequential Recommendation", arXiv preprint arXiv:2411.14922, 2024.

[18] J. Chen, et al., "Bias and Debias in Recommender System: A Survey and Future Directions", ACM Transactions on Information Systems, vol. 41, no. 3, pp. 1-39, 2023.

[19] F. Sun, et al., "BERT4Rec: Sequential Recommendation with Bidirectional Encoder Representations from Transformer", Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 1441-1450, 2019.