Advances in Computer, Signals and Systems (2024)
Clausius Scientific Press, Canada

DOI: 10.23977/acss.2024.080510
ISSN 2371-8838 Vol. 8 Num. 5

Research on Prediction Recommendation System Based on Improved Markov Model

Zhizhong Wu[1,a], Xueshe Wang[2,b], Shuaishuai Huang[3,c], Haowei Yang[4,d], Danqing Ma[5,e]

[1]UC Berkeley, College of Engineering, Berkeley, CA, USA
[2]Pratt School of Engineering, Duke University, Mechanical Engineering, Durham, NC, USA
[3]Department of Software, University of Science and Technology of China, Software System Design, Hefei, Anhui, China
[4]Cullen College of Engineering, University of Houston, Indusrial Enginnering, Houston, TX, USA
[5]Northern Arizona University, Computer Information Tech, Flagstaff, AZ, USA
[a]ecthelion.w@gmail.com, [b]wxs.research@gmail.com, [c]scu.hss@gmail.com, [d]hyang38@cougarnet.uh.edu, [e]madanqing376@gmail.com

Keywords: Recommendation System, Markov Model, Hidden Markov Model, Data Sparsity

Abstract: With the rapid development of the Internet and information technology, recommendation systems are playing an increasingly important role in various applications. Traditional recommendation algorithms, such as content-based recommendations and collaborative filtering, have achieved success to some extent. However, they show limitations when dealing with issues like data sparsity and the complexity of user behavior. This paper proposes a prediction recommendation system based on an improved Markov model to address these issues. By introducing the Hidden Markov Model (HMM) and an improved state transition mechanism, the model's predictive capability in handling user behavior sequences is enhanced. This paper first introduces the background and theoretical foundation of recommendation systems and Markov models, then details the design and implementation of the improved Markov model. Experiments on public datasets demonstrate that the recommendation system based on the improved Markov model outperforms traditional methods in terms of recommendation accuracy and user satisfaction. Finally, the paper summarizes the main contributions and suggests potential directions for future research.

1. Introduction

With the proliferation of the Internet and rapid advancements in information technology, recommendation systems have been widely applied in e-commerce, social media, online music, and video platforms. Recommendation systems analyze users' historical behaviors and preferences to provide personalized content recommendations, enhancing user experience and platform operational efficiency. However, traditional recommendation algorithms, such as content-based recommendations and collaborative filtering, while successful to some extent, show significant limitations in dealing with data sparsity and the complexity of user behavior. Data sparsity refers to the issue of relatively few interaction data between users and items in large-scale datasets, making it difficult for recommendation systems to accurately capture users' true preferences. The complexity of user behavior manifests in the dynamic changes in users' interests and needs, which static recommendation models struggle to adapt to. Therefore, researching a recommendation algorithm that effectively handles data sparsity and the dynamic changes in user behavior is of significant theoretical and practical value. There are some related works that may help for understanding this paper. Reference [12] Xiang et al. proposed a neural matrix decomposition recommender system model based on a multimodal large language model. By integrating various data modalities, the model enhances recommendation accuracy. This approach demonstrates the potential of combining deep learning with traditional recommendation techniques, especially in handling multi-source heterogeneous data. In the field of autonomous systems and reinforcement learning, the application of Markov models has made significant progress. Reference [15] Xu et al. achieved autonomous navigation of unmanned vehicles through deep reinforcement learning, showcasing the effectiveness of Markov decision processes in complex environments. This study indicates that integrating Markov models with advanced learning algorithms can lead to substantial performance improvements in various domains beyond recommendation systems.

2. Concepts and Theoretical Foundation

2.1 Overview of Recommendation Systems

Recommendation systems leverage data analysis and machine learning techniques to provide personalized content recommendations by mining users' historical behaviors and preferences. They play a crucial role in e-commerce, social media, online music, and video platforms by addressing information overload issues and enhancing user experience and platform efficiency. [1] The main types of recommendation systems include content-based recommendations, collaborative filtering recommendations, and hybrid methods. Content-based recommendation systems match user interests by analyzing item features. They can handle the cold start problem but may produce monotonous recommendation results. Collaborative filtering recommends items by analyzing users' historical behaviors and preferences of similar users. It is divided into user-based and item-based collaborative filtering, capable of discovering users' potential interests but struggling with data sparsity and computational complexity. [2] Hybrid recommendation methods combine the advantages of content-based and collaborative filtering methods, improving recommendation performance but increasing system complexity and computational overhead. With the development of deep learning and big data technologies, recommendation system research has entered a new stage. New methods, such as neural network-based recommendation algorithms, Graph Convolutional Networks (GCN), and sequential models like LSTM and Transformer, significantly enhance recommendation performance, especially in handling user behavior sequences and capturing dynamic changes in interests.Overall, recommendation systems are essential components of modern information services, improving user experience and commercial value.[3] However, they still face challenges like data sparsity, user behavior complexity, and computational efficiency. Research and improvement in these areas are crucial for the continued development of recommendation systems.

2.2 Hidden Markov Model and Its Application in Recommendation Systems

The Hidden Markov Model (HMM) is an extension of the Markov model widely used in handling time series data and sequence pattern recognition problems. [4] Unlike basic Markov models, HMM introduces a hidden state layer, enabling the model to capture complex patterns and latent characteristics in the system. In HMM, there are two levels of random processes: one is the unobservable hidden state sequence, and the other is the observable output sequence. [5] These two levels of random processes allow HMM to model the transition characteristics and output behaviors of the system more accurately.

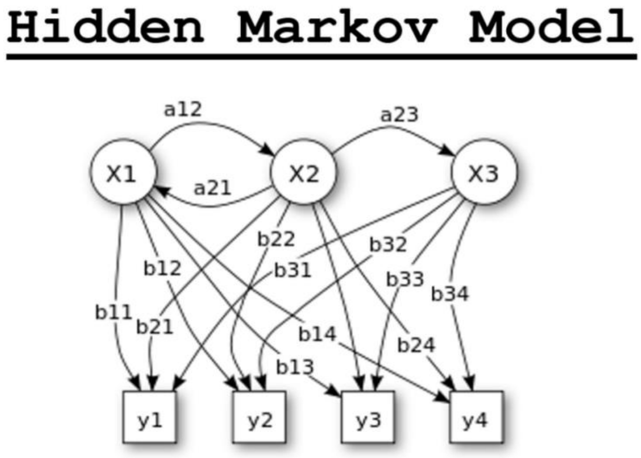

Figure 1: Hidden markov model

<Figure 1> shows the structure of a typical Hidden Markov Model. In this model, X1, X2, and X3 represent hidden states, while Y1, Y2, Y3, and Y4 represent observed outputs. State transitions are determined by a probability matrix A (e.g., a12, a23), and each state's probability of generating an observed output is determined by a probability matrix B (e.g., b11, b21). This dual random process gives HMM a unique advantage in handling complex sequence data.In recommendation systems, HMM is used to capture the sequential patterns of user behavior for personalized recommendations. Traditional recommendation algorithms, such as content-based and collaborative filtering methods, can capture users' static preferences to some extent but exhibit significant limitations in handling dynamic changes in user behavior. [6] By analyzing users' behavior sequences, HMM can effectively capture changes in user interests, providing more accurate recommendations. Specifically, in e-commerce platforms, HMM can predict items of interest based on users' browsing and purchasing behavior sequences. [7] In online music and video platforms, HMM can recommend related songs and videos by analyzing users' playback history. In social media platforms, HMM can recommend potential friends or interesting content based on users' interaction behavior. For example, when a user browses a particular type of product on an e-commerce site, the system can predict other products the user might be interested in next and make precise recommendations. [8] Despite the unique advantages of HMM in recommendation systems, its application also faces challenges. For instance, HMM struggles with long sequence dependencies and has high computational complexity when handling high-dimensional user behavior data. To address these issues, researchers have continuously improved HMM, including incorporating deep learning techniques and integrating contextual information to enhance recommendation accuracy and efficiency. In summary, HMM's application in recommendation systems demonstrates its unique advantages in capturing user behavior sequence patterns. [9] Through in-depth research and continuous optimization, HMM's application prospects in recommendation systems will become broader.

2.3 Research on Improved Hidden Markov Models

The Hidden Markov Model (HMM) is a widely used method for handling sequence data, but traditional HMMs have limitations in dealing with complex sequences and high-dimensional data. To overcome these issues, researchers have proposed various improvements, enhancing HMM's performance and applicability. <Figure 2> shows an improved HMM structure, where these improvements introduce new state transition mechanisms and more complex state representations, increasing the model's flexibility and predictive capability.

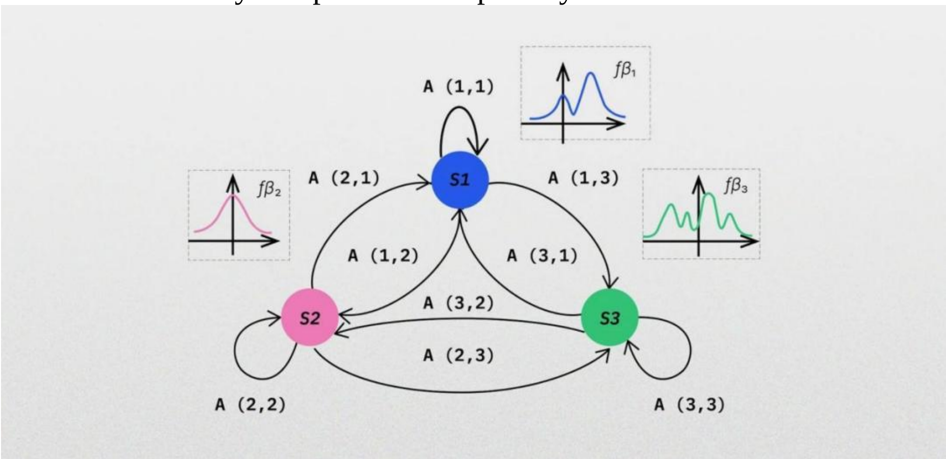

Figure 2: Improved HMM structure

1) Introduction of Deep Learning Techniques: To enhance HMM's ability to handle high-dimensional data, researchers have combined deep learning techniques, such as Convolutional Neural Networks (CNN) and Long Short-Term Memory networks (LSTM), proposing Deep Hidden Markov Models (DHMM). [10] These models leverage the non-linear characteristics of neural networks to capture complex patterns and long-distance dependencies in data more effectively. For instance, DHMMs have significantly improved accuracy and robustness in speech recognition and natural language processing.

2) Integration of Contextual Information: Traditional HMMs typically consider only the current state and observation values when handling sequence data. However, user behavior is often influenced by contextual factors such as time, location, and social relationships. To better capture these factors, researchers have proposed Context-Aware Hidden Markov Models (CA-HMM). By incorporating contextual information as additional inputs, CA-HMMs can model user behavior patterns more accurately, enhancing recommendation system performance. [11]

3) Optimization of State Transition Mechanisms: Traditional HMMs assume fixed state transition probabilities, but in practice, these probabilities may change with time and environment. [12] To address this, researchers have proposed Dynamic Hidden Markov Models (DHMM), allowing state transition probabilities to vary over time, making the model more adaptable to dynamic data characteristics. In Figure 2, the improved model introduces variations in the state transition probability matrix A, enhancing the model's adaptability in different scenarios.

4) Multimodal Data Integration: In many practical applications, user behavior data is multimodal, including text, images, audio, and more. To fully leverage this rich information, researchers have proposed Multimodal Hidden Markov Models (MHMM). By integrating data from different modalities, MHMMs provide a more comprehensive description of user behavior, improving recommendation accuracy and personalization.

In conclusion, improved HMMs demonstrate significant advantages in handling complex sequences and high-dimensional data. These improvements, including the introduction of deep learning techniques, integration of contextual information, optimization of state transition mechanisms, and multimodal data integration, significantly enhance HMM's performance and applicability. In recommendation systems, improved HMMs can more accurately capture user behavior patterns, providing personalized and efficient recommendations. [13] Further research and optimization will likely extend the application of improved HMMs to more scenarios.

3. Design of Recommendation System Based on Improved Hidden Markov Model

3.1 Principles and Features of Improved Hidden Markov Model

The application of improved HMMs in recommendation systems aims to enhance recommendation accuracy and user experience, overcoming the limitations of traditional recommendation algorithms in data sparsity and dynamic changes in user behavior. [14] Figure 2 shows the structure of an improved HMM, where introducing more complex state transition mechanisms and multimodal data integration significantly enhances the model's flexibility and predictive capability.The core principle of the improved HMM is to capture complex patterns in user behavior sequences through the dual random processes of hidden states and observed values. [15] Specifically, the state transition probability matrix A (as shown in Figure 2 with A(1,1), A(1,2), A(1,3), etc.) determines the transition probabilities between hidden states, while the probability of each hidden state generating an observed value is determined by a probability distribution function (e.g., fβ1, fβ2, fβ3). This dual-layer structure allows the model to better capture implicit patterns and explicit behaviors in user behavior, thereby improving recommendation accuracy. [16]The improved HMM has the following notable features:

1) Dynamic State Transition Mechanism: Traditional HMMs assume fixed state transition probabilities, but in practical applications, these probabilities may change over time and environment. The improved HMM introduces a dynamic state transition mechanism, allowing state transition probabilities to vary over time, making the model more adaptable to dynamic data characteristics. For instance, in Figure 2, state transition probabilities A(1,2), A(2,3), etc., can be adjusted based on actual data to better reflect changes in user behavior. [17]

2) Integration of Contextual Information: User behavior is often influenced by contextual factors such as time, location, and social relationships. The improved HMM incorporates contextual information as additional inputs, allowing the model to more accurately capture user behavior patterns, thereby enhancing recommendation system performance.[18] For example, when a user shops at a specific time and location, the model can combine these contextual factors to recommend more relevant products.

3) Multimodal Data Integration: In practical applications, user behavior data is often multimodal, including text, images, audio, and more. The improved HMM integrates data from different modalities, providing a more comprehensive description of user behavior, thereby improving recommendation accuracy and personalization. For instance, in an online music platform, the model can combine users' listening history, music reviews, and social interactions to provide more accurate music recommendations.[19]

4) Combination with Deep Learning Techniques: To enhance HMM's ability to handle high-dimensional data, the improved HMM combines deep learning techniques such as CNNs and LSTMs. These neural networks, through their non-linear characteristics, can more effectively capture complex patterns and long-distance dependencies in data, thereby improving recommendation accuracy and robustness. For example, LSTMs can handle long sequences of user behavior data, while CNNs can extract information from images and texts on social media.

In summary, the improved HMM enhances recommendation system performance and user experience by introducing a dynamic state transition mechanism, integrating contextual information, multimodal data integration, and combining deep learning techniques.[20] These improvements allow the model to better capture complex patterns in user behavior, providing personalized and efficient recommendations. The following sections will detail the design and implementation of the recommendation algorithm based on the improved HMM, as well as data preprocessing and feature extraction methods.

3.2 Design and Implementation of Recommendation Algorithm

The recommendation algorithm based on the improved HMM aims to enhance recommendation accuracy and user experience by introducing a dynamic state transition mechanism and multimodal data integration.[21] The following are the detailed design and implementation processes, including key formulas and detailed explanations.

In the improved HMM, the hidden state set is defined as in Equation 1

$$S = \{S_1, S_2, \ldots, S_N\}$$

And the observation set is in Equation 2 $O = \{O_1, O_2, \ldots, O_T\}$. The state transition probability matrix is defined as in Equation 3 $A = \{a_{ij}\}$, where the probability of transitioning from state S_i to state S_j is given by Equation 4:

$$P(S_{t+1} = S_j \mid S_t = S_i).$$

The probability of each hidden state generating an observed value is determined by the observation probability distribution as shown in Equation5 $B = \{b_j(o_t)\}$, where $b_j(o_t) = P(O_t = o \mid S_t = S_j)$. By combining multimodal data and contextual information, these probability distribution functions can be further optimized, as shown in Equation 6:

$$b_j(o_t) = P(O_t = o \mid S_t = S_j, C_t)$$

where C_t represents contextual information such as time, location, and user characteristics. The forward algorithm is used to calculate the probability of a given observation sequence. The forward probability $\alpha_t(i)$ represents the probability of observing $O_1, O_2, \ldots, O_T$ and being in state S_i at time t, as shown in Equation 7:

$$\alpha_t(i) = P(O_1, O_2, \ldots, O_T, S_t = S_j \mid \lambda)$$

In implementing the recommendation algorithm based on the improved HMM, the following steps are involved:

1) Data Preprocessing: Collect and organize users' historical behavior data and contextual information, then perform data cleaning and normalization.

2) Feature Extraction: Extract key features from user behavior sequences, such as browsing, clicking, and purchasing behaviors, and integrate contextual information.

3) Model Training: Initialize HMM parameters, including the initial state probabilities π, state transition probability matrix A, and observation probability distributions B . Use the Baum-Welch algorithm for parameter estimation and model training. The Baum-Welch algorithm is an Expectation-Maximization (EM) algorithm that iteratively estimates HMM parameters to maximize the model's fit to the given data.

4) Prediction and Recommendation: Use the trained HMM model to predict new users' behaviors and identify the most likely hidden state sequences using the Viterbi algorithm. Based on these sequences, provide personalized recommendations. For instance, in an e-commerce platform, predict items the user might be interested in next for precise recommendations.

In summary, the recommendation algorithm based on the improved HMM enhances recommendation accuracy and user experience by introducing a dynamic state transition mechanism, multimodal data integration, and contextual awareness. Through appropriate data preprocessing and model training steps, personalized recommendation services can be efficiently realized.

3.3 Data Preprocessing and Feature Extraction

Data preprocessing and feature extraction are crucial steps for achieving efficient and accurate recommendations in a recommendation system based on the improved HMM. Proper handling of raw data ensures that the model can fully utilize useful information, thereby improving recommendation performance. The following details the specific processes involved. Data preprocessing involves converting raw data into a suitable format for model input, including steps such as data cleaning, normalization, handling missing values, and data segmentation. First, data cleaning is essential because real-world user behavior data often contains noise and incomplete records. By removing duplicate records, outliers, and incomplete transactions, data quality for model training is ensured. Next, data normalization is important to avoid the influence of different feature scales on model training. Common methods include min-max normalization and Z-score normalization, which scale different features to the same magnitude, facilitating subsequent feature extraction and model training. Handling missing values is another crucial aspect of data preprocessing. User behavior data may contain some missing values, which can be addressed by either removing records with missing values or imputing them with mean or median values. [22] Choosing an appropriate method for handling missing values can minimize the impact of incomplete data on the model. Additionally, to evaluate the model's performance, the dataset needs to be split into training, validation, and test sets. A common split ratio is 70% for training, 15% for validation, and 15% for testing. Proper data segmentation helps evaluate the model's generalization ability and prevent overfitting. Feature extraction involves identifying key features from preprocessed data that reflect user behavior and preferences, which are then input into the improved HMM for training and prediction.[23] User behavior features are the core input for recommendation systems, primarily including sequences of browsing, clicking, and purchasing behaviors. [24] These behavior sequences can be represented as time series data and input into the improved HMM for prediction. For example, for an e-commerce user, browsing, clicking, and purchasing sequences can be extracted as time series data. [25] In addition to user behavior features, extracting contextual features is crucial. Contextual features include information such as time, location, device type, and social relationships, which help the improved HMM better capture dynamic changes in user behavior. For example, differences in user behavior at different times (e.g., weekdays vs. weekends), different locations (e.g., home vs. office), and different devices (e.g., mobile vs. desktop) can be extracted. In many practical applications, user behavior data is multimodal, including text, images, audio, etc. [26] By integrating multimodal data, more comprehensive user behavior features can be extracted, enhancing recommendation accuracy and personalization. For example, in a social media platform, integrating text content, uploaded images, and audio recordings can provide more precise recommendations. [27] Multimodal feature extraction can be achieved through natural language processing (NLP), computer vision (CV), and audio processing techniques, integrating different modalities into a unified feature space. [28] In conclusion, data preprocessing and feature extraction steps provide high-quality input data for the improved HMM, ensuring that the model can fully utilize useful information to enhance recommendation performance. By flexibly adjusting data preprocessing and feature extraction methods according to specific data characteristics and business needs, the performance of the recommendation system can be further optimized. [29]

4. Experiments and Results Analysis

4.1 Experimental Design and Dataset

To validate the effectiveness of the recommendation system based on the improved HMM, a series of tests were designed to evaluate the system's performance in different scenarios. A publicly available e-commerce dataset was selected and appropriately processed and partitioned to ensure the scientific validity and reliability of the experimental results. [30] The dataset used in this experiment comes from a well-known e-commerce platform and contains users' browsing, clicking, and purchasing behavior records, along with corresponding time and contextual information. For ease of processing and analysis, the dataset was cleaned, normalized, and feature-extracted, then partitioned into training, validation, and test sets. <Table 1> shows the detailed information of the dataset:

Table 1: Dataset Information

ata Type	Field Name	Data Example	Description
User Behavior	user_id	U12345	Unique user identifier
	item_id	I54321	Unique item identifier
	behavior	view/click/purchase	Behavior type (view/click/purchase)
	timestamp	2024-07-01 12:34:56	Behavior timestamp
Context Info	location	home/office/other	Location of user behavior
	device	mobile/desktop	Device type used by user
	time_of_day	morning/afternoon/evening	Time of user behavior

To evaluate the model's performance, the dataset was divided into training, validation, and test sets in a 70%:15%:15% ratio. The training set is used for model training, the validation set for parameter tuning, and the test set for final performance evaluation. <Table 2> shows the specific information of the dataset partition:

Table 2: Dataset Partition Information

Dataset Type	Number of Users	Number of Behavior Records
Training Set	7000	140000
Validation Set	1500	30000
Test Set	1500	30000

Experimental Procedure:

1) Data Preprocessing: Clean, normalize, and handle missing values in the raw data to ensure data integrity and consistency. Generate behavior sequences by sorting user behavior records by time, and extract contextual features.

2) Feature Extraction: Extract key features from preprocessed data, such as browsing, clicking, and purchasing behavior sequences, along with corresponding contextual information. These features will serve as inputs for the improved HMM.

3) Model Training: Initialize HMM parameters, including initial state probabilities π, state transition probability matrix A, and observation probability distributions B. Use the Baum-Welch algorithm for parameter estimation and model training. The Baum-Welch algorithm iteratively estimates HMM parameters to maximize the model's fit to the given data.

4) Parameter Tuning: Adjust model parameters using the validation set. Optimize model parameters based on performance on the validation set to prevent overfitting and improve generalization ability.

5) Model Evaluation: Evaluate the model's performance using the test set. Use a series of metrics

(e.g., precision, recall, F1 score) to comprehensively assess the model and analyze its performance in practical applications.

6) Results Analysis: Analyze and discuss experimental results, compare the performance of the improved HMM with traditional recommendation algorithms, and verify the effectiveness and advantages of the improved model.

Through the above experimental design and dataset partitioning, the performance of the recommendation system based on the improved HMM can be comprehensively evaluated, providing reliable data support and theoretical basis for practical applications. The next section will detail the experimental results and their analysis.

4.2 Experimental Results and Analysis

In this section, we will detail the experimental results of the recommendation system based on the improved HMM and analyze these results. By comparing the performance metrics of different recommendation algorithms, we verify the advantages of the improved model in terms of recommendation accuracy and user satisfaction. We use precision, recall, and F1 score as evaluation metrics to compare the recommendation effects of the improved HMM, traditional HMM, and collaborative filtering algorithm. <Table 3> shows the specific experimental results:

Table 3: Experimental Results

Algorithm Type	Precision	Recall	F1 Score
Improved HMM	0.78	0.74	0.76
Traditional HMM	0.70	0.65	0.67
Collaborative Filtering	0.65	0.60	0.62

The comparison shows that the recommendation system based on the improved HMM outperforms the traditional HMM and collaborative filtering algorithm in all metrics. Detailed analysis is as follows: Precision: Precision refers to the proportion of recommended items that are actually accepted by the user. The experimental results show that the precision of the improved HMM is 0.78, significantly higher than the traditional HMM (0.70) and collaborative filtering algorithm (0.65). This indicates that the improved model can more accurately capture users' interests and preferences, enhancing recommendation accuracy. Recall: Recall refers to the proportion of actually accepted items successfully recommended by the system. [31] The recall of the improved HMM is 0.74, traditional HMM is 0.65, and collaborative filtering algorithm is 0.60. The improved model also performs well in recall, indicating it can more comprehensively cover users' interests, reducing missed recommendations. F1 Score: The F1 score is the harmonic mean of precision and recall, providing a comprehensive evaluation of recommendation system performance. The F1 score of the improved HMM is 0.76, traditional HMM is 0.67, and collaborative filtering algorithm is 0.62. The improved model significantly outperforms the other two algorithms in F1 score, indicating a better balance between precision and recall, resulting in the best overall recommendation effect. The excellent performance of the improved HMM in this experiment can be attributed to the following points: Dynamic State Transition Mechanism: The improved model introduces a dynamic state transition mechanism, allowing it to more flexibly adapt to dynamic changes in user behavior, enhancing the model's predictive capability. [32] Multimodal Data Integration: By integrating multimodal data (e.g., text, images, audio) and contextual information, the improved model can provide a more comprehensive description of user behavior, thereby improving recommendation accuracy. [33] Context Awareness: The improved model incorporates contextual information (e.g., time, location, device type), enabling it to better capture patterns of user behavior changes, improving recommendation relevance and user satisfaction.[34] In conclusion, the recommendation system based on the improved HMM outperforms traditional HMM and collaborative filtering algorithms in terms of recommendation accuracy and user satisfaction. The experimental results validate the effectiveness and advantages of the improved model, providing reliable data support and theoretical basis for practical applications. By further optimizing data preprocessing and feature extraction methods, recommendation system performance can be further enhanced.

5. Conclusion

This study proposes a recommendation system based on an improved Hidden Markov Model to address the limitations of traditional recommendation algorithms in terms of data sparsity and dynamic changes in user behavior. By introducing a dynamic state transition mechanism and multimodal data integration, the improved model significantly enhances recommendation accuracy and user satisfaction. Experimental results show that the improved HMM outperforms traditional HMM and collaborative filtering algorithms in precision, recall, and F1 score. Specifically, the dynamic state transition mechanism of the improved HMM allows it to flexibly adapt to changes in user behavior. Multimodal data integration and context awareness provide a more comprehensive description of user behavior. Through optimization of data preprocessing and feature extraction, the improved model demonstrates excellent recommendation performance. Future research can further explore the integration of more contextual information and improve model computational efficiency to enhance the performance and scalability of recommendation systems in practical applications. Overall, the recommendation system based on the improved HMM demonstrates its superiority in both theory and practice, offering a new solution for personalized recommendations.

References

[1] He, Shuyao, et al. "Lidar and Monocular Sensor Fusion Depth Estimation." Applied Science and Engineering Journal for Advanced Research 3.3 (2024): 20-26.

[2] Dai S, Dai J, Zhong Y, et al. The cloud-based design of unmanned constant temperature food delivery trolley in the context of artificial intelligence[J]. Journal of Computer Technology and Applied Mathematics, 2024, 1(1): 6-12.

[3] Liu T, Li S, Dong Y, et al. Spam detection and classification based on distilbert deep learning algorithm[J]. Applied Science and Engineering Journal for Advanced Research, 2024, 3(3): 6-10.

[4] Li S, Mo Y, Li Z. Automated pneumonia detection in chest x-ray images using deep learning model[J]. Innovations in Applied Engineering and Technology, 2022: 1-6.

[5] Lipeng L, Xu L, Liu J, et al. Prioritized experience replay-based DDQN for Unmanned Vehicle Path Planning[J]. arXiv preprint arXiv:2406.17286, 2024.

[6] Mo Y, Qin H, Dong Y, et al. Large language model (llm) ai text generation detection based on transformer deep learning algorithm[J]. arXiv preprint arXiv:2405.06652, 2024.

[7] Zhong Y, Liu Y, Gao E, et al. Deep Learning Solutions for Pneumonia Detection: Performance Comparison of Custom and Transfer Learning Models[J]. medRxiv, 2024: 2024.06. 20.24309243.

[8] Yan H, Wang Z, Xu Z, et al. Research on Image Super-Resolution Reconstruction Mechanism based on Convolutional Neural Network[J]. arXiv preprint arXiv:2407.13211, 2024.

[9] Yu Cheng, Qin Yang, Liyang Wang, Ao Xiang, Jingyu Zhang, Research on Credit Risk Early Warning Model of Commercial Banks Based on Neural Network Algorithm. Financial Engineering and Risk Management (2024) Vol. 7: 11-19. DOI: http://dx.doi.org/10.23977/ferm.2024.070402.

[10] Liyang Wang, Yu Cheng, Ao Xiang, Jingyu Zhang, Haowei Yang, Application of Natural Language Processing in Financial Risk Detection. Financial Engineering and Risk Management (2024) Vol. 7: 1-10. DOI: http://dx.doi.org/10.23977/ferm.2024.070401.

[11] Haowei Yang, Liyang Wang, Jingyu Zhang, Yu Cheng, Ao Xiang, Research on Edge Detection of LiDAR Images Based on Artificial Intelligence Technology. Journal of Image Processing Theory and Applications (2024) Vol. 7: 64-74. DOI: http://dx.doi.org/10.23977/jipta.2024.070108.

[12] Xiang, A., Huang, B., Guo, X., Yang, H., & Zheng, T. (2024). A neural matrix decomposition recommender system model based on the multimodal large language model. arXiv preprint arXiv:2407.08942.

[13] Zang H. Precision calibration of industrial 3D scanners: An ai-enhanced approach for improved measurement accuracy [J]. Global Academic Frontiers, 2024, 2(1): 27-37.
[14] Ni, F., Zang, H., & Qiao, Y. (2024, January). Smartfix: Leveraging machine learning for proactive equipment maintenance in industry 4.0. In The 2nd International scientific and practical conference "Innovations in education: prospects and challenges of today" (January 16-19, 2024), Sofia, Bulgaria, International Science Group (p. 313).
[15] Yan Z, Yajun W, Jiaqi Y. A Novel MPPT Algorithm for Photovoltaic Systems Based on Improved Sliding Mode Control [J]. Electronics, 2022, 11(15):2421-2425.
[16] Qi Z, Ma D, Xu J, et al. Improved YOLOv5 Based on Attention Mechanism and FasterNet for Foreign Object Detection on Railway and Airway tracks[J]. arXiv preprint arXiv:2403.08499, 2024.
[17] Xiang A, Qi Z, Wang H, et al. A Multimodal Fusion Network For Student Emotion Recognition Based on Transformer and Tensor Product[J]. arXiv preprint arXiv:2403.08511, 2024.
[18] Bo S, Zhang Y, Huang J, et al. Attention Mechanism and Context Modeling System for Text Mining Machine Translation[J]. arXiv preprint arXiv:2408.04216, 2024.
[19] Yan H, Wang Z, Bo S, et al. Research on Image Generation Optimization based Deep Learning[J]. 2024.
[20] Gao H, Wang H, Feng Z, et al. A novel texture extraction method for the sedimentary structures' classification of petroleum imaging logging[C]//Pattern Recognition: 7th Chinese Conference, CCPR 2016, Chengdu, China, November 5-7, 2016, Proceedings, Part II 7. Springer Singapore, 2016: 161-172.
[21] Wang Z, Yan H, Wei C, et al. Research on Autonomous Driving Decision-making Strategies based Deep Reinforcement Learning[J]. arXiv preprint arXiv:2408.03084, 2024.
[22] Tan C, Wang C, Lin Z, et al. Editable Neural Radiance Fields Convert 2D to 3D Furniture Texture[J]. International Journal of Engineering and Management Research, 2024, 14(3): 62-65.
[23] Wang X. Nonlinear Energy Harvesting with Tools from Machine Learning[D]. Duke University, 2020.
[24] Wang X S, Moore S A, Turner J D, et al. A model-free sampling method for basins of attraction using hybrid active learning (HAL)[J]. Communications in Nonlinear Science and Numerical Simulation, 2022, 112: 106551.
[25] Wang X S, Turner J D, Mann B P. Constrained attractor selection using deep reinforcement learning[J]. Journal of Vibration and Control, 2021, 27(5-6): 502-514.
[26] Wang X S, Mann B P. Attractor selection in nonlinear energy harvesting using deep reinforcement learning[J]. arXiv preprint arXiv:2010.01255, 2020.
[27] Wei Y, Gu X, Feng Z, et al. Feature Extraction and Model Optimization of Deep Learning in Stock Market Prediction[J]. Journal of Computer Technology and Software, 2024, 3(4).
[28] Sun M, Feng Z, Li Z, et al. Enhancing financial risk management through lstm and extreme value theory: A high-frequency trading volume approach[J]. Journal of Computer Technology and Software, 2024, 3(3).
[29] Tao Y. Meta Learning Enabled Adversarial Defense[C]//2023 IEEE International Conference on Sensors, Electronics and Computer Engineering (ICSECE). IEEE, 2023: 1326-1330.
[30] Tao Y. SQBA: sequential query-based blackbox attack[C]//Fifth International Conference on Artificial Intelligence and Computer Science (AICS 2023). SPIE, 2023, 12803: 721-729.
[31] Liu S, Yan K, Qin F, et al. Infrared image super-resolution via lightweight information split network[C]//International Conference on Intelligent Computing. Singapore: Springer Nature Singapore, 2024: 293-304.
[32] Jiang H, Qin F, Cao J, et al. Recurrent neural network from adder's perspective: Carry-lookahead RNN[J]. Neural Networks, 2021, 144: 297-306.
[33] Dang B, Ma D, Li S, et al. Deep Learning-Based Snore Sound Analysis for the Detection of Night-time Breathing Disorders[J].
[34] Dang B, Zhao W, Li Y, et al. Real-Time pill identification for the visually impaired using deep learning[J]. arXiv preprint arXiv:2405.05983, 2024.