doi: 10.20532/cit.2024.1005845

Leveraging Deep Learning for Personalized Book Recommendations: A Big Data Algorithm Combining Capsule Networks and Attention Mechanisms

Jiali Liao[1] and Tianxiang Li[2]

[1]School of Information Engineering, Sichuan Top IT Vocational Institute, Sichuan, China
[2]School of Information Engineering, Chengdu Industry and Trade College, Sichuan, China

In the era of big data, personalized book recommendations have become crucial for enhancing user satisfaction and improving information retrieval efficiency. This study addresses the limitations of existing book recommendation algorithms by proposing a novel Hybrid Book Recommendation Algorithm Considering Different Preferences (HBRACDP). Our approach integrates Capsule Networks and Self-Attention Mechanisms to model both short-term and long-term user borrowing preferences. We construct separate models for these preferences and combine them using a controllable multi-interest network with label attention. Experimental results on the Goodreads dataset demonstrate the superiority of HBRACDP, achieving an accuracy of 0.984, recall of 0.987, and F1 score of 0.988 in ablation tests. In practical scenarios with 1000 students, HBRACDP significantly outperformed traditional algorithms, with a recommendation accuracy of 97.89% and an error rate of only 0.08%. This study provides new insights for developing more accurate and efficient big data recommendation systems in library services and beyond.

ACM CCS (2012) Classification: Information systems → Information retrieval → Retrieval tasks and goals → Recommender systems

Computing methodologies → Machine learning → Machine learning approaches → Neural networks

Keywords: CNN, SAM, Book recommendations, Users, AM, CN

1. Introduction

In the era of the rapid development of information technology, people's material and spiritual lives have been greatly improved. However, as data volume increases, it becomes more difficult for people to process data. Therefore, big data technology is increasingly applied in people's daily lives. While the big data industry has gradually become a strategic industry, the personalized recommendation systems have also attracted significant attention.

Personalized recommendation systems have been gradually introduced in many fields, but in the library field, there are still many problems. Libraries store extensive book resources and as society advances, both the number of published books and the diversity of topics continue to grow. Therefore, libraries need to constantly update their collections, leading to an increase in the number of books each year.

With the increasing number of information sources, managing library functions becomes more complex. When faced with a rich collection of books and materials, readers are often unable to choose the right or favorite books without specific needs. In this way, readers are often unable to get targeted personalized help, thus weakening the readers' ability to use the books effectively. Therefore, it is essential to improve the availability and utility value of library resources to prevent more library resources from being unused and wasted.

Recommendation systems can be used to analyze the book borrowing records and user browsing records, mine the readers' interest in borrowing books, extract a large amount of original data and convert them into valuable information, and recommend the books in line with user preferences. However, at present, the library book recommendation systems are still in early development stages, relying on outdated algorithms, mainly based on reader evaluation and scoring records, and cannot accurately analyze readers' interests and hobbies, leading to less precise recommendations. Manual screening methods further contribute to an overwhelming volume of irrelevant data, requiring readers to spend more time finding suitable books.

At the same time, current library book recommendation systems also have problems such as limited functionality and user inconvenience. Therefore, delivering personalized and accurate recommendation services for users from massive book databases has become an important topic in the research field of information retrieval and recommendation systems [1-2].

The traditional Book Recommendation Algorithm (BRA) mainly includes methods such as Collaborative Filtering (CF) and content-based recommendation. This type of method performs poorly in the face of cold start and data sparsity problems, and relies heavily on book attribute information, limiting flexibility in capturing changes in user interests [3-5].

Deep learning technology has shown strong capabilities in the field of recommendation systems, especially in handling complex high-dimensional data and capturing nonlinear relationships. Capsule Network (CN), as an emerging neural network model, is seen as an improvement on traditional Convolutional Neural Network (CNN). At present, CN has achieved significant results in fields such as image recognition with its unique dynamic routing mechanism and powerful feature representation ability. Self-Attention Mechanism (SAM) has also made breakthrough progress in natural language processing in recent years.

To avoid factors such as data sparsity affecting the recommendation accuracy, this study designed a novel Book Big Data Recommendation Algorithm (BBDRA) that combines CN, SAM, and user borrowing preferences at different times. The innovation of the research lies in the ability of CN to capture high-level features of books, while SAM can enhance the model's understanding of user behavior. By combining the two, more accurate recommendations can be achieved in complex book big data environments.

2. Literature Review

With the progress of the information industry and the Internet, various big data issues have gradually attracted wide attention, and recommendation systems have also emerged. In terms of deep learning, regarding the issue of e-commerce product recommendation, Latha *et al.* [6] proposed a deep learning-based recommendation framework to enhance the sales of e-commerce websites. The proposed model achieved an average recall rate of 94.80%, a precision rate of 93.64%, and an accuracy rate of 96.92% on the Amazon product review database, which is superior to traditional CNN.

Lin *et al.* [7] developed a service recommendation method based on deep neural CF to address the computational challenges of service recommendation systems in handling distributed and multi-source big data resources and constructed a recommendation model using the cloud edge collaborative computing paradigm. This method had high recommendation accuracy.

Chiranjeevi and Rajaram [8] propose a lightweight deep learning-based recommendation system that generates recommendations through sentiment analysis of Twitter comments. First, the data collected from Twitter is cleaned, and then the pre-processed data is fed into a lightweight deep-learning recommendation model to learn four categories of features. Finally, the model classifies the data into positive, negative, and neutral emotions based on the learned features, and generates recommendations based on the classification results. Experimental results show that the model has excellent performance in terms of accuracy, accuracy, recall, F-value, and error rate, especially in the Twitter dataset, achieving 95% accuracy.

In the traditional recommendation system, Bhaskaran *et al.* [9] proposed an enhanced vector space recommender aimed at utilizing it to automatically track the interests, needs, and knowledge levels of learners. This study aimed to generate better recommendation lists by improving content filtering and adjusting cosine similarity. The average absolute error value of the model ranged from 5.08% to 25.26%, and the accuracy was between 80% and 93%.

Li *et al.* [10] proposed a sequence recommendation framework based on the diffusion model DiffuRec. DiffuRec can reflect multiple user interests and multiple characteristics of a product by representing the product as a distribution rather than a fixed vector. In the diffusion phase, DiffuRec represents the target commodity embedding as a Gaussian distribution, which is used to generate sequential commodity representations and inject uncertainty. The commodity representation is then reconstructed to make predictions. Experimental results show that DiffuRec performs significantly better than traditional methods on the four data sets.

In terms of mixed recommendation methods, in response to the cold start problem of traditional recommendation techniques, Wei *et al.* [11] designed a mixed probability multi-objective evolutionary algorithm in a multi-objective recommendation system, achieving effective optimization of conflict indicators. Wen [12] proposed a music feature extraction algorithm built on a mid-level feature structure to extract the underlying features of different music scene images. This algorithm had the highest recognition rate in indoor music scenes, about 87.6%. Ma *et al.* [13] proposed a user CF algorithm built on kernel methods and multi-objective optimization. This algorithm aimed to further improve the diversity and accuracy of systems by introducing kernel density estimation. The results of the Netflix dataset showed that the proposed algorithm has improved accuracy by 5.6% and increased diversity.

Yannam *et al.* [14] proposed a group recommendation method based on deep collaborative filtering to improve the recommendation effect. This method makes use of metadata for prediction and alleviates the problem of sparse data. Through the combination of multi-layer perceptron and generalized matrix decomposition, the model first learns from group-item interaction and then combines metadata and item metadata for prediction. The experimental results show that the proposed method effectively alleviates the cold start problem in the group recommendation.

Bakariya *et al.* [15] proposed a facial expression recognition and music recommendation system based on CNN. The system analyzes the user's facial expressions to infer the mood and recommends music based on the mood. Experimental results show that the accuracy rate of this model is 73.02%, which effectively improves the application effect of real-time facial recognition and sentiment analysis in music recommendation.

To summarize, many scholars have carried out research on various recommendation tasks, including e-commerce recommendation, project recommendation, music recommendation, *etc.*, with the primary goal of enhancing user satisfaction and loyalty. However, the accuracy and recommendation results depend on the huge amount of user rating data. If the data is missing, data sparsity, cold start, and other problems will occur, which are difficult to overcome relying solely on traditional recommendation algorithms.

Therefore, this research focuses on book recommendations by leveraging insights from readers' borrowing behaviors captured through mobile devices. User's borrowing behavior is categorized into short-term borrowing and long-term borrowing behavior. Short-term behavior includes users' recent reading records, borrowing records, and emotional records. Long-term borrowing behavior encompasses the user's long-term borrowing record. Using short-term borrowing behavior data to make recall recommendations improves the diversity of recall recommendation algorithm on the premise of ensuring the accuracy of recommendation. Combining both short- and long-term data in ranking recommendations, aims to address the problem of user interest drift.

To tackle the traditional book recommendation algorithms' limitation in recognizing user interest diversity, this study proposes a controllable fusion model that serializes diverse users interests for book recommendations. This model adopts a dynamic routing algorithm based on capsule network to identify multiple user's interests by aggregating different historical behavior sequences into distinct sets. Each set is analyzed to learn specific interests, generating user vectors. A merging strategy based on user-item similarity then aggregates these varied interests to rank the final recommended items. The research aims to further improve the accuracy rate of university book recommendation within the context of big data.

3. Research Methodology

To improve the book recommendation performance of university libraries for different students, this study not only deeply analyzed the book borrowing habits of university students at different stages, but also designed recommendation algorithms to recommend suitable books for students.

3.1. Construction of a Long- and Short-Term Book Borrowing Preference Model

In short-term borrowing, assuming user u and each user's short-term borrowing book information is L, L is used as the input sequence for the user's short-term preference model. Let the vector embedding dictionary matrix of the book be $M \in R^{L \times d}$, where d denotes the dimension of vector embedding and R is the range of the matrix. According to the above definition, the Vector Embedding Matrix (VEM) expression for User Short-Term Borrowing Behavior (USTBB) is obtained as shown in equation (1) [16].

$$E = [m_{s1}, m_{s2}, ..., m_{sL}] \in R^{L \times d} \tag{1}$$

In equation (1), E is the VEM of USTBB. $[m_{s1}, m_{s2}, ..., m_{sL}]$ is each vector in the matrix. To accurately predict the next real borrowing situation based on USTBB, this study introduces SAM and adds a positional VEM in E that can actively complete behavioral learning. This matrix is denoted as $P = [p_1, p_2, ..., p_L] \in R^{L \times d}$. At this point, the input matrix of SAM is equation (2).

$$X^{(0)} = [x_1, x_2, ..., x_L] \in R^{L \times d} \tag{2}$$

In equation (2), $X^{(0)}$ is the input matrix of SAM, and $[x_1, x_2, ..., x_L]$ are the vectors in $X^{(0)}$. In $X^{(0)}$, the initialized User's Short-Term Borrowing Preferences (USTBP) are shown in equation (3).

$$x_l^{(0)} = m_{sl} + p_l,\ l \in [1, 2, ..., L] \tag{3}$$

In equation (3), $x_l^{(0)}$ is the initialized USTBP. m_{sl} represents the USTBB vector in the initial state. p_l is the initialization weight in the adaptive mechanism. The output of the b-th block obtained by inputting $X^{(0)}$ into multiple overlapping self-attention blocks is defined in equation (4).

$$X^{(b)} = SAB^{(b)}(X^{(b-1)}),\ b \in [1, 2, ..., B] \tag{4}$$

In equation (4), $X^{(b)}$ and $SAB^{(b)}$ are the outputs and SAM of the b-th block. $X^{(b-1)}$ represents the output of the $b-1$-th block. Every time self-attention blocks are applied, the data in the sequence is further abstracted and optimized. By recursively applying self-attention blocks to process and extract data, each layer of dynamic preferences of users can be better recognized.

In Long-Term Borrowing (LTB) behavior, due to different historical borrowing information having different weights, this study introduces the Attention Mechanism (AM) to complete weight design [17]. In AM, "query" usually refers to the current focus or target, used to retrieve and determine which parts from larger datasets are most relevant. The weight optimization is completed using "query", and the reassignment calculation process is equation (5).

$$y = LBA(E') = \text{softmax}(q^s(E'W'_K)^T)E'W'_V \tag{5}$$

In equation (5), the second E' represents the user's long-term preference vector. The third E' is the long-term behavioral preference matrix of the input. W'_K and W'_V are two different weighted key and value matrices. $LBA(E')$ is a linear weighted combination based on E', which determines the importance of each vector embedding through AM. q^s is a learnable parameter of AM, used to control the distribution of attention weights. Adding a dropout network to the LTB preference model yields an input of $y_l = Dropout(y)$. By combining equation (5) and $y_l = Dropout(y)$, the LTB preference matrix can be obtained as $Y \in R^{L \times d}$.

3.2. Design of a Book Recall Recommendation Model Based on USTBB

Due to the fact that the embedding vector of unified user borrowing preferences cannot reflect multiple interests of users during a certain period of time, this study proposes a Controllable Multi-Interest Network with Label-Attention for Book Recommendation Model (CMINLA-BRM) that integrates diverse borrowing behaviors of users based on USTBB. CMINLA-BRM can extract borrowing behavior preferences from user short-term behavior sequences, retrieve candidate items, and then input the retrieved items into the aggregation module to calculate user preference similarity. The overall structure of CMINLA-BRM is illustrated in Figure 1.

In Figure 1, the input of each sample can be defined as a triplet, *i.e.*, (I_u, O_u, F_u). I_u represents the historical behavior of user u. O_u represents other attribute features such as user code and user type. F_u represents other characteristics of the book, such as book code, book type, and collection location. The main task of the multivariate interest extraction layer is to learn the function that maps the original features to the user representation vector, as shown in equation (6) [18].

$$V_u = f_{user}(I_u, O_u) \tag{6}$$

In equation (6), $V_u = \left(\vec{v}_u^{\,1}, ..., \vec{v}_u^{\,K}\right) \in R^{d\times K}$ is the representation vector of user u. d represents the embedding dimension. K is the number of interests of the user. f_{user} represents the mapping function. The embedding vector function $\vec{e}_{i'}$ of candidate book i' is equation (7).

$$\vec{e}_{i'} = f_{item}(F_{i'}) \tag{7}$$

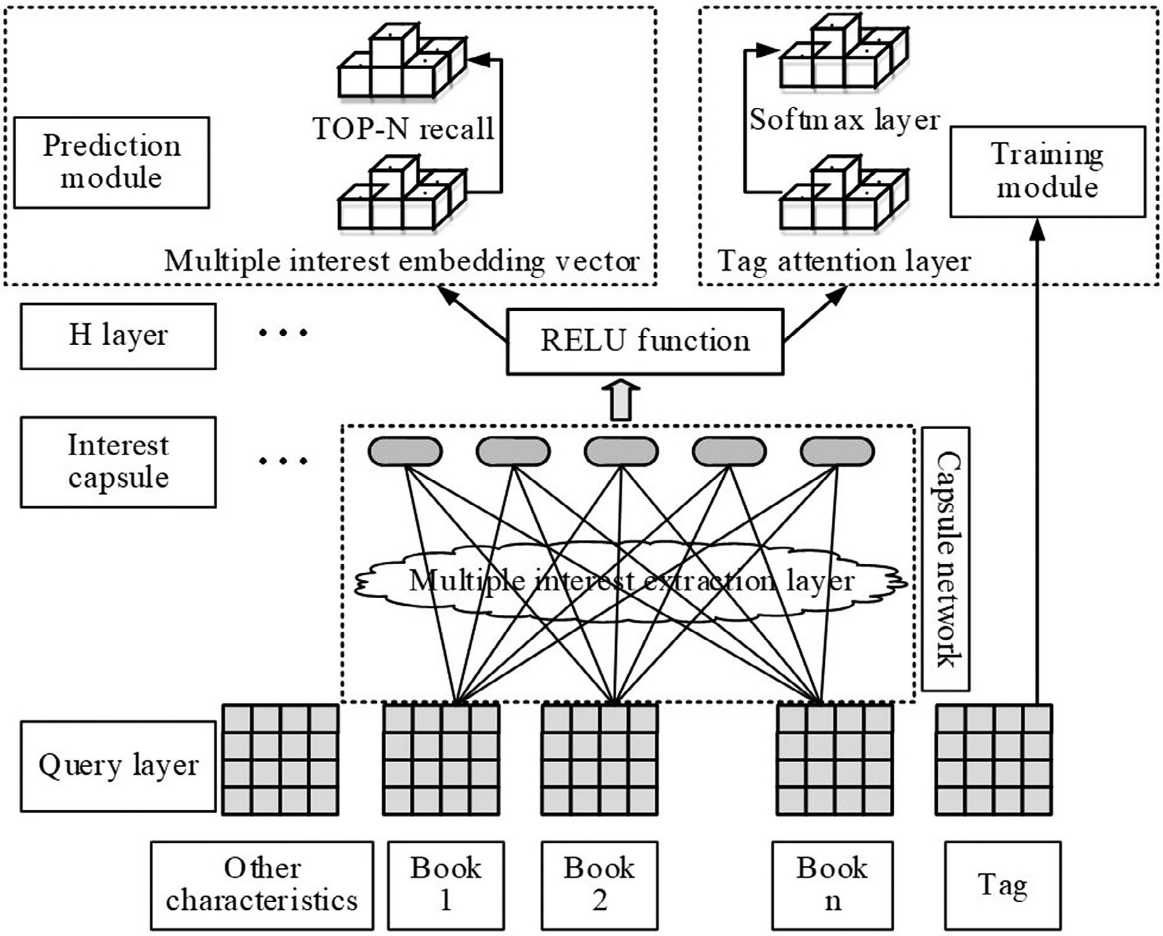


Figure 1. Structure of CMINLA-BRM model.

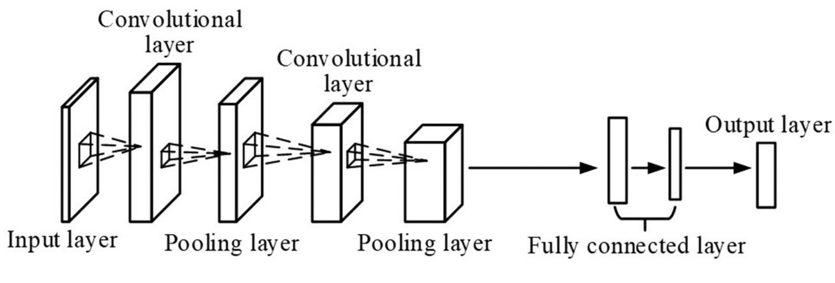
In equation (7), $\vec{e}_i \in R^{d\times K}$, f_{item} is the pooling operation of the embedding vector, and the recommendation is based on the maximum inner product $F_{i'}$ of the candidate book and the user representation vector. In the book recall stage, this study uses multiple representation vectors to retrieve user interests and utilizes dynamic routing methods in CN to aggregate and classify user historical behavior. Books with strong relevance will be grouped together to represent a certain user interest. As a special form of CNN, CN has better feature recognition ability compared to traditional CNN. It can model the hierarchical relationships of data in a more structured way and deal with the matter of information loss that CNN may cause during pooling operations. The structures of CN and CNN are shown in Figure 2.

In Figure 2, in CN, a process called "dynamic routing" determines how information should be transmitted from low-level to high-level capsules in the network. This mechanism replaces the pooling layer in traditional CNN, enabling the network to better maintain spatial hierarchical relationships and effectively identify different perspectives, sizes, and deformations of visual objects [19]. The dynamic routing of CN iteratively calculates the value of the interest capsule through the initial capsule, treating the embedding vector of the user behavior sequence as the initial capsule and the multivariate user interests as the interest capsule . The calculation process is defined in equation (8).

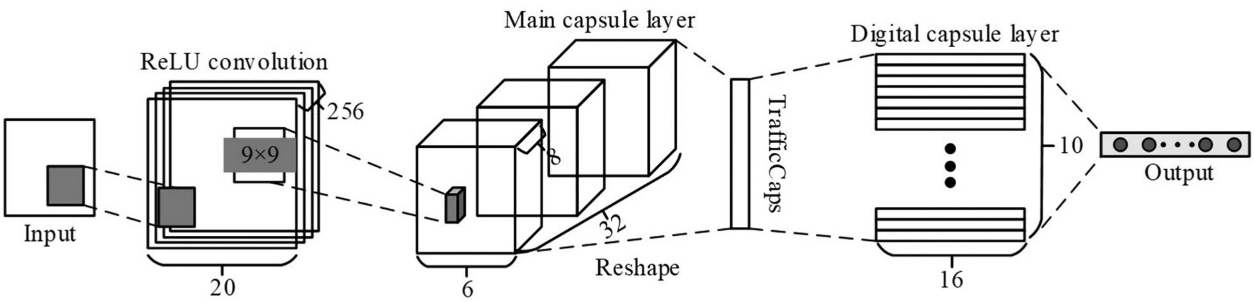
$$\begin{cases} \hat{e}_{j|i} = W_{ij} e_i \\ s_j = \sum_i c_{ij} \hat{e}_{j|i} \\ c_{ij} = \exp(b_{ij}) \Big/ \sum_k \exp(b_{ik}) \end{cases} \tag{8}$$

In equation (8), e_i is the initial capsule. W_{ij} is the transformation matrix. $\hat{e}_{j|i}$ represents the prediction vector. s_j is the weighted sum of all prediction vectors. c_{ij} is the coupling coefficient. b_{ij} is the logarithmic prior probability of i and j coupling. By using nonlinear functions to compress short vectors towards 0 and long vectors slightly below 1, the output vector of the interest capsule can be obtained as shown in equation (9).

$$v_j = squash(s_{ij}) = \frac{\|s_j\|^2}{1+\|s_j\|^2} \frac{s_j}{\|s_j\|} \tag{9}$$



(a) Traditional CNN structure diagram.



(b) CN structure diagram.

Figure 2. Structure diagram of CNN and CN.

In equation (9), v_j represents the output capsule. The output interest capsule matrix of user u is $V'_u = [v_1, ..., v_K] \in R^{d \times K}$. After extracting the diverse interests of users, the similarity between interest capsules and candidate books can be calculated to evaluate the user's interest in specific books. The output vector $\vec{v}_u$ of user candidate books is defined in equation (10).

$$\begin{aligned} \vec{v}_u &= Attention(\vec{e}_{i'}, V_u, V'_u) \\ &= V_u\, soft\max(pow(V_u^T \vec{e}_{i'}, p')) \end{aligned} \tag{10}$$

In equation (10), pow represents the exponential operation of each element. p' represents an adjustable parameter used to adjust attention distribution. When p' approaches 0, each interest capsule gains the same weight. When $p' > 1$ is reached, the weight obtained by points with larger values is proportional to the value of p'. After obtaining $\vec{v}_u$, algorithm training can be carried out based on the provided training samples. After completion, multiple interest vectors of the user can be obtained, and each interest vector can retrieve multiple most relevant candidate books.

3.3. Design of BBDRA that Integrates SAM with Different Borrowing Preferences

In the current BRA, due to the explosive growth in the number of books and user data, many traditional recommendation algorithms have drawbacks such as low computational efficiency, high computational costs, slow algorithm operation, and poor user recommendation performance. Considering the diversity and dynamic characteristics of user interests, this study first starts with USTBB and LTB behavior and constructs two different book borrowing preference models. Secondly, a novel BRA based on CMINLA-BRM is proposed, referred to as Hybrid Book Recommendation Algorithm Considering Different Preferences (HBRACDP). The overall structural framework of HBRACDP is depicted in Figure 3.

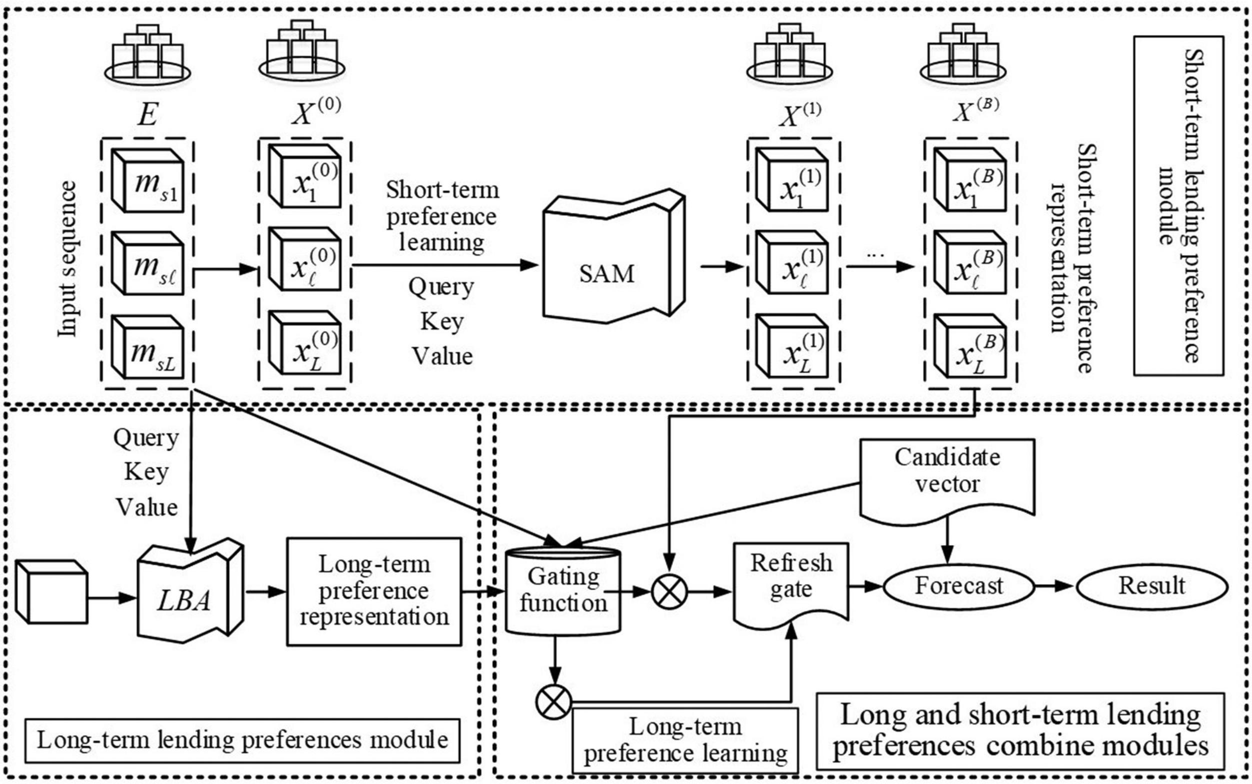


Figure 3. Structural framework of HBRACDP.

The HBRACDP framework in Figure 3 can be divided into three parts, namely the USTBP module, LTB preference module, and Long Short-Term Borrowing (LSTB) preference joint module. In the USTBP module of HBRACDP, the retrieval and recall of USTBP are mainly completed by CN and SAM. In the LTB module of HBRACDP, the long-term interest preferences of users are modeled by combining CN and AM. Finally, the user's LSTB preferences are combined to obtain the output vector through a gating function.

When constructing a recommendation model, the selection of hyperparameters will affect the performance of the model. The main hyperparameters considered in this study include the dimension of the embedding vector, the number of attention heads, and the number of iterations of the dynamic route. First, the dimension of the embedding vector is selected as 128. This choice is based on common embedding dimension settings in the literature and combined with the size and number of features of this dataset. Using 128 dimensions enables striking a balance between model performance and computational overhead. Secondly, the number of attention heads selected is 4. The multi-head attention mechanism allows the model to focus on different information in different subspaces, and four attention heads can improve the representation of the model while avoiding excessive computational complexity. Finally, the number of iterations of the dynamic route is set to 300. The dynamic routing mechanism is used for information transfer in the capsule network, and 300 iterations can achieve a good balance between performance and computational efficiency.

In the HBRACDP algorithm, SAM plays a crucial role in creating sequence models for different user behaviors, thereby improving recommendation accuracy. The structure of SAM is illustrated in Figure 4.

In Figure 4, the input sequence of SAM is first processed to generate three different sets of vectors, namely Query, Key, and Value. Firstly, each input element will be converted into these three types of vectors. Secondly, an attention score is generated by calculating the similarity between the Query and all Key vectors. These scores are then fed into a softmax layer, which will normalize the scores to make their sum 1. The score normalized through the softmax layer is called an attention map, which is used to weight the value vector in order to focus more attention on the more relevant input parts. Finally, these weighted value vectors are added up to form the final output, which captures the information most relevant to the current query in the input sequence. Through this approach, SAM enables the model to focus on the most relevant part of the input sequence to the current task, thus enabling the final recommendation model to make appropriate book recommendations based on the user's reading preferences. The flow framework diagram of the whole big data

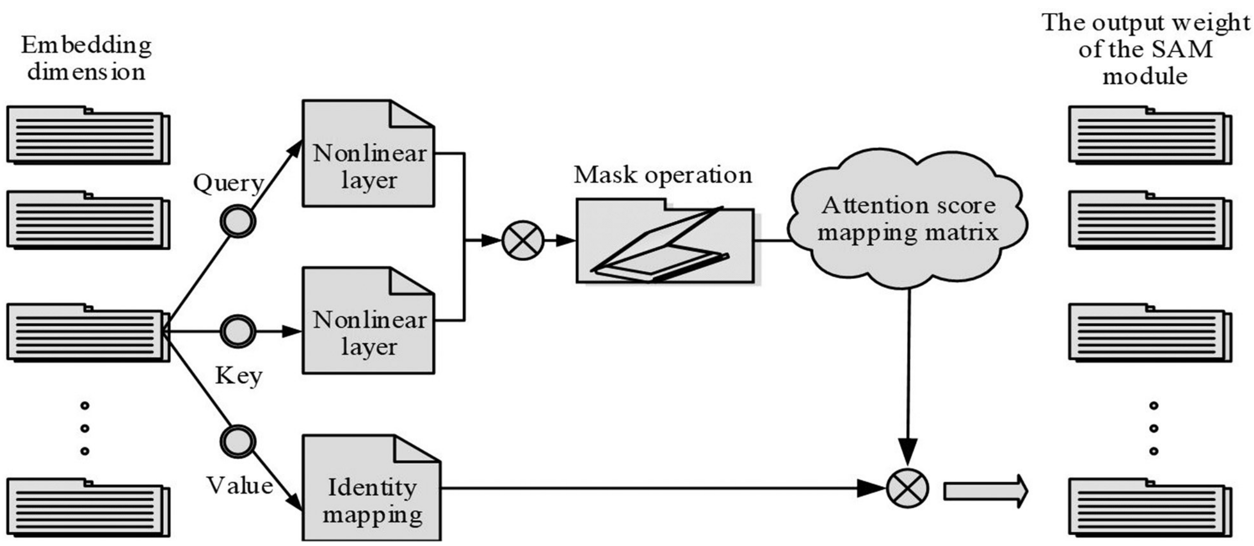


Figure 4. SAM structure.

recommendation algorithm with fusion CNN and attention mechanism is given in Figure 5.

The proposed algorithm in Figure 5 presents the book recommendation algorithm from the two stages of recall and sorting, respectively. A series of problems such as how to establish profiles for readers, how to provide readers with diversified and accurate recommendations, and how to reduce the sparse data and the interest drift of readers are analyzed and studied.

4. Results and Discussion

To demonstrate the effectiveness of the proposed HBRACDP algorithm based on BBDRA, this study evaluated its benchmark performance and actual recommendation performance. In addition, the latest references in relevant fields are also introduced for discussion and analysis.

4.1. Data Preprocessing and Feature Selection

The algorithm was implemented using the deep learning framework PyTorch, with GPU acceleration to optimize network parameter updates. To maintain experimental consistency, all algorithms in the study were executed within this framework. The experimental environment details are provided in Table 1.

The experimental dataset consists of borrowing records from a university library, covering reader activity during the second half of 2019. The dataset includes 20148 students in half a year and 672032 borrowing behavior records, with each student having at least 10 borrowing operation records.

The dataset is divided into three parts: user data, library data, and borrowing behavior data. User data includes user ID, reader type, and reader grade. Book data includes book ID, book type, and library collection. Borrowing behavior data includes user history borrowing record, user recent book record, user recent stay location, and user borrowing emotion record. User emotional records are obtained based on the average monthly retrieval frequency of books. The higher the borrowing frequency of the user, the higher the borrowing intention. The behavioral sequence cutoff length was 30 for each training sample.

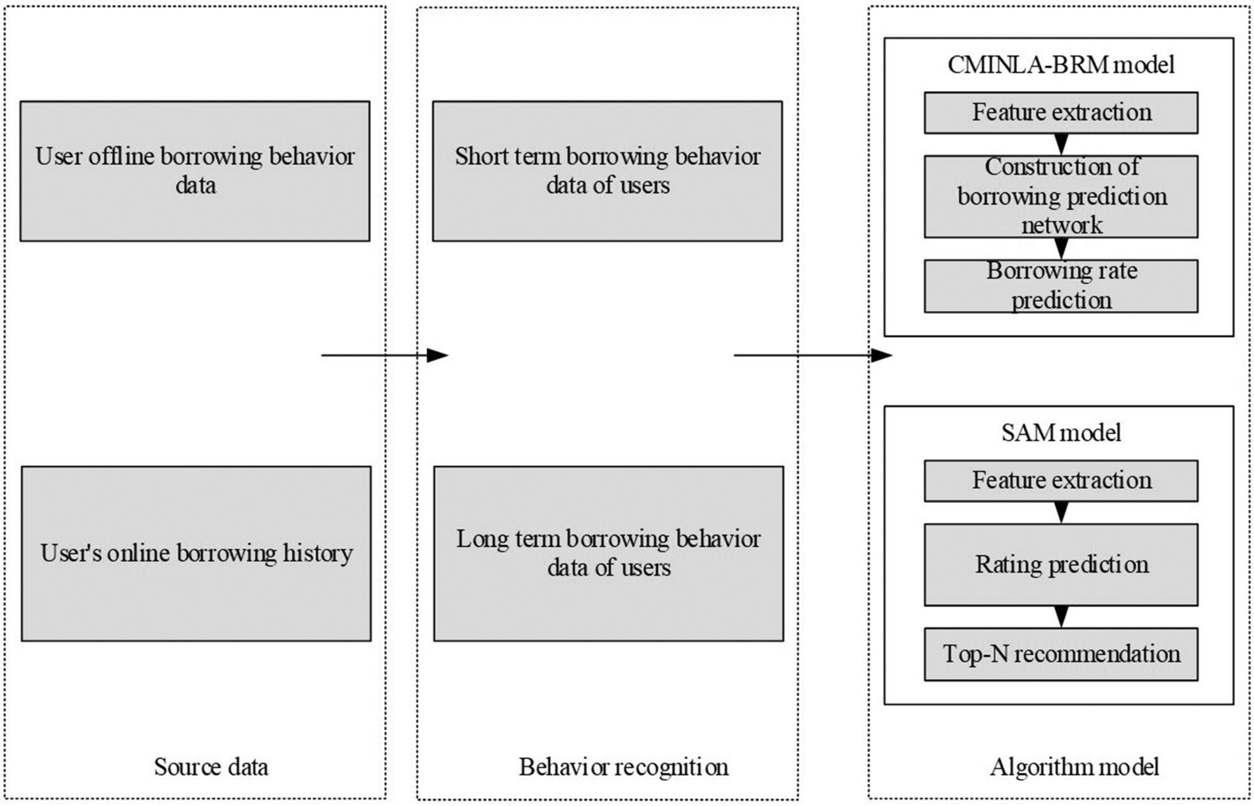


Figure 5. Framework diagram of the proposed algorithm.

Table 1. Experimental environment.

Configure	**Parameter**
Operating system	Windows10
Processor	Intel(R)Core(TM)i5-10210U CPU @1.60GHz 2.11 GHz
Internal memory	16GB
Integrated development tools	Pycharm
Programming language	Python 3.7
Skeleton frame	Pytorch
Storeroom	Numpy and Pandas

4.2. Results

4.2.1. Model Benchmark Performance Test

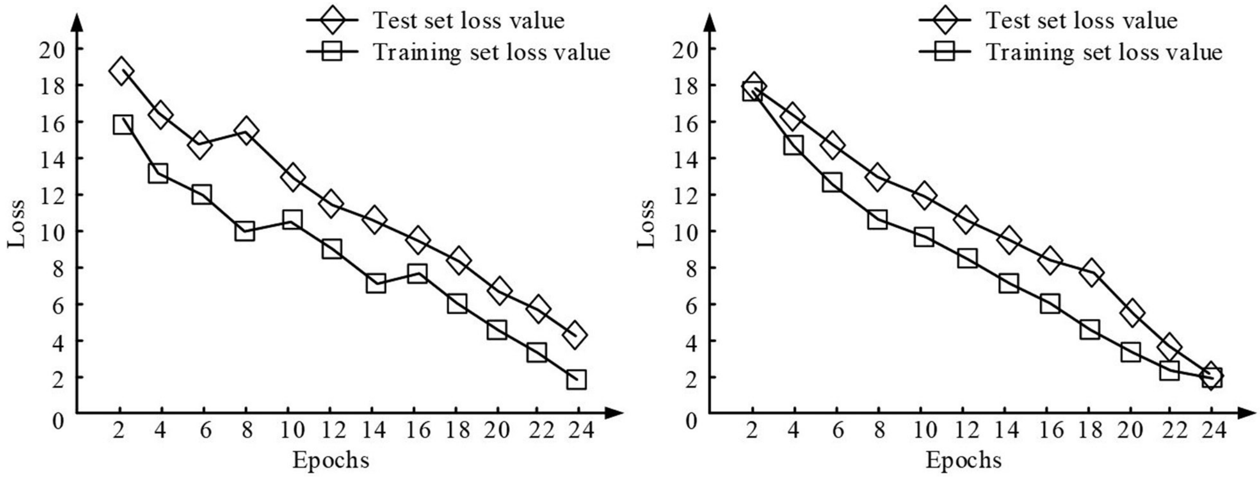
To verify the benchmark performance of BB-DRA-based HBRACDP, this study chooses the publicly available dataset Goodreads as the experimental dataset. It includes user ratings for different books and various metadata related to books, which can evaluate the recommendation performance of algorithms. The Goodreads dataset contains more than 2 million user ratings and about 500,000 books. During data preprocessing, users and books with less than 5 rating records were first removed, and the data was then divided into a training set and a test set in an 8:2 ratio. In addition, the user rating is standardized to improve the stability and effectiveness of model training. Due to the fact that the final designed HBRACDP is composed of multiple parts, this study first conducts ablation testing on HBRACDP. Table 2 shows the detailed results.

Table 2. Ablation test results of HBRACDP.

Model	**Precision**	**Recall**	**F1 value**
CNN	0.856	0.867	0.862
CN	0.871	0.885	0.878
CNN+SAM	0.881	0.886	0.885
CNN+AM	0.891	0.883	0.887
CN+AM	0.912	0.925	0.923
CN+SAM (CMINLA-BRM)	0.945	0.938	0.941
CN+SAM+AM (HBRACDP)	0.984	0.987	0.988

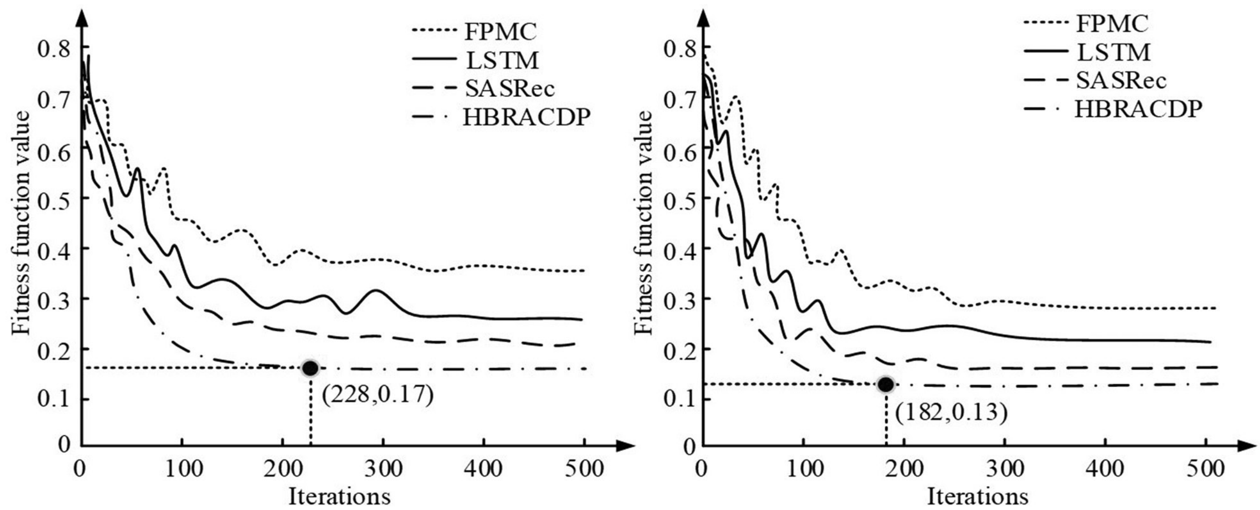
In Table 2, the HBRACDP algorithm combining CN, SAM, and AM structures has the best ablation test results, with the highest accuracy, recall, and F1 values of 0.984, 0.987, and 0.988. The performance of algorithms using separate CNN and CN structures is poor, with the lowest accuracy, recall, and F1 values of 0.856, 0.867, and 0.862. To avoid overfitting in the algorithm, regularization operations are performed on the HBRACDP algorithm before testing, and the results are shown in Figure 6.

In Figure 6(a), when dropout=0, the curve changes of HBRACDP in both sets are more oscillatory, and the loss function on the training set is lower, indicating overfitting of the model. In Figure 6(b), when dropout=0.4 is set, the changes in both curves are relatively stable, and the loss function is ultimately unified. The experiment selects Factorizing Personalized Markov Chains (FPMC), Long Short-Term Memory Network (LSTM), and Self-Attention Sequential Recommendation (SASRec) as comparative algorithms to test the iterative performance of the four algorithms in the dataset, as shown in Figure 7.



(a) Loss curves without dropout regularization. *(b)* Loss curves using dropout regularization.

Figure 6. Comparison chart of loss training curves with and without dropout regularization.



(a) Training set. *(b)* Test set.

Figure 7. Iterative performance of different algorithms in various datasets.

Figures 7(a) and (b) show the iterative curves of fitness values for the four algorithms in the datasets. HBRACDP has the best iterative performance in both sets. In the training set, HBRACDP only needs 228 iterations to reach a stable state, with a fitness of 0.17. In the test set, the algorithm can reach a stable state after 182 iterations, and the fitness value in the stable state is 0.13. Figure 8 shows the PR curves of each algorithm for further testing on two datasets.

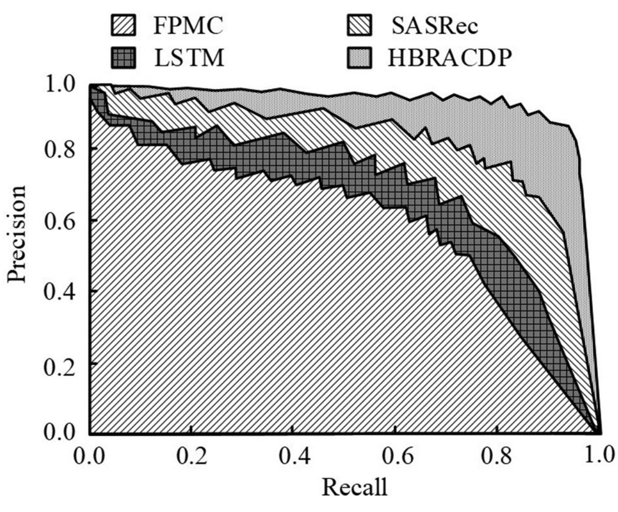
The PR curves in Figures 8 (a) and 8 (b) indicate that the maximum AUC areas of FPMC, LSTM, SASRec, and HBRACDP in the training and testing sets are 0.73, 0.86, 0.91, and 0.96, respectively. This indicates that HBRACDP has the best recommendation performance in benchmark testing.

4.2.2. Model Application Effect Test

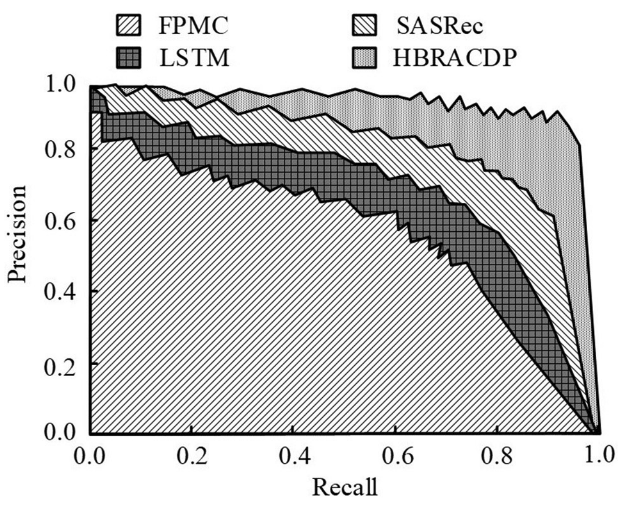
In addition to verifying the benchmark performance of HBRACDP, this study also tested its practical application effectiveness. To ensure the existence of LSTB behavior, a total of 100 students from a certain university with borrowing records greater than 50 times in the past year were selected as the research subjects. The book recommendation performance of four algorithms on 100 students was tested, as shown in Figure 9.

In Figure 9, N means the number of books recommended by the recommendation algorithm on the basis of the user's borrowing preferences for candidate books, and a larger N value indicates a higher recommendation value. In Figures 9 (a) and 9 (b), when the number of N is 70, the recommendation errors of FPMC, LSTM, SASRec, and HBRACDP are 0.34%, 0.29%, 0.23%, and 0.06%, and the recommendation accuracy is 78.5%, 83.8%, 86.9%, and 97.2%. This is because the use of self-attention book sequence can fully tap the readers' short-term interest preference information. It can effectively simulate the phenomenon of reader personalized aggregation in the book sequence recommendation, so as to improve the performance of the recommendation. At the same time, using the gating function to calculate the similarity between candidate items and users' long-term and short-term interests to assign different weights on long-term and short-term interests, which can effectively weaken the impact of interest drift in sequence recommendations, and thus improve the performance of recommendations.100 students were segmented into LTB group and Short-Term Borrowing (STB) group, and the average recommendation time of four recommendation algorithms for the two groups of students is Figure 10.

In Figure 10, FPMC has the highest average recommendation time for LTB and STB students, with values of 0.35 and 0.44. HBRACDP

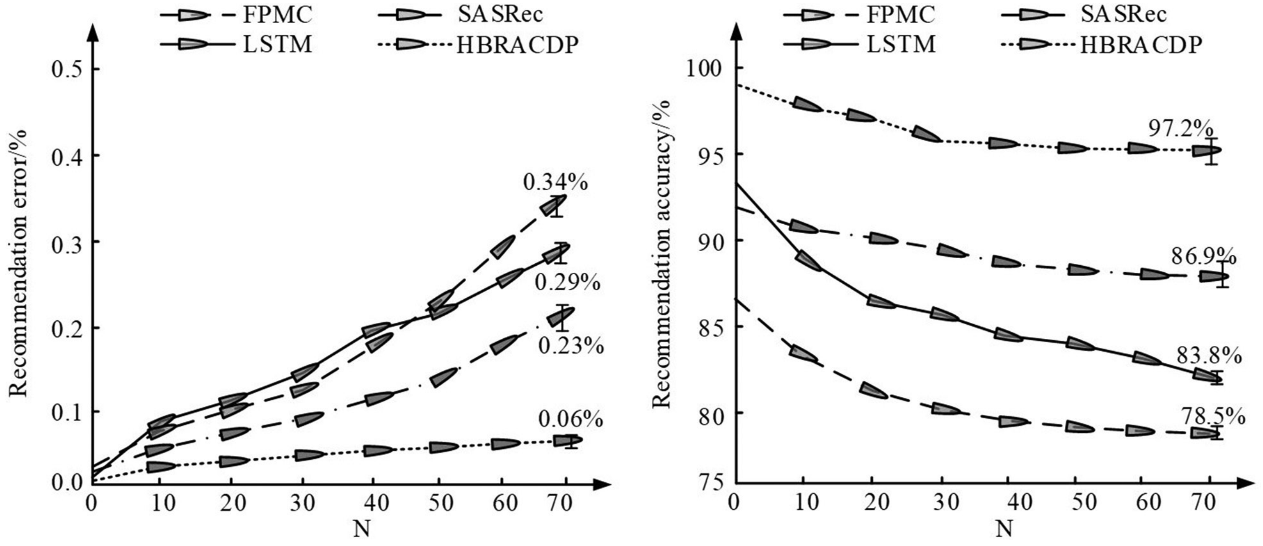


(a) PR curves of the four detection algorithms in the training set.



(b) PR curves of the four detection algorithms in the test set.

Figure 8. Comparison of PR curves of various algorithms.



(a) Recommendation error of four algorithms.

(b) Recommendation accuracy of four algorithms.

Figure 9. Error values and recommendation accuracy of four algorithms under different N values.

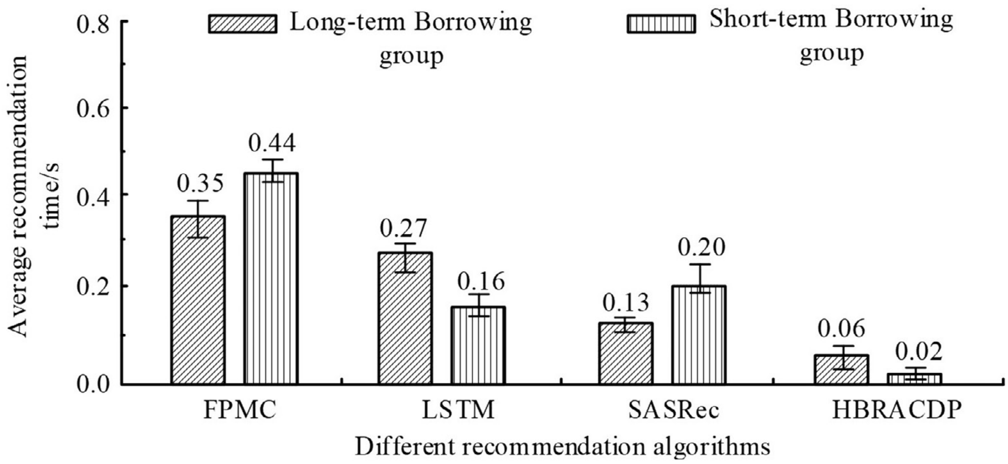


Figure 10. Average recommendation time of four algorithms for two groups of students.

has the shortest average recommendation time for both groups of students, with values of 0.06 and 0.02, respectively, while LSTM and SASRec fall between these two algorithms. This shows that the proposed sequence recommendation model integrating long-term borrowing behavior is effective and feasible and can effectively solve the problem of user interest drift. Combined with the user's offline borrowing behavior sequence, the proposed model uses the self-attention mechanism to effectively extract the user's short-term interest preference, fully excavates and extracts the user's short-term interest preference, and makes up for the problem of sparse user borrowing records. Innovative use of a gating function to model users' long and short-term borrowing preferences, integrates users' long and short-term preferences, and effectively solves the problem of users' interest drift. In addition, the proposed model showed excellent recommendation performance in the comparison experiments with other benchmark models.

To better demonstrate the performance of the recommendation model in big data applications, the study expanded the sample size of the real-world application test to 1,000 college students with more than 100 borrowing records in the past year. The number of recommended books was set to 100, and the comparison of the performance of four recommendation algorithms is shown in Table 3.

In Table 3, through the test results of 1000 college students, it can be seen that when the number of recommended books is 100, HBRACDP algorithm achieves the shortest recommendation time, the highest recommendation accuracy, and the lowest recommendation error, which is significantly better than the other three algorithms. The recommendation time of HBRACDP is as low as 0.09s, the recommendation accuracy is as high as 97.89%, and the recommendation error is as low as 0.08%. In order to see the superiority of the research model more intuitively, the research visually compares the precision rate, recall rate, mAP and other data indicators of the traditional model and the research model in the process of training the data set. The results are shown in Figure 11.

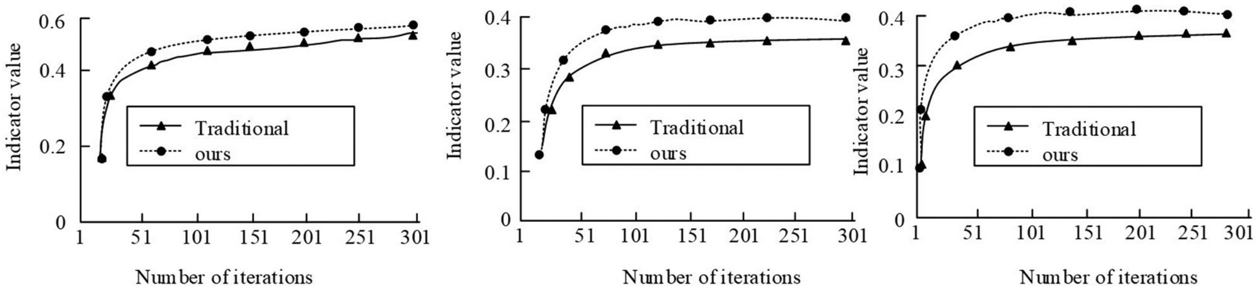
From Figure 11, it can be seen that both the precision and recall of the study model stabilized after 51 iterations, while the mAP stabilized after 25 iterations. Both the precision and recall of the traditional algorithm stabilized after 101 iterations, and the mAP stabilized after 51 iterations. Compared with traditional algorithms, the proposed model is more efficient and accurate.

4.3. Discussion

To improve the performance of traditional BRA, this study designed a BBDRA-based HBRACDP that combines CN and SAM and tested its performance. Compared with the multi-modal deep learning framework proposed by Li Y *et al.* [20], which combined image processing and text analysis, although this framework performed well on multi-modal data, it performed poorly when facing single text data. However, HBRACDP not only improved the accuracy of feature extraction but also enhanced the model's understanding of user behavior.

Table 3. Recommendation effects of different models.

Recommendation model	Recommended time/s	Recommended accuracy rate/%	Recommended error/%
FPMC	0.38	82.36	0.62
LSTM	0.26	87.43	0.48
SASRec	0.22	91.25	0.29
HBRACDP	0.09	97.89	0.08



(a) Precision. *(b)* Recall. *(c)* mAP.

Figure 11. Comparison results of different parameters.

In the experiment, the HBRACDP algorithm performed the best in ablation testing, with accuracy, recall, and F1 values of 0.984, 0.987, and 0.988, significantly better than models using CNN or CN alone. In addition, HBRACDP also exhibited excellent iterative performance in both training and testing sets. On the training set, HBRACDP only needed 228 iterations to reach a stable state with a fitness value of 0.17. In the test set, it only needed 182 iterations with a fitness value of 0.13. In contrast, the personalized book recommendation system proposed by Sarma D *et al.* [21] used algorithms based on CF and machine learning. Although it has improved in handling user preferences, it lacks in handling large-scale data and addressing data sparsity issues.

In practical application testing, HBRACDP showed significantly better book recommendation performance than other algorithms for 100 students. When N=70, the recommendation error of HBRACDP was the lowest, only 0.06%, and the recommendation accuracy was the highest, reaching 97.2%. In the average recommendation time tests of LTB and STB students, HBRACDP's recommendation times were 0.06 and 0.02, which were much lower than other comparative algorithms. This indicates that the HBRACDP algorithm not only performed well in benchmark testing, but also could quickly and accurately complete recommendation tasks in practical applications.

The proposed HBRACDP algorithm shows superior performance in the book recommendation task. However, compared with other algorithms, its computational complexity and resource requirements are higher. The HBRACDP algorithm combines CN and SAM, both of which require a lot of computational resources in the process of model training and prediction. CN's dynamic routing mechanism and SAM's multi-head attention mechanism increase the expressiveness of the model, but also significantly increase the computational complexity and memory consumption. In contrast, the traditional FPMC and LSTM algorithms have lower computational complexity and resource requirements and are more suitable for handling small-scale data sets and simple recommendation tasks.

In addition, although the HBRACDP algorithm has excellent performance in recommendation accuracy and time, it still has some limitations. First, high computational complexity and resource requirements may limit its popularity in large-scale practical applications. In future research, model pruning, knowledge distillation, and quantization techniques can be used to optimize the algorithm, reduce the computational overhead, and improve the inference speed. Secondly, the current algorithm mainly relies on borrowing behavior data of users. In the future, multi-modal data fusion technology can be introduced to further improve the recommendation effect by combining the information of users' social relations and interest labels.

In summary, HBRACDP has demonstrated strong capabilities in processing large-scale book data and capturing user behavior, providing new technical support and optimization ideas for university book recommendation. The disadvantage of the current research is that the evaluation index of the recommendation algorithm is not unified in the industry, and whether the book diversity evaluation index used in the research conforms to the actual situation still needs to be tested. In addition, the research regards the user's borrowing behavior directly as the user's interest, but in fact, the potential interest cannot be fully reflected by the displayed behavior. Therefore, how to design an index that can evaluate both accuracy and diversity is one of the future research directions. In future research, it is necessary to make a more full and in-depth analysis of the possible negative feedback information in the user behavior series.

4.4. Actual Impact

Handheld reading has become a popular habit among young people, making personalized book recommendation systems valuable for enhancing library services. Such systems can proactively suggest books that match readers' interests, reducing the idle time of less popular books and increasing the reuse rate of library collections. While most colleges and universities have digitized their libraries, offering features like precise, fuzzy, and categorized searches as well as popular book lists, these systems still require users to input specific search information. They lack the ability to provide personalized, diverse, and intelligent recommendations based on user characteristics and preferences.

Compared with a digital library, a smart library can fully perceive users' needs and actively provide users with personalized and intelligent services. For instance, e-commerce platforms use real-time click records, collections, and ratings to predict users' purchasing likelihood. However, university libraries often lack book browsing, click, and rating data, relying instead only on users' borrowing, search records, and basic user information. With advances in wearable devices and data storage, using readers' offline borrowing behavior information combined with historical data and extensive book information to provide personalized book recommendations is now feasible.

In the big data environment, wearable devices like smartphones and smartwatches can collect comprehensive user-book interaction data, such as borrowing location, browsing and copying information, and emotional information. In addition, Libraries also maintain vast knowledge resources and valuable borrowing data for teachers and students. This study combines these behavioral insights into a book recommendation system, creating personalized and diversified intelligent book recommendations that align with users' unique characteristics and preferences.

5. Conclusion

This study sets out to address the limitations of traditional book recommendation algorithms in the context of big data. We proposed the HBRACDP algorithm, a novel approach that integrates Capsule Networks and Self-Attention Mechanisms to model both short-term and long-term user borrowing preferences. Our algorithm demonstrated superior performance in both benchmark tests and practical applications.

In benchmark testing, HBRACDP significantly outperformed existing methods, achieving high accuracy (0.984), recall (0.987), and F1 value (0.988). More importantly, in real-world applications with 1000 university students, our algorithm showed remarkable efficiency and accuracy, with a recommendation time of just 0.09 seconds, an accuracy rate of 97.89%, and an error rate of only 0.08%. These results underscore the potential of HBRACDP to revolutionize book recommendation systems in university libraries and beyond. By more accurately capturing user preferences and efficiently processing large-scale data, our algorithm can significantly enhance user experience and resource utilization in digital library systems.

However, it's important to note that the computational complexity of HBRACDP may present challenges for implementation in some contexts. Future research should focus on optimizing the algorithm's efficiency, exploring its applicability in diverse recommendation scenarios, and investigating ways to incorporate additional types of user data while maintaining privacy. In conclusion, HBRACDP represents a significant step forward in big data recommendation systems, offering new possibilities for personalized and efficient information retrieval in the digital age.

References

[1] H. Zheng *et al.*, "Medication Recommendation System Based on Natural Language Processing for Patient Emotion Analysis", *Academic Journal of Science and Technology*, vol. 10, no. 1, pp. 62–68, 2024. http://dx.doi.org/10.54097/v160aa61

[2] X. Xia, "Learning Behavior Mining and Decision Recommendation Based on Association Rules in Interactive Learning Environment", *Interactive Learning Environments*, vol. 31, no. 2, pp. 593–608, 2023. http://dx.doi.org/10.1080/10494820.2020.1799028

[3] G. Liu, "An Ecommerce Recommendation Algorithm Based on Link Prediction", *Alexandria Engineering Journal*, vol. 61, no. 1, pp. 905–910, 2022. http://dx.doi.org/10.1016/j.aej.2021.04.081

[4] S. Forouzandeh *et al.*, "A Hybrid Method for Recommendation Systems Based on Tourism with an Evolutionary Algorithm and Topsis Model", *Fuzzy Information and Engineering*, vol. 14, no. 1, pp. 26–50, 2022. http://dx.doi.org/10.1080/16168658.2021.2019430

[5] C. Iwendi *et al.*, "Pointer-based Item-to-item Collaborative Filtering Recommendation System Using a Machine Learning Model", *International Journal of Information Technology & Decision Making*, vol. 21, no. 1, pp. 463–484, 2022. http://dx.doi.org/10.1142/S0219622021500619

[6] Y. M. Latha and B. S. Rao, "Product Recommendation Using Enhanced Convolutional Neural Network for E-commerce Platform", *Cluster Computing*, vol. 27, no. 2, pp. 1639–1653, 2024. http://dx.doi.org/10.1007/s10586-023-04053-3

[7] W. Lin *et al.*, "A Deep Neural Collaborative Filtering Based Service Recommendation Method with Multi-source Data for Smart Cloud-edge Collaboration Applications", *Tsinghua Science and Technology*, vol. 29, no. 3, pp. 897–910, 2023. http://dx.doi.org/10.26599/TST.2023.9010050

[8] P. Chiranjeevi and A. Rajaram, "A Lightweight Deep Learning Model Based Recommender System by Sentiment Analysis", *Journal of Intelligent & Fuzzy Systems*, vol. 44, no. 6, pp. 10537–10550, 2023. http://dx.doi.org/10.3233/JIFS-223871

[9] S. Bhaskaran and R. Marappan, "Enhanced Personalized recommendation system for machine learning Public Datasets: Generalized Modeling, Simulation, Significant Results and Analysis", *International Journal of Information Technology*, vol. 15, no. 3, pp. 1583–1595, 2023. http://dx.doi.org/10.1007/s41870-023-01165-2

[10] Z. Li *et al.*, "Diffurec: A Diffusion Model for Sequential Recommendation", *ACM Transactions on Information Systems*, vol. 42, no. 3, pp. 1–28, 2023. https://doi.org/10.1145/3631116

[11] G. Wei *et al.*, "A Hybrid Probabilistic Multiobjective Evolutionary Algorithm for Commercial Recommendation Systems", *IEEE Transactions on Computational Social Systems*, vol. 8, no. 3, pp. 589–598, 2021. http://dx.doi.org/10.1109/TCSS.2021.3055823

[12] X. Wen, "Using Deep Learning Approach and IoT Architecture to Build the Intelligent Music Recommendation System", *Soft Computing*, vol. 25, no. 4, pp. 3087–3096, 2021. http://dx.doi.org/10.1007/s00500-020-05364-y

[13] T. Ma *et al.*, "Research on Diversity and Accuracy of the Recommendation System Based on Multi-objective Optimization", *Neural Computing and Applications*, vol. 35, no. 7, pp. 5155–5163, 2023. http://dx.doi.org/10.1007/s00521-020-05438-w

[14] V. R. Yannam *et al.*, "Improving Group Recommendation Using Deep Collaborative Filtering Approach", *International Journal of Information Technology*, vol. 15, no. 3, pp. 1489–1497, 2023. https://doi.org/10.1007/s41870-023-01205-x

[15] B. Bakariya *et al.*, "Facial Emotion Recognition and Music Recommendation System Using CNN-based Deep Learning Techniques", *Evolving Systems*, vol. 15, no. 2, pp. 641–658, 2024. https://doi.org/10.1007/s12530-023-09506-z

[16] J. Wu *et al.*, "A Graph-based CNN-LSTM Stock Price Prediction Algorithm with Leading Indicators", *Multimedia Systems*, vol. 29, no. 3, pp. 1751–1770, 2023. https://doi.org/10.1007/s00530-021-00758-w

[17] K. Al Fararni *et al.*, "Hybrid Recommender System for Tourism Based on Big Data and AI: A Conceptual Framework", *Big Data Mining and Analytics*, vol. 4, no. 1, pp. 47–55, 2021. http://dx.doi.org/10.26599/BDMA.2020.9020015

[18] W. Chen *et al.*, "Applying Machine Learning Algorithm to Optimize Personalized Education Recommendation System", *Journal of Theory and Practice of Engineering Science*, vol. 4, no. 1, pp. 101–108, 2024. https://doi.org/10.53469/jtpes.2024.04(01).14

[19] D. Xia *et al.*, "A Bidirectional-a-star-based Ant Colony Optimization Algorithm for Big-data-driven Taxi Route Recommendation", *Multimedia Tools and Applications*, vol. 83, no. 6, pp. 16313–16335, 2024. http://dx.doi.org/10.1007/s11042-023-15498-4

[20] Y. Li *et al.*, "Multimodal Deep Learning Framework for Book Recommendations: Harnessing Image Processing with VGG16 and Textual Analysis via LSTM-Enhanced Word2Vec", *Traitement du Signal*, vol. 40, no. 4, pp. 1367–1376, 2023. http://dx.doi.org/10.18280/ts.400406

[21] D. Sarma *et al.*, "Personalized Book Recommendation System Using Machine Learning Algorithm", *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 1, pp. 212–219, 2021. http://dx.doi.org/10.14569/IJACSA.2021.0120126

Received: June 2024
Revised: September 2024
Accepted: September 2024

Contact addresses:
Jiali Liao
School of Information Engineering
Sichuan Top IT Vocational Institute
Sichuan
China
e-mail: JialiLiaoscetop@163.com

Tianxiang Li*
School of Information Engineering
Chengdu Industry and Trade College
Sichuan
China
e-mail: litx2006@163.com
*Corresponding author

Jiali Liao received the BSc degree in software engineering from China West Normal University, in 2015. She is currently a lecturer at the School of Information Engineering, Sichuan Top IT Vocational Institute in Sichuan, China. Her interests include big data, artificial intelligence, and software technology.

Tianxiang Li received the MSc degree in software engineering from Sichuan University, in 2013. He is currently an associate professor at the School of Information Engineering, Chengdu Industry and Trade College in Sichuan, China. His interests include big data, software technology, and industrial Internet applications.