Automation of Book Categorisation Based on Network Centric Quality Management System

Tingting Liu[1]*, Qiyuan Liu[2], Linya Fu[3]
Library, Harbin University, Harbin 150000, China[1]
Changchun Sixth High School, Changchun 130000, China[2]
Fenghua Middle School, Harbin 150000, China[3]

***Abstract*—In order to improve the efficiency of automatic book classification, the study uses a crawler to crawl book data from regular websites and perform data cleaning and fusion to build a structured knowledge graph. Meanwhile, the processed data is applied to a pre-trained model to improve it, and migration learning is used to improve the results. Fusion of Multiple Attention Mechanisms, Recurrent Neural Network, and Convolutional Neural Network modules into the classification model and feature fusion is used to enhance feature extraction. In addition, the study designed a pre-trained model architecture to help automatically categorise and manage book resources. The results of this study show a significant improvement in the classification of Chinese books on the Chinese Book L2 Subject Classification, iFlytek, and THUCNews datasets with significant performance improvement. The fusion of long and short-term memory and convolutional network Transformer-based bi-directional encoding models improved the accuracy by 0.19%, 1.54% and 0.42% on the two datasets, respectively, while the weighted average F1 scores improved accordingly. Through wireless technology, the automatic classification efficiency of books is realized and the management ability of the library is improved.**

Keywords—Crawler; books; automated; classification; Recurrent Neural Network; Multiple Attention Mechanism; knowledge graphs

I. Introduction

In the context of the big data era, the huge amount of complex text data makes libraries lack detailed subject classification labels when purchasing books, which leads to the inefficiency of acquiring book resources. Currently, this task mainly relies on library managers, which is costly and time-consuming. The introduction of an intelligent model greatly improves the accuracy of book categorisation, which helps to reduce labour costs and improve work efficiency. The robots not only automatically identify and classify books, but also operate continuously, significantly optimising library resource management and user services [1-2]. Existing studies rarely consider subject-specific library classification work, and the use of the latest natural language processing techniques, including large-scale pre-trained models such as the Bidirectional Encoder Representations from Transformers (BERT) model, can significantly improve the efficiency and accuracy of automatic classification of library resources in subject domains [3]. Such models can effectively reduce the dependence on a large amount of manually labelled data by virtue of their powerful semantic capture and feature extraction capabilities. Meanwhile, Knowledge Graph, as an intelligent semantic tool, greatly facilitates information sorting and management by interconnecting dispersed data through constructed entities, relationships and attributes [4]. Although the approach limits the application of the system across domains, it tends to show better performance and efficiency within established domains.

Network Centric Quality Management (NCQM) systems often use web-based technologies to collect, analyse and manage quality-related data to improve the efficiency and effectiveness of the overall system or process [5]. In the context of intelligent model book categorisation, NCQM systems can be used to monitor and optimise robot performance, accuracy and efficiency. Therefore, in order to improve the efficiency and accuracy of book management, reduce the dependence on paper records and manual operations, and improve the convenience of users to obtain information, the following work have been made in this study. First of all, the data is collected by efficient crawler technology, and then sorted and cleaned as the basis for the construction of a knowledge map. Based on Neo4j graph database, the book information knowledge graph is quickly constructed according to the pre-defined book knowledge graph Schema. Secondly, the pre-trained data set is divided into training set, verification set and test set, and the transfer learning strategy is used to fine-tune the pre-trained model to achieve the optimal state. By comparing with Text Convolutional Neural Network (Text-CNN) and other baseline models, the proposed model is fully tested on the Chinese book dataset specially constructed for this study. In addition, a feature enhancement method is proposed. This approach combines the multi-head attention mechanism of pre-trained models with the feature extraction and learning advantages of Convolutional Neural Networks (CNN) and Recurrent Neural Network (RNN). The pre-training model of fusion feature extraction technology has an effective effect on the classification of Chinese books. On the basis of using the pre-training model to classify Chinese books, according to the actual needs of university libraries, an automatic classification system of book resources based on the pre-training model is completed.

The study is divided into six sections. Section I discusses the challenges libraries face in the era of big data and proposes the use of intelligent models and natural language processing techniques to improve classification accuracy and efficiency. Section II reviews the research progress of automatic book

*Corresponding Author.

classification, and mentions the application of BERT pre-training model and knowledge graph in information classification. Section III introduces the research methods, including the collection and processing of book data and the construction of a knowledge graph, the collection and storage of data in Neo4j graph database by web crawler technology, and the application of BERT model for transfer learning and feature fusion. Section IV tests the models and compares the classification effects of different models. Section V and Section VI summarize the research results, confirm the effectiveness of the intelligent model in improving classification efficiency and accuracy, and propose future work directions, such as strengthening the subject classification module and improving user experience.

II. Related Work

The application of intelligent model in book categorisation not only improves the efficiency of library management but also provides better reading services to the users. An innovative feature selection optimisation algorithm has been designed by Janani and Vijayarani. To validate the effectiveness of the algorithm, the researchers further proposed a machine learning-based automatic text categorisation algorithm. The results demonstrated the efficiency and accuracy of the algorithm in extracting key features and classifying text documents by content [6]. Zhang designed a classification system based on support vector machines (SVM) for intelligent text classification, which is dedicated to public security information. The results proved the efficiency and practicality of SVM in dealing with specific information classification tasks [7]. For the field of automated text categorisation, Rezaeian and Novikova have compared the efficacy of plain Bayes and support vector machine algorithms, as well as Gaussian kernel function, polynomial kernel function, and Bernoulli's model in plain Bayes in order to enhance the accuracy of Persian textual materials in categorisation. The results of the study showed the better performance of these methods in Persian text classification [8]. Dizaji and other researchers proposed to combine the imperialist competitive algorithm with support vector machines for text classification. After experimental analyses, the results showed that this method showed higher efficiency and accuracy in text classification tasks [9].

The best performing language models for various tasks in the natural language processing direction are pre-trained language models. However, focusing on the topology of knowledge graphs often ignores the potential differences between knowledge graph embeddings and natural language embeddings, which limits the ability to reason effectively using both implicit and explicit knowledge. To address this problem, Cao and Liu designed an innovative model, ReLMKG, which combines pre-trained language models and relevant knowledge graphs. The efficiency and applicability of the model is demonstrated by testing it on the complex WebQuestions and WebQuestionsSP datasets [10]. According to the characteristics of Chinese library classification, Yuhui Z and other scholars used innovative methods to adapt to the problems of extreme multi-label (XMC) and hierarchical text classification (HTC). This paper extracts semantic features by a lightweight deep learning model and combines hierarchical information and other features with a learning ranking (LTR) framework to improve accuracy and classification depth. This model not only understands deep semantics but also has interpretability, is easy to expand and customize, and is suitable for processing tens of thousands of category labels, providing a solid foundation for comprehensive deep classification [11]. Jiang Y et al. combined domain-specific and general text enhancement strategies, such as category mapping and bilingual theme-based method, to solve the problem of small data volume and unbalanced categories in English books with Chinese picture classification numbers, and added punctuation and conjunction to the English text. Experimental results show that this hybrid strategy can improve the model performance. The visualization of BERT word vectors and the analysis of word information entropy can be applied to the classification and recognition of books [12]. To improve the accuracy of Chinese text Classification, Liu X and other researchers designed Chinese Library Classification based on an adaptive feature selection algorithm and tested a variety of Chinese text types. In this paper, an improved mutual information chi-square algorithm is introduced, which combines word frequency and term adjustment, term frequency-inverse document frequency method, and uses the limit gradient enhancement algorithm to improve the word filtering effect. Experiments show that the proposed algorithm can effectively improve the classification performance of different news texts, but the optimal algorithm selection varies according to the text type [13].

In summary, existing models perform well in specific domains, but are less adaptable across domains. For example, some Chinese book classification models have limited ability to generalize on other languages or topics. Despite the introduction of methods such as Multiple Attention Mechanisms, RNNS, and CNNS, feature extraction and fusion are still inadequate, especially when dealing with long text, which may miss key information or be inefficient. In addition, the lack of a large-scale open-source book knowledge graph limits the model's knowledge reasoning and classification accuracy. Training deep learning models requires a lot of resources and time, which small organizations can't afford. Model tuning and parameter adjustment also require a lot of experiments, and the number of books in different categories is unbalanced, resulting in poor classification performance, especially in small categories. Most methods do not perform well in cross-domain book classification, the model depends on the domain, and the application scope is limited. In order to achieve effective text representation, feature extraction and accurate subject categorisation of book resources.

III. Automated Research on Categorisation of Books by Intelligent Model Based on NCQM System

Due to the lack of available open source book knowledge graph and the data requirement of deep learning model, the research first designed an asynchronous anti-anti-crawler based on aiohttp to collect website book information. The collected data was cleaned and used to build the knowledge graph, and the book information knowledge graph was built automatically by using Neo4j. In order to deal with the low efficiency of traditional book classification and the need for professional knowledge, the research adopts the pre-trained model in deep learning for text representation, and optimizes the model structure through transfer learning. The pre-processed data set was divided into training, verification and test sets, and the model was fine-tuned to compare with the existing baseline models such as TextCNN and TextRNN. RNN module and CNN module are added to the feature extraction layer of the book classification model, and a feature fusion method is proposed to enhance the features of the pre-trained model. To improve the accuracy of Chinese book classification. According to the actual demand of university library, the automatic classification system of book resources based on pre-training model is finally completed.

A. Construction of Knowledge Graph of Book Information

The core advantage of Knowledge Graph is that it can connect related data elements, and its construction is closer to the natural logic of human processing information [14]. Moreover, knowledge graph relies on the use of graphs to store data, which makes it possible to achieve good data visualisation. There are two main methods for storing knowledge graphs, the first is the resource description framework introduced by W3C, and the second is to use of graph databases for data preservation. Graph databases are popular among developers and users because of their intuitive data representation and visualisation features, and they are not inferior to RDF in terms of data query performance. Therefore, the vast majority of developers tend to use graph databases to build knowledge graphs. Fig. 1 shows the construction steps of the book knowledge graph.

In Fig. 1, this construction step first collects data from multiple online resources through web crawling techniques and then pre-processes them to remove redundancies and error messages. Next, the processed data is subjected to knowledge extraction using a predefined schema in the study, i.e., converting the textual information into a structured ternary form. Subsequently, a knowledge fusion step is carried out to solve the problems of word polysemy and textual ambiguity by technical means. Ultimately, using Neo4j, a graph database platform, the collated ternary data is constructed into an exhaustive knowledge graph of book information and its visualisation is achieved to facilitate a more intuitive understanding and analysis of the data. In view of the characteristics of the website where the data comes from, it is necessary to design a crawler programme to obtain the data quickly. In terms of language platform selection, Python language is chosen as the platform for building the crawler programme [15]. After choosing the platform and tools, the design of the crawler programme begins, and the running flow of the crawler programme is shown in Fig. 2.

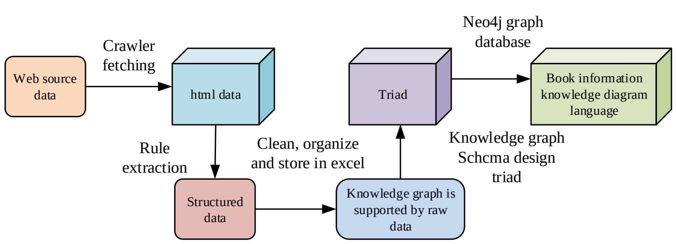


Fig. 1. Flowchart of the construction of book knowledge map.

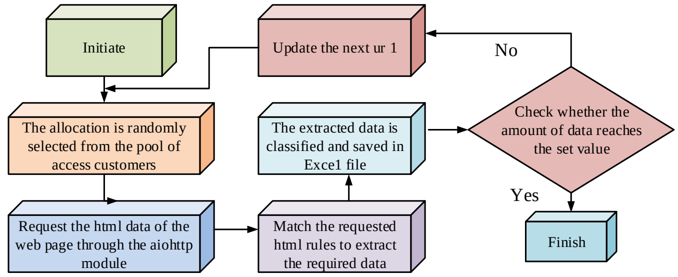


Fig. 2. Crawler operation flow.

According to the flow in Fig. 2, first, the crawler makes use of the asynchronous HTTP web module aiohttp to initiate HTTP requests against the specified book information website URL and obtain the HTML content. The reason for choosing aiohttp as the request tool is that it supports asynchronous processing of HTTP requests, which significantly improves the efficiency of the crawler. Given the large amount of data required to build a knowledge graph of book information, traditional synchronous crawling methods are less efficient, while aiohttp can accelerate the data crawling process. Once the HTML data is successfully retrieved, the program will analyse its content and extract the required textual information. In addition, the study employs the BeautifulSoup tool to crawl the required content from web pages quickly and efficiently. Relying on HTML's Selector for precise positioning, the extraction of target information is achieved. Initially, the acquired data may contain non-targeted components, so it is decided whether to use regular expressions to further remove depending on the situation [16]. Afterwards, it is processed and saved in xlsx format using Python's Xlsxwriter library, a process that converts semi-structured information in HTML into structured data. In constructing the knowledge graph of book information, the study uses Python's py2neo library in combination with the Neo4j graph database. Firstly, pandas are used to extract information from book data in xlsx format, and subsequently entities and relationships are created by py2neo. In order to cope with the polysemy and translation problems in Chinese, translation software is used to translate and staging the foreign language content. When merging entities, if a translated entity is found to have the same primary key as an existing entity, it will be manually reviewed to determine if it should be merged.

B. Automatic Book Classification Method Based on Pre-Trained Model

After initial processing, book title, synopsis and keyword texts are selected for classification. These data were transformed into vectors by a pre-trained model to be adapted for machine processing. The text features are refined through a multi-layer Transformer structured encoder. Subsequently, the learnt text features are fed into the classification system to compute probability values for different classification labels [17]. The final probability distribution is determined by a normalisation method and the label corresponding to the highest probability is assigned as the subject category of the book. The process of Chinese book classification is shown in Fig. 3.

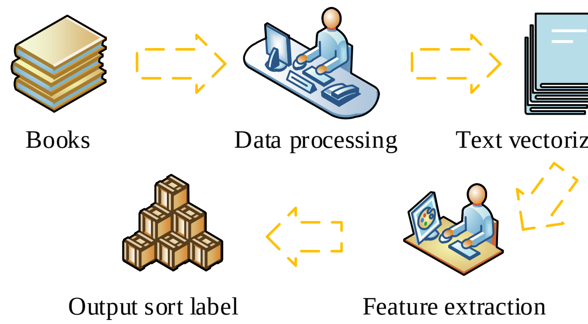


Fig. 3. Process of Chinese book classification.

BERT is a deep learning model based on Transformer, which mainly consists of bi-directional encoders. It is available in two versions, the basic version with 12 layers of encoders and the advanced version with 24 layers [18]. These layers include a multi-head self-attention mechanism and a feed-forward network, as well as residual connectivity and layer normalisation. In pre-training, BERT processes two pieces of text and learns two tasks, next-sentence prediction and masked language modelling. These tasks help BERT to better understand the language structure. Among them, the mask language model is suitable for single or double-paragraph text, while the next sentence prediction is specifically used for two paragraphs. The input representation of BERT consists of the sum of word vectors, block vectors, and position vectors [19]. In order to facilitate the computation, the dimensions of these three vectors are set to e, and the representation of the text is shown in Eq. (1).

$$v = v_t + v_s + v_p \tag{1}$$

In Eq. (1), v_t, v_s, v_p denotes word vector, block vector and position vector respectively, and the length of all three vectors is N. The word vector representation is very similar to the word vector in neural networks, both of which convert a text sequence into a fixed dimensional vector of values. Consider a dictionary with $|\Upsilon|$ words and an e-dimensional word vector space, where the sequence representation x is uniquely encoded into $e^t \in \mathbb{R}^{N\times|\Upsilon|}$ and $W^t \in \mathbb{R}^{e\times|\Upsilon|}$ represents the learnable word embedding matrix. Then, the word vector formula corresponding to x is given in Eq. (2).

$$v_t = e^t W^t \tag{2}$$

The role of the block vector is to tell the model which block the current word belongs to. And the role of the position vector allows the model to obtain the absolute position of each word in the sequence, which enhances the memory ability of the model. Applying BERT to the task of single-sentence text classification, the model is mainly composed of an input layer, a BERT feature extraction layer and a classification output layer. For the input text sequence $x_1 x_2 x_3 \cdots x_n$, the specific markers of the BERT model are added to both ends of the sequence to get the original input of the model X as shown in Eq. (3).

$$X = [CLS]\, x_1 x_2 x_3 \cdots x_n\, [SEP] \tag{3}$$

Next, the mapping of X is performed according to the above processing of word vectors, block vectors, and position vectors, and let the length of the sentence be n, then the input representation of the model is obtained as v, see Eq. (4).

$$v = \text{InputRepresentation}(X) \tag{4}$$

BERT utilises Transformers encoders with a different number of layers (12 layers for the basic version and 24 layers for the large version) to process the input v and learn the implicit associations between lexical items within the text. Let the implied dimension be d and the textual contextual representation be denoted as $h \in \mathbb{R}^{N\times d}$, corresponding to Eq. (5).

$$h = \text{BERT}(v) \tag{5}$$

The BERT model represents the whole sentence vector for next sentence prediction via [CLS] labelling in the pre-training phase. For task alignment, the implicit layer representation of [CLS] is similarly utilised as the vector representation of the whole sentence for single-sentence classification, denoted as h_0. To obtain the corresponding categorical labels of the input sentences are simply the [CLS] implicit layer representation h_0 extracted by BERT into the fully connected layer. Setting K as the number of categories, $W^0 \in \mathbb{R}^{d\times K}$ as the weights, and $b^0 \in \mathbb{R}^K$ as the bias, the label probability distribution $p \in \mathbb{R}^K$ is obtained as shown in Eq. (6).

$$p = Soft\max\left(h_0 W^0 + b^0\right) \tag{6}$$

After obtaining the probability distribution of the classification, it is compared with the real classification labels to calculate the loss, and then backpropagation is performed to update the parameters of the model. Conditional Random Field (CRF) is mainly applied to sequence labelling task to ensure the accuracy of the labelling [20]. The correct sequence is identified by scoring each labelled sequence and calculating the proportion of the total score that a particular labelled sequence scores. The length of the text sequence is l and the length of its true annotation sequence is also l. The total score S_{i,y_i} for position i in the sequence annotation consists of the sum of the current launch score R_{i,y_i}, the total score $S_{i-1,y_{i-1}}$ for the previous position, and the transfer score T_{y_{i-1},y_i}. The total score of the end point represents the score of the whole sequence, which is calculated in Eq. (7).

$$S(X, y) = \sum_{i=1}^{n} \left(T_{y_i, y_{i+1}} + R_{i,y_i}\right) \tag{7}$$

After obtaining the score of a single sequence annotation, in order to confirm whether a sequence annotation is optimal or not, it is necessary to calculate the proportion of its score to the sum of the scores of all possible sequences, and the expression of this proportion is shown in Eq. (8).

$$P\left(y_{now}\middle|X\right)=\frac{e^{S(X,y_{now})}}{\sum_{y\in Y_x}e^{S(X,y)}} \tag{8}$$

In Eq. (8), the larger P is, the closer the current sequence prediction is to the real value. Therefore, Eq. (8) can be used as the loss calculation of CRF model, and its calculation procedure is shown in Eq. (9).

$$\log\left(P\left(y_{now}\middle|X\right)\right)=S\left(X,y_{now}\right)-\log\left(\sum_{y\in Y_x}e^{S(X,y)}\right) \tag{9}$$

Typically, the CRF uses the Viterbi algorithm to find the optimal solution, labelled as the optimal sequence of X. This formula is shown in Eq. (10).

$$\overset{*}{y}=\arg\max_{\tilde{y}\in Y_X} S\left(X,\tilde{y}\right) \tag{10}$$

C. Fine-Grained Book Classification Based on Pre-Trained Model and Feature Fusion

Combining the feature extraction advantages of pre-trained models such as BERT, the newly designed text categorisation model integrates a feed-forward network, using the AdamW optimiser and cyclic unit, convolution and pooling techniques to enhance the feature representation. Based on these techniques, this paper proposes a feature enhancement method of Pre-trained model, PLM-LCN, which combines Long short term memory and Convolution Networks (LCN) and pre-trained language models (PLM). Fig. 4 shows the PLM-LCN model structure.

In Fig. 4, the main flow of the PLM-LCN method model is as follows, firstly, the original input text is encoded into word vectors, block vectors and position vectors, which are combined according to the rules to form the input representation of the Multihead Self-Attention Layer. The text features are extracted after processing in this layer, and the sentence representation with [CLS] characters enters into the convolution and loop unit module. The output features from the convolutional module and the bidirectional long and short term memory network (Bi-LSTM) are combined and spliced to form the ultimate textual representation. These features are used for probabilistic prediction by the classifier and normalised by Softmax, with the highest probability category being the book classification result. The counting formula for the convolution operation is given in Eq. (11).

$$O(i,j)=\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\left(I\left(i+m,j+n\right)K\left(m,n\right)\right) \tag{11}$$

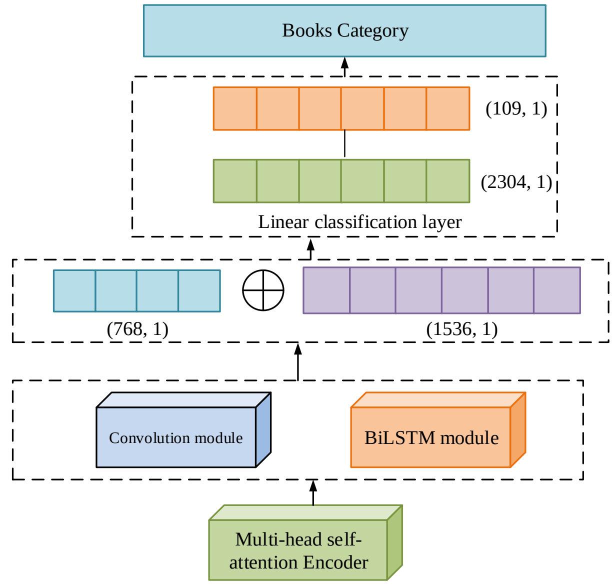


Fig. 4. PLM-LCN model structure.

In Eq. (11), (i,j) represents the output position. I represent the input data into the convolutional layer. K is the convolution kernel and (m,n) denote its size. Attention mechanism, derived from the human characteristic of focusing attention, is introduced into deep learning to highlight key features and downplay secondary information. In this mechanism, Q(Query) locates the target task, while K(Key) and V(Value) form matching pairs. Q It is used to find the corresponding V value in K. The operation of the mechanism and the computation process is shown in Eq. (12).

$$D_V=Attention\left(Q_t,K,V\right)=Soft\max\left(\frac{Q_t,K_s}{\sqrt{d_k}}\right)V_s=\sum_{s=1}^{m}\frac{1}{z}\exp\left(\frac{Q_t,K_s}{\sqrt{d_k}}\right)V_s \tag{12}$$

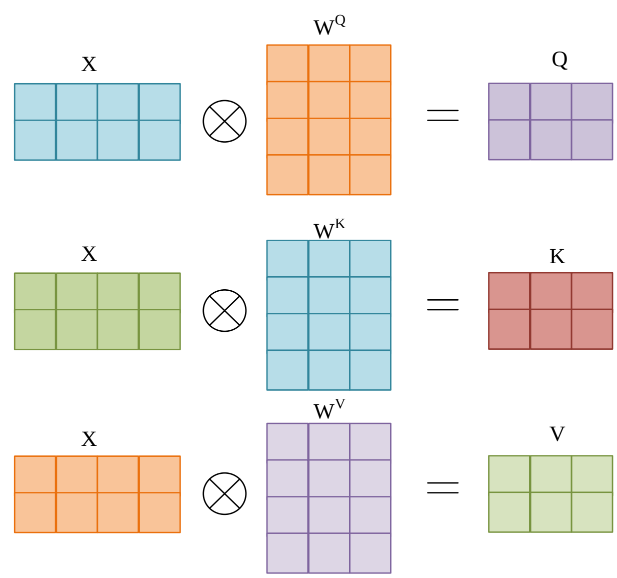


Fig. 5. Feature transformation process in self-attention calculation.

In Eq. (12), z represents the normalisation factor. Q_t represents the query value. The feature transformation process in Self-Attention computation is shown in Fig. 5.

Fig. 5 shows the matrix of word vectors X in the input text, representing a complete sentence. X Multiplying different weight matrices produces the Q,K,V matrix in the attention mechanism, where each row corresponds to a vector of words q,k,v. The parameters of these weight matrices W^Q,W^K,W^V are continuously updated during model training. In order to evaluate the model's learning of a -set of parameters, it is necessary to introduce a loss function, whose computational expression is shown in Eq. (13).

$$CE = -\frac{1}{m}\sum_{i=1}^{m}\sum_{j=1}^{c} y_j^{(i)} \log \hat{y}_j^{(i)} \tag{13}$$

In Eq. (13), $y_j^{(i)}$ denotes the true output result on the j th class for the i st sample, where only the correct class outputs 1 and the rest of the classes output 0. $\hat{y}_j^{(i)}$ denotes the probability that the model belongs to the j th class for the i th sample. Alternatively, Eq. (13) can be transformed into Eq. (14).

$$CE = -\frac{1}{m}\sum_{i=1}^{m} \log \hat{y}_t^{(i)} \tag{14}$$

In Eq. (14), $\hat{y}_j^{(i)}$ denotes the model's probability of predicting the first i sample on the correct category t. To realize the book information query and classification function of the automatic book classification robot system, the system is constructed into four levels. The overall framework of the book's automatic sorting robot is shown in Fig. 6.

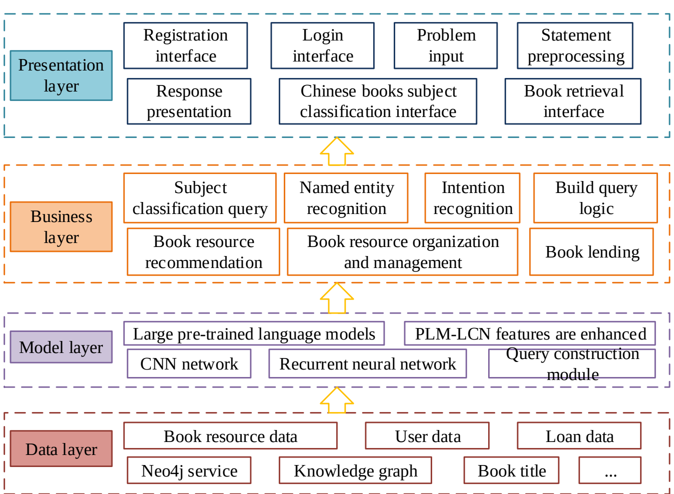


Fig. 6. Overall framework of the book automatic classification robot.

In Fig. 6, the presentation layer mainly realizes the communication between the system and the user. The user enters questions through the visualization window of this layer. The user interaction layer processes the natural language questions provided by the user to some extent, eliminates certain invalid interference content in the questions, and makes the text to be processed more standardized. The model layer calculates the data based on user commands and pushes the results to the presentation layer. The business layer is the bridge of user interaction scenario, covering various functional modules. The data interaction layer mainly relies on the constructed book information knowledge graph. After obtaining the query statement from the logical layer, the Neo4j server queries the book information knowledge graph. The limited number of candidate answers is returned to the user interaction layer after the candidate answers are sorted. The data interaction layer mainly relies on the book information knowledge graph constructed. After obtaining the query statement from the logical layer, the Neo4j server queries the book information knowledge graph. The limited number of candidate answers is returned to the user interaction layer after the candidate answers are sorted.

IV. Analysis of Experimental Results

Model training in deep learning research relies on a large number of matrix operations that place high demands on hardware. Graphics Processing Unit (GPU) is significantly more efficient in handling such tasks with its parallel processing capability and fast dedicated memory. The initial input length of all neural network models is set to 128 and the training period is 10 epochs. BERT-Base-Chinese and its improved version with AdamW optimiser and cross-entropy loss function are selected. Pytorch was chosen as the development framework, and its supporting hardware configuration and RSBC model parameter settings are shown in Table I.

TABLE I. Experimental Environment and Model Parameter Settings

Experimental Environment	Disposition	Model	Argument
Operating system	Windows 10	RoBERTa version	RoBERTa_zh_L12
CPU	Interl I7-11800H	learning_rate	3e-5
GPUs	NVIDIA GeForce RTX 3060	hidden_dim	768
RAM	16G	dropout_rate	0.5
Hard disk	10G disposable space	batch_size	64

The experiments use seven traditional neural network models as baseline models to participate in the comparison experiments. Accuracy (acc) represents the proportion of correctly classified samples in the total samples, and is a basic index for evaluating classification models. Macro Precision (mc_p) calculates and averages the accuracy of each class separately, reflecting the average accuracy of the model in each class, regardless of class imbalance. Macro Recall (mc_r) also calculates and averages the recall rate for each category separately, and measures the average recall rate of the model across all categories, ignoring the class imbalance problem. Macro F1-Score (mc_f) is the harmonic average of macro average accuracy and macro average recall, taking into account the performance of both accuracy and recall. Weighted

Precision (w_p) takes into account the number of samples in each category and makes a weighted average to reflect the comprehensive accuracy rate of the model in different categories, which is applicable in the case of unbalanced categories. Weighted F1-Score (w_fl) is also a harmonic average of weighted accuracy and weighted recall to comprehensively evaluate the model's performance in the case of class imbalance. The experimental results on different neural network models on the secondary subject classification dataset of Chinese books are shown in Fig. 7.

As can be seen from Fig. 6, each benchmark neural network model achieves more than 75% accuracy and average F1 score, indicating that the models show good performance after a number of training cycles. TextRCNN fails to outperform TextRNN. The Fasttext model performs better, with high speed and second only to the TextRNN-Att model. The loss values for the Fasttext model and the TextRNN-Att model has a loss value of 0.62 and 0.59 respectively, while the TextRNN model with embedded attention mechanism has the best result in all comparisons, which highlights the feature extraction advantage of the attention algorithm, and verifies the feasibility of studying the classification research with the pre-trained model based on the attention mechanism. The LERT model retains the original architecture of the BERT, but optimises the training tasks and methods, which provides a good opportunity to study the BERT series of models. The LERT model retains the original BERT architecture but optimises the training tasks and methods, providing guidance for the study of the BERT series of models. The learning rate is set to 1e-5, the batch size is 8, and 10 rounds of training are performed, taking the maximum sequence length of the input text as 16, 32, 64, 128, 192, 256, and 512 for the experiments, respectively. The experimental results corresponding to Fig. 8 are obtained.

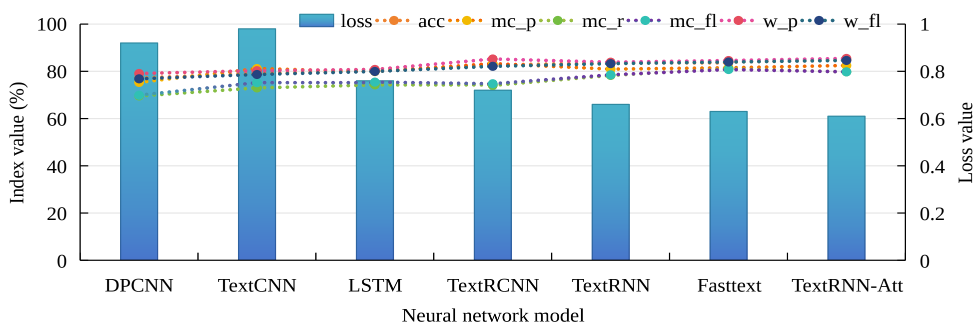
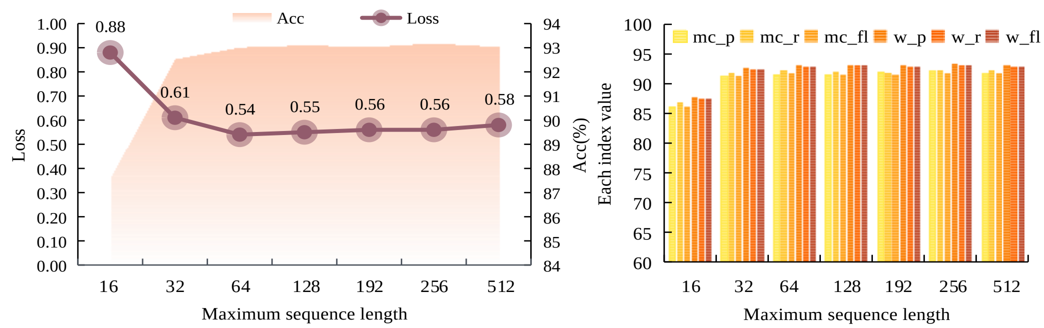


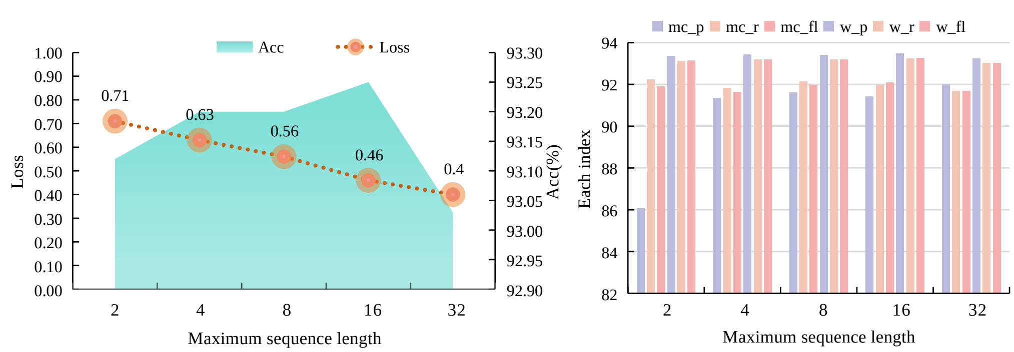
Fig. 7. Comparison of indicators of each model on the Chinese books' secondary subject classification data set.



(a) Loss and Acc of LERT model under different maximum sequence lengths

(b) Each index of LERT model under different maximum sequence lengths

Fig. 8. Experimental results of the LERT model under different maximum sequence lengths.



(a) Loss and Acc of LERT model under different batch sizes

(b) Indicators of LERT model under different batch sizes

Fig. 9. Experimental results of the LERT model under different batch sizes.

As can be seen in Fig. 7, the performance of the LERT model when dealing with different maximum sequence lengths, where the best performance is achieved when 256 is the sequence length. A sequence that is too short (e.g., 16) will lose more information because the important content is truncated, while setting it too long results in filling too many invalid 0-values, which affects feature extraction and computational efficiency. Therefore, setting 256 as the most suitable text length for the model to handle on the task of secondary subject classification of Chinese books is a significantly preferred solution. Fig. 9 shows the experimental results of the LERT model under different batch sizes.

Fig. 9 shows that the performance of the model typically improves with increasing batch size, although the effect decreases after a certain point. The minimum batch is theoretically good for optimisation, but too low may cause unstable convergence and prolong training time. On the contrary, too large batches tend to trigger memory overruns and impair accuracy. Considering the experimental results and computational efficiency, 16 is chosen as the preferred batch size setting for the BERT pre-training model on the Chinese book secondary subject classification dataset. Fig. 10 shows the experimental results of the LERT model at different learning rates.

As can be seen from Fig. 10, the LERT model performs similarly when the learning rate is set to 1e-5 and 5e-5, both outperforming other higher learning rate settings. A learning rate that is too small may slow down the training speed and cause the model to fall into a local optimum, while a learning rate that is too high may cause oscillations in the loss function and prevent the model from converging. Therefore, based on the trade-off between model performance and training efficiency, 1e-5 or 5e-5 is a more appropriate choice of learning rate to promote the model to achieve good training results on the Chinese book secondary subject classification dataset. Experiments are conducted using the Chinese book secondary subject classification dataset, the iFlytek dataset and the THUCNews dataset for ablation experiments on the five models, and the results are shown in Fig. 11.

As can be seen from Fig. 11, the BERT-LCN model exhibits better performance compared to the original BERT when dealing with the Chinese book secondary subject classification, iFlytek and THUCNews datasets. Its accuracy is improved by 0.19%, 1.54% and 0.42%, while the weighted average F1 score achieves an increase of 0.19%, 2.73% and 0.47%, respectively. Meanwhile, due to the original expressive power of the multiple self-attention mechanism, the BERT-LCN model demonstrated significant results on all three different attribute datasets reflecting its highly generalised nature. The ablation experiments were conducted using three datasets on six models and the results are shown in Fig. 12.

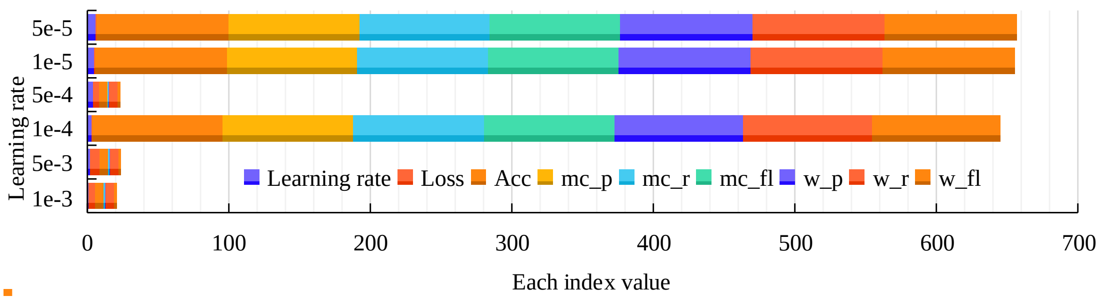
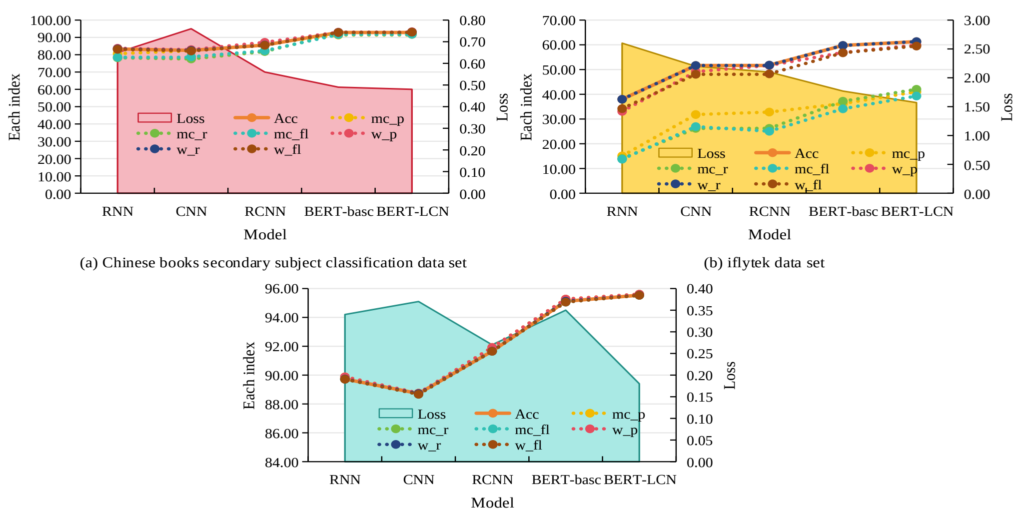


Fig. 10. Experimental results of the LERT model under different learning rates.

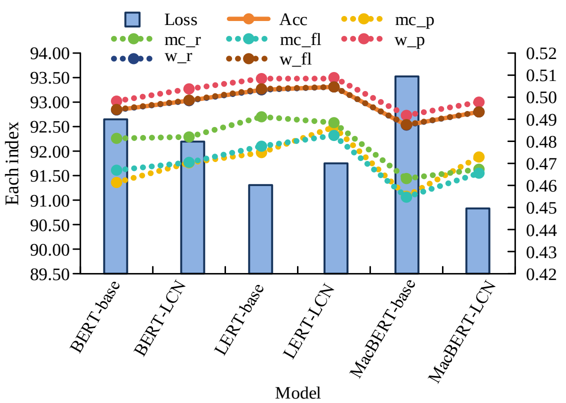


(a) Chinese books secondary subject classification data set

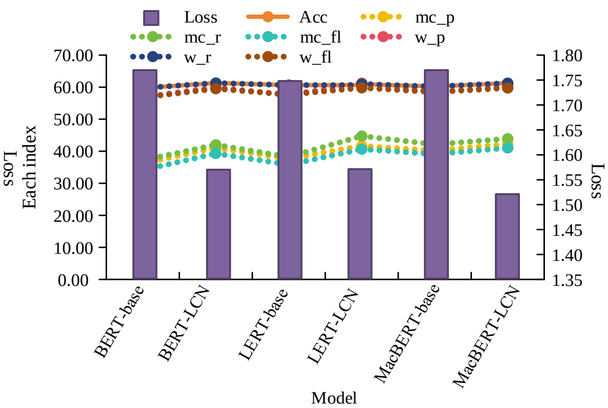
(b) iflytek data set

(c) THUCNews data set

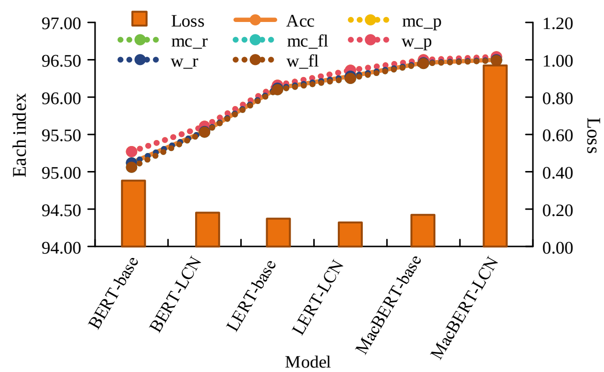
Fig. 11. Ablation results of the BERT-LCN method model on three datasets.



(a) Chinese books secondary subject classification data set



(b) iflytek data set



(c) THUCNews data set

Fig. 12. Experimental results of the PLM-LCN method model on three datasets.

As can be seen from Fig. 12(a), in the BERT-LCN, LERT-LCN and MacBERT-LCN models, the accuracy is increased by 0.18%, 0.07% and 0.28%, the weighted average F1 score is increased by 0.18%, 0.05% and 0.27%, respectively, and the macro-averaged F1 scores are increased compared to the original model. As can be seen from Fig. 12(b), the accuracy of these models increased by 1.55%, 0.32% and 1.04%, with weighted average F1 score enhancements of 2.74%, 2.13% and 1.28%, respectively, and the macro-averaged F1 scores likewise showed significant improvements. As can be seen in Fig. 12(c), these models showed increases in accuracy of 0.43%, 0.17% and 0.04%, and weighted average F1 score boosts of 0.47%, 0.15% and 0.04%, respectively. This difference in effect enhancement may be related to the characteristics of different datasets. In THUCNews and Chinese book subject classification, the baseline model has already achieved a high accuracy of more than 95%, so the room for improvement is relatively small. In contrast, the highest accuracy of the iFlytek baseline model is only 60.64%, so the performance enhancement of the LCN-enhanced model is more significant here.

To further verify the automatic method of book classification based on Network-centric quality management system proposed in this paper, it is compared with existing text classification methods, as shown in Table II.

In Table II, the feature extraction efficiency (96.49±0.74%), flexibility (94.23±0.37%) and accuracy (98.45±0.47%) of this method are superior to other literatures. The corresponding data in the study [6] were 87.37±0.64%, 88.47±0.68% and 86.46±0.42%. Study [8] were 87.97±0.47%, 87.26±0.59% and 89.68±0.73%. Study [10] were 90.67±0.14%, 90.63±0.62% and 92.67±0.36%. All P values were less than 0.05, indicating that the difference between different methods was statistically significant. In summary, the research method has the best performance in each index.

TABLE II. COMPARATIVE RESULTS OF VARIOUS BOOK CLASSIFICATION METHODS

Method	Feature Extraction Efficiency (%)	Flexibility (%)	Accuracy Rate (%)
Research method	96.49±0.74	94.23±0.37	98.45±0.47
Study [6]	87.37±0.64	88.47±0.68	86.46±0.42
Study [8]	87.97±0.47	87.26±0.59	89.68±0.73
Study [10]	90.67±0.14	90.63±0.62	92.67±0.36
P	<0.05	<0.05	<0.05

V. DISCUSSION

Compared with the feature selection optimization algorithm of Janani and Vijayarani et al. [6], this study can not only obtain a large amount of data quickly, but also ensure the high quality and comprehensiveness of the knowledge graph. In terms of feature fusion technology, the method proposed in this study is more flexible and scalable than the Bayesian algorithm and SVM used by Rezaeian and Novikova et al. [8] in Persian text classification, and is suitable for processing large-scale and complex data sets. The pre-trained model BERT was also used for optimization in combination with transfer learning. Although Cao and Liu's ReLMKG et al. [10] model combined the pre-trained language model and knowledge graph, this study further integrated RNN and CNN to make the model perform better in the task of Chinese book classification. Overall, this study shows significant advantages in data acquisition, feature extraction and model optimization, especially in the Chinese book classification task showing higher accuracy and efficiency.

VI. Conclusion

In the face of the challenge that the Chinese map classification no longer meets the needs of the emerging disciplines, the research proposes to adopt the existing natural language processing technology to solve the problem of classifying Chinese resources in libraries. The research team constructed a Chinese book database, used neural networks and pre-trained models for text analysis, feature extraction and category classification on this dataset, and created an NCQM-based solution to automate the archiving process with the help of an intelligent model. The results show that various benchmark neural network models achieved more than 75% accuracy and average F1 scores on the Chinese book classification dataset, indicating that the dataset is moderately difficult and the model training performs well. Specifically, the single-layer LSTM model outperforms the CNN in text sequence processing, while the TextRCNN does not exceed the performance of the TextRNN. The Fasttext model stands out with its high speed and excellent performance, second only to the TextRNN-Att model with embedded attention mechanism. For models using BERT-LCN, LERT-LCN, and MacBERT-LCN, the results show that they achieved improvements in accuracy and weighted average F1 scores compared to the original models on the Chinese book secondary subject classification, iFlytek, and THUCNews datasets. These enhancements reflect the generalisation ability and feature extraction advantages of the models when dealing with different attribute datasets. In particular, the BERT-LCN model, by combining the CNN and RNN modules, enhances the feature extraction capability and therefore shows significant performance gains on multiple datasets. In addition, there is still room for improvement in the research. In future work, it is planned that the subject classification module will be enhanced to provide diverse model choices and a top ten subject classification display, and the library seat reservation function will be added to improve user experience and convenience.

References

[1] Zeng L. Classification and English translation of book titles in Dunhuang documents. China Terminology, 2022, 24(3): 41-48.

[2] Wang J, Yue K, Duan L. Models and techniques for domain relation extraction: A survey. Journal of Data Science and Intelligent Systems, 2023, 3(1): 16-25.

[3] Li Y, Yang Y, Ma Y, Yu D, Chen Y. Text adversarial example generation method based on BERT model. Computer Application, 2023, 43(10): 3093-3098.

[4] Wang Y, Zhang X, Dang Y, Ye P. Knowledge Graph Representation of Typhoon Disaster Events based on Spatiotemporal Processes. Journal of Geo-Information Science, 2023, 25(6): 1228-1239.

[5] Biswash S K. Device and network driven cellular networks architecture and mobility management technique for fog computing-based mobile communication. Journal of Network and Computer Applications, 2022, 200(4): 1-16.

[6] Balakumar J, Vijayarani S. Automatic text classification using machine learning and optimisation algorithms. Soft Computing, 2021, 25(2): 1129-1145.

[7] Zhang L. Implementation of classification and recognition algorithm for text information based on support vector machine. International Journal of Pattern Recognition and Artificial Intelligence, 2020, 34(8): 1-16.

[8] Rezaeian N, Novikova G. Persian text classification using naive bayes algorithms and support vector machine algorithm. IAES Indonesia Section, 2020, 8(1): 178-188.

[9] Dizaji Z A, Asghari S, Gharehchopogh F S. An improvement in support vector machines algorithm with imperialism competitive algorithm for text documents classification. Signal and Data Processing, 2020, 17(1): 117-130.

[10] Xing C, Yun L. ReLMKG: reasoning with pre-trained language models and knowledge graphs for complex question answering. Applied Intelligence: The International Journal of Artificial Intelligence, Neural Networks, and Complex Problem-Solving Technologies, 2023, 53(10):12032-12046.

[11] Yuhui Z. A study of automated deep classification of literature based on Chinese library classification. Libraly Journal, 2024, 43(395): 61-74.

[12] Jiang Y. English books automatic classification according to CLC. Beijing Da Xue Xue Bao, 2023, 59(1): 11-20.

[13] Liu X, Wang S, Lu S, Yin Z, Li X, Yin L, Tian H W, heng, W. Adapting feature selection algorithms for the classification of Chinese texts. Systems, 2023, 11(9): 483-483.

[14] Suganya G, Mariappan P, Dubey P, Drolia A R, Srihari S. Subjective areas of improvement: A personalised recommendation. Procedia Computer Science, 2020, 172: 235-239.

[15] Sun J, Zhu M, Jiang Y, Liu Y, Wu L. Hierarchical attention model for personalised tag recommendation. Journal of the Association for Information Science and Technology, 2021, 72(3): 173-189.

[16] Zhou Y. Design and implementation of book recommendation management system based on improved apriori algorithm. Intelligent Information Management, 2020, 12(3): 75-87.

[17] Li G, Zhuo J, Li C, Hua J, Yuan T, Niu Z, Ji D, Wu R, Zhang H. Multi-modal visual adversarial Bayesian personalized ranking model for recommendation. Information Sciences, 2021, 572(1): 378-403.

[18] Zhang X, Liu X, Guo J, Bai W, Gan D. Matrix factorization based recommendation algorithm for sharing patent resource. IEICE Transactions on Information and Systems, 2021, E104. D(8): 1250-1257.

[19] Aljunid M F, Manjaiah D H. Multi-model deep learning approach for collaborative filtering recommendation system. CAAI Transactions on Intelligence Technology, 2020, 5(4): 268-275.

[20] Wang X, Ma W, Guo L, Jiang, H, Liu F, Xu C. HGNN: Hyperedge-based graph neural network for MOOC Course Recommendation. Information Processing & Management, 2022, 59(3): 1-18.