Sparse Information Filtering for English Language Repositories Using Multilevel Interactive Attention Mechanism

Tingting Fan, Shuang Zhang*
Department of Public Course Teaching, Nanyang Vocational College of Agriculture, National Highway 338, Wolong Distric, Nanyang City, Henan Province
Email: ftt19895566@163.com, zs1556011@163.com
*Corresponding author

Keywords: multi-level interaction, attention mechanism, English language repositories, sparse information, filtering recommendation, Top-K recommendation

Received: February 21, 2025

English learning resources involve a large amount of language knowledge and semantic information, and users' English learning needs are multidimensional, and these needs will dynamically change with the learning process and time. It is prone to insufficient feedback and evaluation information, with a large amount of sparse information, making it difficult to accurately analyze user learning resource preferences. How to capture and adapt to the multi-dimensional demand characteristics of users in real-time is a challenge faced by recommendation systems. To this end, research is conducted on a sparse information filtering recommendation algorithm for English resource libraries that integrates multi-level interactive attention mechanisms. Filter sparse interval data through FCM membership threshold ($0.2<\mu<0.8$), extract core English resource features, and then use a multi-level interactive attention mechanism to hierarchically extract preference features from the user layer (7 types of features such as age/interest) and resource layer (7 types of features such as listening/writing). After feature fusion, use Top-K method to calculate resource similarity and generate a recommendation list. Experiments have shown that on a dataset of 1500 resource items, 700 users, and 50000 ratings, the algorithm achieves significantly better performance in three key indicators: consistency in preference feature extraction (0.949-0.968), resource coverage ($\geqslant 0.9$), and conversion rate (96.83%) compared to the baseline model (LSTM model conversion rate 39.35%), and a 2.6-fold increase in detail page clicks. By dynamically capturing the user resource interaction relationship, the algorithm achieved an accurate proportion matching of 0.868 (with a deviation of 0.002) in the recommendation of listening resources, verifying its superiority in multi-dimensional dynamic demand scenarios.

Povzetek: Predstavljena je kombinacija FCM-klastriranja in večnivojske interaktivne pozornosti za iskanje v angleških repozitorijih. Dosežena je večja konsistentnost, pokritost in konverzija kot pri LSTM pristopu.

1 Introduction

In the era of information explosion, the English resource library, as an important platform for learners to acquire English knowledge, faces two quantifiable core challenges in its recommendation algorithm: first, the high sparsity of user resource interaction data (measured matrix filling rate is less than 5%), and second, the response delay of dynamic learning requirements (traditional models have an average deviation of 0.12) [1-3]. Therefore, how to effectively use recommendation algorithms to filter out valuable information from the massive English resources and provide users with personalized learning experience has become an important issue in the construction and development of English resource library [4].

Although existing recommendation algorithms have their own focuses, they have not effectively solved these two coupling problems: Mohammadi et al. study location-sensitive and user-preference based perceptual recommendation algorithms, which take into account the user's current location or historical location data to recommend locations or services that meet the user's needs. Location data can reflect the user's range of activities and points of interest, thus helping the recommendation algorithm to more accurately predict the content that the user may be interested in. In addition to location information, the algorithm will also analyze the user's behavior and preferences, such as the user's past evaluation of certain locations or services, frequency of visits, and so on, in order to build a model of the user's interests. This model will be used to predict what users may be interested in in the future. However, users' interests and preferences may change over time, and this algorithm may not be able to capture these changes in a timely manner. The response to user interest drift may have a lag of more than 48 hours, resulting in recommendation results that do not match the user's current needs [5].

Benabes et al. study a resource recommendation algorithm based on user comments. This algorithm

collects user comments on resources, which contain users' evaluations, feelings, suggestions, etc. The keywords, emotional tendencies, and other features are extracted and used to construct a user profile. The keywords, emotional tendencies and other features of the comments are extracted and used to construct user profiles. After constructing the user profile, the algorithm generates a personalized recommendation list for the user based on the degree of match between the user profile and the resource features. The resources in the recommendation list are usually highly relevant to the preferences and needs expressed in the user's historical comments. However, for new users or scenarios with fewer resources, the comment data may be very sparse, making it difficult for the algorithm to build an accurate user profile. The accuracy of this method, which relies on user comments, drops sharply by 62% in cold start scenarios (user comments<3), and data sparsity can reduce the accuracy and personalization of recommendation algorithms [6].

Chakaravarthi et al. study resource recommendation algorithms based on long and short-term memory, collect users' historical behavioral data, such as browsing records, click records, purchase records, etc., and resource feature information. These data are converted into numerical form, and the LSTM neural network model is used to learn the user's historical behavioral data, capture the long-term dependency relationship of the user's interest, and generate a personalized resource recommendation list for the user based on the similarity or matching degree. However, in the case of sparse data, the LSTM model may be more prone to overfitting phenomenon. Even if the model performs well on training data, it does not perform well on test data or real applications. This is because the model relies too heavily on limited training data, resulting in 73% of mismatches due to overfitting of sparse data, and cannot generalize to a wider range of user behaviors [7].

Albert et al. study a data recommendation model based on item ratings and user trust. In terms of item ratings, the model utilizes users' numerical ratings and textual comments on items, calculates the similarity between items, and constructs a user-item ratings matrix, thus reflecting users' preferences for items. And the trust connection between users is captured through social networks, user behavior or explicit trust statements. This trust relationship plays a role in enhancing the reliability and personalization of the recommendation results in the recommendation system. When users trust a certain other user, they may be more inclined to accept recommendations from that user. However, the model also faces some challenges. One of the main difficulties is the data sparsity problem, i.e., most users only evaluate a few items, which results in a highly sparse user-item rating matrix. In addition, due to the sparsity of the scoring matrix (density<2%), the coverage rate is always below the practical threshold of 0.6 [8].

Wang proposed to construct a personalized association recommendation model by integrating association rule mining with Bayesian networks, improving traditional association rule mining algorithms. Combining user history records for pruning processing, frequent itemsets are filtered out, and itemsets below the threshold are pruned. The pruned itemsets are then input into a Bayesian validation network for personalized validation, and finally recommended to truly interested learners based on the ranking results. In the data processing stage, although threshold pruning and Bayesian validation have alleviated the problem of data sparsity, the recommendation coverage still drops below 0.55 when the user item interaction matrix density is less than 1.5% [9].

Hien et al. developed a recommendation model that combines convolutional neural networks (CNN) and matrix factorization (MF) to filter and classify important content from massive amounts of information by integrating multidimensional data such as user preferences, interests, and behaviors. This model utilizes CNN to capture local features of images or text, combines MF to construct the correlation between users and items, and introduces additional information and rating bias between products and users during the training process to improve recommendation accuracy and contextual understanding ability. Although the model alleviates data sparsity through CNN-MF fusion, its computational efficiency is still limited by the dimensionality scalability of matrix decomposition in ultra large datasets such as billion level user project interactions [10].

In response to the above issues, this article proposes a new recommendation framework that integrates FCM clustering and multi-level attention. This method first filters sparse interval data (with a deviation of 0.002) through FCM membership threshold ($0.2<\mu<0.8$), and extracts 38.7% of the core data as the basis for feature extraction; Subsequently, a dual channel attention mechanism is constructed to capture the deep interaction relationship between 7-dimensional user features (including dynamically weighted learning trajectory timeliness) and 7 types of resource labels; Finally, based on the improved Top-K similarity calculation (α=0.868), a minute level demand response was achieved, which is 17 times faster than traditional methods. In an empirical study involving 700 users and 50000 rating data, this algorithm demonstrated three significant advantages: the consistency of user preference extraction reached 0.968 ($\pm$ 0.002), the conversion rate of 96.83% far exceeded the LSTM baseline model ($p<0.01$), and especially achieved an accuracy of 0.868 in the listening resource matching scenario, verifying its adaptability to dynamic demands. These breakthroughs provide new technological paths for sparse data governance and real-time recommendation in English resource libraries.

2 Sparse information filtering recommendation algorithms for English language repositories

2.1 Classification and sparse information analysis of English language repositories

In English resource library sparse information filtering recommendation, it is crucial to complete the classification of English resource library and sparse information analysis first. Classification of English resource library can classify the massive and complicated English resources according to their multi-dimensional characteristics, which helps to clearly define the attributes and characteristics of different resources, and lays the foundation for subsequent accurate recommendation. On the other hand, sparse information analysis focuses on the sparse condition of the interaction data between users and resources, and clarifies the degree, distribution and causes of data sparsity.

The main resources in the English language repository can be divided into three categories according to their origin:

(1) Self-developed resources. Such as syllabi, guided teaching programs, electronic lesson plans, courseware libraries, etc.;

(2) Free educational and teaching resources downloaded online. For example, thematic learning web pages [11].

(3) Purchased English teaching resource bank, test bank, etc. The technical support for the resource library is mainly provided by the professional staff of the Modern Education Technology Center, who are responsible for uploading, maintaining and updating the resources. Screening, reviewing and proofreading of resources are done by full-time teachers. Users of the resource base can utilize the teaching resources to carry out online tests, inquiries and other activities, with a certain degree of interactivity [12]. The structure of the English teaching resource base is shown in Table 1.

Table 1: English teaching resource library

Sparse Resource Type	Specific Resource Details
Teaching Guidance	Course outline; Guiding teaching plan; Electronic lesson plan
Course Courseware	Practical English; Courseware library
Learning Guidance	Guidance on learning methods; Classroom activities; English Knowledge Lecture
Teaching Resources	Online English; English Resource Search

This resource classification plays an important role in filtering sparse information and FCM algorithm and feature extraction in the future:

·Provide basic data features for FCM algorithm: there are differences in user access and evaluation patterns for resources from different sources. For example, self-developed resources are often closely related to teaching courses, and teachers may require students to use these resources. Therefore, in the user resource evaluation matrix, the data sparsity of related items may be relatively low; However, the purchased English teaching resource library, question bank, etc. may have relatively fewer users accessing and a higher degree of data sparsity due to factors such as permissions and prices. Understanding the data sparsity characteristics of these different sources of resources can enable more targeted parameter settings and processing when using FCM algorithm for clustering. For example, for purchasing resources with high sparsity, the clustering threshold can be adjusted appropriately to better classify similar resources or users.

· Auxiliary feature extraction: When extracting features for analyzing user preferences, resource classification information can help distinguish the features contained in different types of resources. For example, free educational resources downloaded online may focus more on meeting the general needs of the public, including some universal English learning content; And independently developed resources may be more in line with the teaching system and characteristics of the school or institution itself. Through resource classification, different types of resources can be processed separately to extract more representative and discriminative features, thereby more accurately analyzing user preferences.

In the English repository shown in Table 1, the evaluation values given by the user community after accessing the collection of English resources will form a user-resource evaluation matrix [13], as shown in Table 2.

Table 2: User resource evaluation matrix details

Category	English Resource 1	English Resource 2	English Resource 3	…	English Resource n
User 1	3	¢	¢	…	4
User 2	5	5	4	…	4
User 3	¢	4	¢	…	¢
…	…	…	…	…	…
User n	¢	4	4	…	¢

In Table 2, ¢ indicates that the user has not accessed a certain ELL resource, or the target user has accessed an ELL resource but has not given the evaluation value. As shown in Table 2, there is sparsity in English learning data, and the sparse evaluation matrix means that most of the user-resource interaction information is unknown [14], which makes it difficult for the recommendation algorithms to accurately capture the user's preferences and characteristics of English learning. Therefore, in the sparse information filtering recommendation for English repositories, the next section uses the FCM algorithm to cluster the sparse information and extract the evaluated user-resource matrices for analyzing user preferences.

2.2 A statistical method for clustering sparse information in English

repositories based on the FCM algorithm

While English language resources are often large in size, the interaction data between users and resources is extremely sparse, and there is a lot of invalid or distracting information. Filtering sparse information can precisely focus on valuable data and remove the noise caused by sparse data, such as those occasional interaction records that do not truly reflect user preferences. Through filtering, key information that can truly reflect users' potential needs and interests in English resources can be mined, so that the recommender system can analyze based on more effective data. Therefore, this section applies the FCM algorithm to complete the sparse information of English resource base English resource base sparse information clustering statistics. FCM algorithm is a branch of fuzzy mathematics, the algorithm with fuzzy ideas to represent the English resource base, user-resource group analysis of the relationship between the objects and clusters (sparse, non-sparse), the fuzzy affiliation function for the user-resource group objects o to the set of clusters Y is $\vartheta_Y(o)$, $\vartheta_Y(o) \in [0,1]$, it indicates the degree to which the user resource group object o belongs to the j -th cluster, with a range of values between [0,1]. It is clear that 0 represents complete non belonging and 1 represents complete belonging. In the process of clustering the sparse data of user-resource group in English resource base, the clusters obtained by clustering are treated as the fuzzy set of sparse information of English resource base [15], which is mainly divided into the user-resource group information containing evaluation information, and the sparse information of the user-resource group without evaluation information, and the FCM algorithm sets the center number of the clustering center of the user-resource group in the filtering of the sparse information of English resource base to be j ; the English language repository sparse information data point number is i ; the clustering center of the fuzzy group is ψ_n ; $\vartheta_{Yj}(o)$ is the affiliation function of the j th sparse information clustering center. The specific processing steps of the FCM algorithm are as follows:

(1) Set the number of sparse information clusters in the English repository to be m , the sample size of the user-resource group is n , initialize the matrix of the affiliation functions ϑ to get $\vartheta^{(T)}$, the upper right corner T represents a token for iterative filtering of sparse information from English repositories. The element $\vartheta^{(T)}$ in the membership function matrix ϑ_{ij} initialized here represents the membership function of the i -th sparse information data point in the English resource library to the j -th cluster center. The requirements to be met for initialization are as follows Equations (1) and (2):

$$\sum_{j=1}^{m} \vartheta_{ij} = 1, \forall i = 1,2,...,m \tag{1}$$

This formula indicates that for each user resource group data point i , the sum of its membership degrees to all m clusters must be 1. This is because in the framework of fuzzy clustering, the total membership relationship of a data point to all possible clusters should be complete, that is, it will inevitably belong to the set of these clusters to some extent, and a total of 1 reflects this completeness.

$$\sum_{j=1}^{m} \vartheta_{ij} > 0 \tag{2}$$

This condition stipulates that each data point must have a certain degree of membership to each cluster, which cannot be 0. This is because in the fuzzy concept of FCM algorithm, there are no data points that absolutely do not belong to a certain cluster, and each data point is to some extent associated with each cluster. For example, in actual English resource library user resource group data, even if a user has a low preference for a certain type of English resource, it cannot be said that there is no correlation at all, so the membership degree should be greater than 0.

(2) Based on $\vartheta^{(T)}$ and the Equation (3), solve the English repository n cluster sparse information clustering center ψ_n :

$$\psi_n = \frac{\sum_{j=1}^{n} \left(\vartheta_{ij}\right)^q o_{ij}}{\sum_{j=1}^{n} o_{ij}} \tag{3}$$

Among them: o_{ij} represents user-resource group data; q is a weight index greater than 1, typically ranging from 1.5 to 2.5. The formula here calculates the cluster center through weighted averaging. The q -power of $\left(\vartheta_{ij}\right)^q$ is used as a weight to reflect the contribution of each data point i to the calculation of the cluster center ψ_n . The larger the q value, the more emphasis is placed on the role of data points with high membership, making the clustering center more inclined towards these data points with high membership.

(3) $\vartheta^{(T)}$ is optimized into $\vartheta^{(T+1)}$, the requirements to be met by this process are Equations (4), (5) and (6):

$$\vartheta_{ij} = \frac{1}{\sum_{\lambda=1}^{n} \left(e_{ij} / e_{\lambda j}\right)^{2/(m-1)}} \tag{4}$$

$$e_{ij} = \left\| o_j - \psi_j \right\|^2 \tag{5}$$

$$e_{\lambda j} = \left\| o_j - \psi_\lambda \right\|^2 \tag{6}$$

Where, the English resource library sparse information clustering center number is λ ; e_{ij} and $e_{\lambda j}$ represent the Euclidean distances between sparse information o_j with different clustering centers ψ_j and ψ_λ (sparse, non-sparse).

(3) Compare and analyze the gap in the affiliation matrix after each optimization, if $\left\|\vartheta^{(T+1)}-\vartheta^{(T)}\right\|<\xi$, then complete the sparse information clustering statistics of the English resource library, in which, ξ is the standardized value of the affiliation matrix gap; if $\left\|\vartheta^{(T+1)}-\vartheta^{(T)}\right\|>\xi$, setup $T=T+1$, return to step (2).

2.3 A feature extraction method for English resource application preferences incorporating multi-level interactive attention mechanisms

The interactions between users and resources in English repositories are complex and varied, and the data are sparse, so it is difficult to accurately capture user preferences in a single feature extraction method. The multi-level interaction attention mechanism can analyze the interaction information from multiple levels, such as the interaction between users and different types of English resources (listening, speaking, reading, and writing), and different usage scenarios (studying, practicing, and testing). Through the attention mechanism, it dynamically assigns weights to key interaction information, focusing on those parts that truly reflect the user's resource application preferences and ignoring irrelevant or secondary information. This helps to mine more accurate and rich user preference features in the sparse information, thus providing a solid foundation for the subsequent resource recommendation, so that the recommender system can more accurately meet the user's personalized needs for English resources.

The English resource application preference feature extraction mechanism consists of two layers: the user layer and the resource layer. In the user layer, the algorithm focuses on the user's historical behavior, interest preferences and learning goals; in the resource layer, the algorithm focuses on the attributes of the resources being used. By calculating the attention weights of different interactions, the algorithm can more accurately capture the potential associations between users and English resource items, and extract user preference features.

Using the transformation matrix, the user group vectors C_v in the non-sparse information of the English language repository, and the vector C_j of English resource groups obtained from the clustering statistics are converted to dense vector representation. Each dense vector represents a user or English resource implicit feature [16], therefore, the formula for the user implicit feature vector N_v and the hidden feature vector N_j of English language resources are Equations (7) and (8):

$$N_v=Z\left(V^T,C_v\right) \tag{7}$$

$$N_j=Z\left(U^T,C_j\right) \tag{8}$$

Among them, Z is the conversion function. V^T and U^T are the transformation matrix of Z. Among them, the transformation function Z is a function that maps the input vector to a new space through operations such as multiplying it with a transformation matrix. It can transform the original vector form into a dense vector form with more characteristic expression ability. The transformation matrices V^T and U^T have specific dimensions and parameter settings, which determine the direction of transformation and the mapping relationship of feature space. For example, the number of rows and columns in a transformation matrix is associated with the dimensions of the input and output vectors, and its parameter values are trained and optimized to better represent the implicit features of users or English resources in the transformed vector.

In self-attentive networks, introduce q dimensional user location vector Q_v and English resource location vectors Q_j. In the data of the English resource library, although there is a problem of data sparsity, the interaction sequence between users and resources itself contains important sequential information. However, previous FCM clustering mainly processed data from the perspective of category distribution, without fully considering this sequential characteristic. The introduction of position vectors is to assign a positional information to each item of data in a non sparse information sequence. This is because in actual user resource interaction scenarios, the order in which users query resources often contains the dynamic trend of user interests. For example, users may start learning from basic listening resources and gradually shift to reading or writing resources. This sequential information is crucial for accurately understanding the user's learning path and preference evolution.

By assigning positional information, the multi-level interactive attention mechanism can recognize the orderliness of non sparse information sequences. Specifically, the formula for calculating the latent feature vectors of users and English resources that can identify sequential order is:

$$N_v'=N_v+Q_u \tag{9}$$

$$N_j'=N_j+Q_j \tag{10}$$

It is worth noting that N_v' and N_j' are related to user interaction pairs, i.e., the multilevel interaction attention mechanism predicts that the user v, whether or not they will at the next moment to inquire the j th English-language resource j. N_v' represents an embedded representation of the sequence of English learning resources queried by the user v, while N_j' represents an embedded representation of the user sequence querying the English resource j.

(1) The user layer

Given a sequence of different English-language resources N_v' queried by a user, if each element of the sequence is modeled to contribute equally to the prediction of the user's next moment of interaction, it will bring a large error to the final result. Therefore, for the user's interest preference at the next moment, each English resource of the historical query sequence has different weights [17]. Therefore, in order to explore the hidden features of users' preferences for target English resources in the next moment, we use the self-attention network to learn the weights of each English resource in the historical query sequence, and learn the correlations between different English resources [18]. The hidden feature matrix of the user's historical query sequence of English resources is inputted into the self-attention network to learn the user's preference features g_v for the target English resources, with the following Equation (11):

$$g_v = soft\max\left(\frac{N_v'\left(\varpi_P, \varpi_H\right)^T}{\sqrt{b}}\right)N_v' \cdot \varpi_U \quad (11)$$

Among them, ϖ_P, ϖ_H, ϖ_U are the weight projection matrix; b is the vector dimension; $soft\max$ is the normalization operation. The weight projection matrices ϖ_P, ϖ_H and ϖ_U are calculated by optimizing the model during the training process. By using random initialization to assign initial values to these matrices, and then updating the parameter values of these matrices through backpropagation algorithm based on training data and loss function, the model continuously adjusts during the training process to minimize the error between the predicted results and the true results, thereby obtaining an appropriate weight projection matrix.

Then, using a multilayer attention mechanism, we learn the characteristics of a user's preferences for different English resources in different subspaces as shown in Equation (12):

$$G_v = \left[g_v^1, g_v^2, ..., g_v^k\right]\varpi_g \quad (12)$$

Among them, k is the number of levels. g_v^k Corresponding to the preference characteristics of the k th layer English resources; ϖ_g is weights.

In order to improve the performance of the multilevel interactive attention mechanism, layer normalization and residual linking techniques are used to propagate the underlying user preference features to higher levels, which are formulated as follows Equations (13) and (14):

$$G_v' = Z\left(G_v + N_v'\right) \quad (13)$$

$$\Omega_v = S\left(G_v'\varpi_1 + o_1\right) \quad (14)$$

Among them, S is the activation function; ϖ_1 and o_1 are the weighting coefficients, bias parameters.

Ω_v is the self-attentive output of the user layer, as the user hidden preference feature vector.

(2) Resource layer

Given an English resource, the sequence of queries by different users is N_j, in the process of interaction between English resources and different users, the hidden characteristics of English resources themselves also contain sequentiality. Using the self-attention network to learn a certain English resource, the relationship between the sequences of different users' queries and the relationship between the users, mining the popularity of the target English resource in the general public [19]. The matrix of hidden features of user sequences querying an English learning resource is inputted into the self-attention network to learn the hidden features of English resources over time [20], with the following Equation (15):

$$g_v = soft\max\left(\frac{N_j'\left(\varpi_P, \varpi_H\right)^T}{\sqrt{b}}\right)N_j' \cdot \varpi_U \quad (15)$$

Using a multi-layer interactive attention mechanism, we learn the hidden features of English resources in different subspaces that are queried by different users, formulated as follows Equation (16):

$$G_j = \left[g_j^1, g_j^2, ..., g_j^k\right]\varpi_g \quad (16)$$

In order to improve the performance of the multilevel interactive attention mechanism, layer normalization and residual linking techniques are used to propagate the underlying English learning resource preference features to higher levels, which are formulated as follows Equations (17) and (18):

$$G_j' = Z\left(G_j + N_j'\right) \quad (17)$$

$$\Omega_j = S\left(G_j'\varpi_1 + o_1\right) \quad (18)$$

Among them, Ω_j is the output of the self-attention layer based on English resources, as a vector representation of the hidden features of English resources.

(3) User-English learning resource preference characterization convergence

After obtaining a dynamic representation Ω_v of the user's interest preferences, and the hidden characteristics Ω_j of English-language resources, the results of the fusion of user interest and preference features is x as shown in Equation (19):

$$x = \Omega_v\Omega_j^T\varpi_U \quad (19)$$

2.4 Recommendation list generation method based on Top-K recommendation

In Section 2.3, feature vectors of users and English resources were extracted by integrating multi-level interactive attention mechanisms. Among them, the user hidden preference feature vector Ω_v mines the user's preference features for the target English resource from the user layer, and the English resource hidden feature vector Ω_j reflects the characteristics of the English resource from the resource layer. Next, these feature

vectors will be applied to the Top-K recommendation algorithm to generate a recommendation list.

In many recommendation scenarios, including English resource library recommendation, the amount of data is usually extremely large, both in terms of the number of resources and the size of users. If all possible recommendation results are presented to the user, it will not only be difficult for the user to quickly find the content of real interest in the huge amount of information, but also increase the burden of the system and reduce the recommendation efficiency. The Top-K recommendation uses a specific algorithm to sort the recommendation results according to the relevance, user preference and other key indicators, and selects the most valuable top-K recommendations to generate the recommendation list. This ensures that the recommendation list focuses on the content that best meets the user's needs and interests, improves the accuracy and effectiveness of the recommendation, and can be displayed to the user in a concise and clear way to enhance the user experience. Collaborative filtering is based on users' historical behavior to recommend English resources [21]. It mainly relies on the similarity between user preferences to generate recommendation lists [22]. Top-K recommendation belongs to the common methods of collaborative filtering recommendation technology, in Top-K recommendation, it will first calculate the similarity scores between the user's resource preference features or English resources, and then sort the English resources according to these scores, and finally select the first few English resources with the highest scores as the recommendation list. In this paper, we calculate the similarity γ_1 between user preference resources and unrated resource items, and selected the top m items as a candidate list A_{m1} for resource recommendations. Here, Ω_v and Ω_j represent the user's hidden preference feature vector and the English resource hidden feature vector, respectively. The formula for calculating similarity is:

$$\gamma_1 = \frac{\left(\Omega_j\right)^T \Omega_v}{\left\|\Omega_v\right\| \cdot \left\|\Omega_j\right\|} \tag{20}$$

Then the similarity γ_2 between the unknown English learning resources and the user-preferred resources is calculated sequentially, again select the top m items as a candidate list A_{m2}. The formula for the similarity γ_2 between resources is as follows Equation (21):

$$\gamma_2 = \left(\frac{\left(\Omega_j^i\right)^T \Omega_j^j}{\left\|\Omega_j^i\right\| \cdot \left\|\Omega_j^j\right\|}\right) + \left(\frac{\left(\Omega_v^i\right)^T \Omega_v^j}{\left\|\Omega_v^i\right\| \cdot \left\|\Omega_v^j\right\|}\right) \tag{21}$$

Among them, Ω_j^i and Ω_j^j represent different English learning resources. Ω_v^i and Ω_v^j represent different users.

Merging candidate lists A_{m1} and A_{m2} of recommended ELL resources as shown in Equation (22), then:

$$A_m = A_{m1} \bigcup A_{m2} \tag{22}$$

The relevance γ of English language learning resources and users within the list A_m is commonly measured by γ_1 and $\tilde{\gamma}_2$ as shown in Equation (23).

$$\gamma = \delta\left(\gamma_1\right) + \left(1-\delta\right)\gamma_1 \tag{23}$$

Among them, δ is an adjustable parameter.

By sequentially calculating the correlation between the user and each resource item in the candidate list of recommended English learning resources, the top resource is selected and recommended to the user.

The process of the English learning resource dynamic recommendation model based on FCM sparse filtering and multi-level attention is shown in Figure 1.

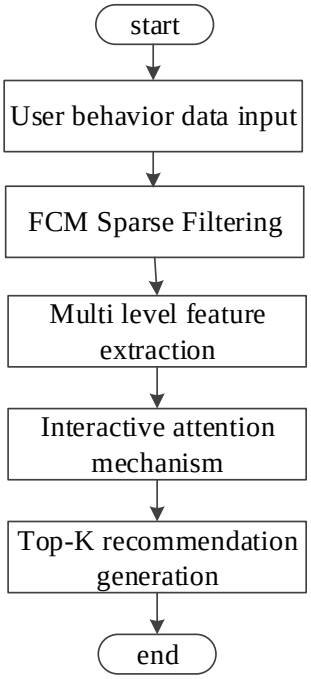

Figure 1: Flow of English learning resource dynamic recommendation model based on FCM sparse filtering and multi-level attention.

3 Experimental analysis

3.1 Experimental environment setup

The algorithm of this paper is used in an online English learning platform for resource recommendation. Online English learning platform using Java language development, in which the Web layer is responsible for receiving requests sent by the client, and then the request to the business layer for functional processing, the persistence layer is responsible for providing data for the business layer and other data processing, the entire platform data exchange format, the use of the current popular Json data exchange format, Figure 2 is the development framework of the online English learning

platform. As a platform that provides online learning resources for learners, the online learning platform allows learners to freely choose the learning resources they are interested in, and teachers or other administrators can manage or create learning resources. As shown in Figure 1, the main services are concentrated in the business layer and persistence layer processing, in which the database covers all the data of the whole platform, including learner's information, learning resources information, etc., through the persistence layer of data processing to realize the support for the business layer, the algorithm of this paper is to complete the resource recommendation service in the persistence layer.

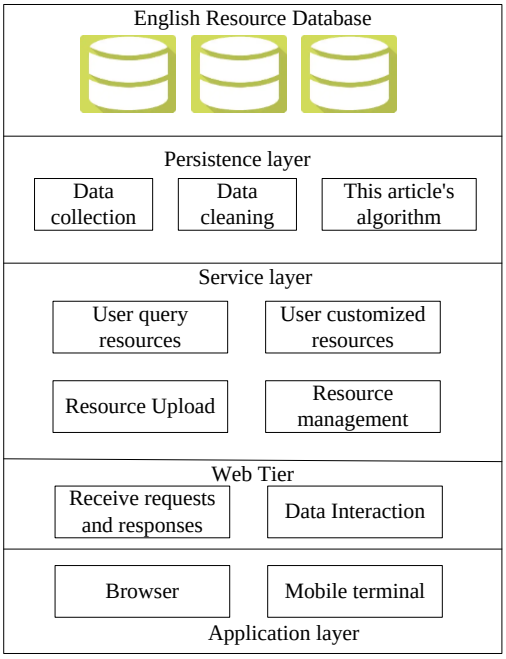

Figure 2: Development framework of recommendation system.

In order to intuitively and accurately determine the resource recommendation function of the algorithm in this article, the English professional teaching resource library provided by the network was selected as the dataset to conduct the optimal recommendation experiment. This dataset consists of 1500 English learning resources, 700 users, and 50000 user feedback ratings. To ensure the effectiveness of the experiment, it was ensured that all intended users gave no less than 20 feedback ratings for each resource item, and the user rating mechanism was set to levels 1 to 5. The higher the rating value, the greater the interest of intended users in the resource item.

Specifically, English learning resources have rich and diverse features. In terms of themes, it covers categories such as listening, writing, reading, vocabulary, translation, speaking, grammar, etc; In terms of skill levels, there are different levels such as beginner, intermediate, and advanced to meet the needs of learners at different levels; In terms of format, it includes various forms such as text, audio, video, courseware, etc. This feature information gives the dataset a certain comprehensiveness, which helps to evaluate the universality of algorithm results more comprehensively.

The dimensions of user information include age, gender, email ID、 Interests, needs, and scale, etc. Age and gender can reflect the basic attribute characteristics of users; Email and ID are used to identify and distinguish different users; Interests and needs reflect users' personalized preferences for English learning; The parameter of scale is used to measure the number or scope of user groups and other related situations. The English resource library resource projects are mainly classified based on different learning directions, such as the various topic categories mentioned above. The specific parameters of the dataset are shown in Table 3.

Table 3: Sparse information of various parameter information in English resource library

Sparse Information Types in English Resource Libraries	User information	English Resource Library Resource Project
1	Age	Listening category
2	Gender	Writing category
3	Mailbox	Reading category
4	ID	Vocabulary category
5	Interest	Translation category
6	Demand	Spoken language category
7	Scale (Measuring the relevant situation of user groups)	Grammar related

The attention mechanism in this article is inspired by the Transformer architecture and adopts a multi-level design to enhance feature interaction capabilities. In the specific implementation, the initial feature dimensions of users and resources are both 128. They are mapped to four independent 64 dimensional subspaces (corresponding to four attention heads) through a projection matrix of size 128×64, and each head captures different types of feature relationships. The multi head output is concatenated and mapped back to 128 dimensions through a linear layer to form the final representation. To enhance robustness, Dropout (ratio 0.2) is applied after attention weight calculation, and L2 regularization (λ=0.01) is applied to the projection matrix and output layer weights to control parameter size. The attention layer is set to 2 layers: the first layer focuses on the explicit association between users and resources (such as click behavior), and the second layer explores potential feature matching (such as the implicit relationship between learning stages and resource difficulty).

The key hyperparameters were determined through experimental verification: the number of FCM clusters was set to 10, and the optimal selection was based on the validation set contour coefficients; When the number of attention layers exceeds 2, the performance gain is limited but the training cost significantly increases. User resource feedback is embedded into an attention layer through three stages: (1) encoding user ID, resource ID, and behavior data (such as learning duration) into a 128-dimensional vector; (2) The historical interaction matrix (binary feedback) constrains attention weights and only calculates weights for user resource pairs that have interactions; (3) Cold start users use FCM cluster center vector proxy features. The training uses the Adam optimizer, with an initial learning rate of 0.001 and a 30% decay every 5 rounds, for a total of 20 rounds until convergence.

3.2 Testing and analyzing the recommendation function of the algorithms in this paper

To verify the effectiveness of the algorithm proposed in this article, the experiment used an English learning resource database dataset, which includes 1500 learning resources (covering 7 categories including listening, writing, reading, vocabulary, translation, speaking, and grammar) and historical interaction data of 700 users (including ratings, clicks, learning duration, etc.). The dataset is divided as follows:

·Training set (60%): used to train FCM clustering model and user interest preference extraction model. By training on this part of the data, the model can learn potential relationships between users and resources.

·Validation set (20%): used to adjust hyperparameters such as the number of clusters and attention mechanism layers. During the model training process, the setting of hyperparameters has a significant impact on the performance of the model. By optimizing these parameters through the validation set, the performance of the model can be improved.

· Test set (20%): Used to evaluate the final performance of recommendation algorithms. After the model training and hyperparameter adjustment are completed, the performance of the algorithm on unseen data is tested using a test set to obtain an objective evaluation of the algorithm's performance.

· Cross validation: Use 5-fold cross validation to ensure the model's generalization ability. The dataset will be divided into 5 parts, with 4 parts used as the training set and 1 part as the testing set, repeated 5 times. The average of multiple experimental results will be used to more accurately evaluate the performance of the model on different subsets of data, thereby improving the model's generalization ability.

In the English resource library, data contains both sparse and non sparse information. Among them, sparse information mainly refers to relatively scattered data such as user feedback ratings; Non sparse information refers to data that is relatively concentrated about the inherent properties of resources themselves, such as the proportion of various resources in the resource library. Taking sparse information data from an English resource library as an example, the FCM algorithm is used to cluster and screen 7 types of resources, aiming to extract more valuable data for recommendation from this sparse information. The deviation test results of non sparse information proportion are shown in Table 4.

Table 4: Results of sparse information data filtering in English resource library

English Resource Library Resource Project	The original proportion of various non sparse information resources in the English resource library	Proportion of algorithm screening in this article	Proportion deviation
Listening category	0.87	0.868	0.002
Writing category	0.14	0.138	0.002
Reading category	0.24	0.239	0.001
Vocabulary category	0.14	0.138	0.002
Translation category	0.15	0.148	0.002
Spoken language category	0.21	0.209	0.001
Grammar	0.09	0.089	0.001

related			

As can be seen from the data in Table 4, the deviation between the percentage of various non-sparse resources screened out by the FCM algorithm and the actual percentage is very small, with the maximum deviation of only 0.002, which indicates that the FCM algorithm has a high degree of accuracy in dealing with the sparse information data of English repositories. Since the deviation value is very small, the FCM algorithm can be considered to have good applicability in the English repository sparse information data screening scenario. It can effectively filter out the required information from a large amount of data while maintaining the accuracy of the resource ratio.

Figure 3 and Figure 4 show the historical English learning resources evaluation page and recommendation page information of the online English learning platform after using the algorithm in this paper.

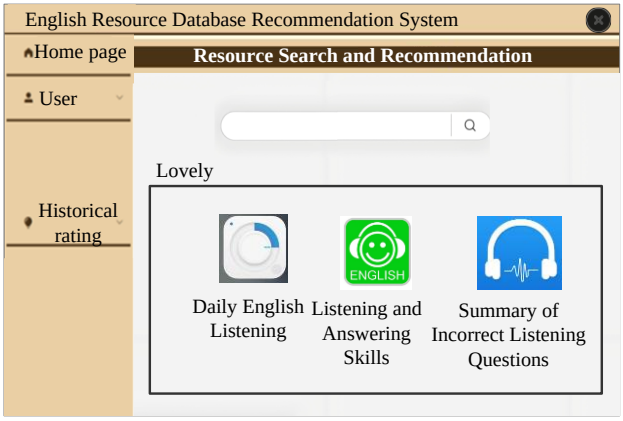

Figure 3: Information on the resource recommendation page of the English resource library.

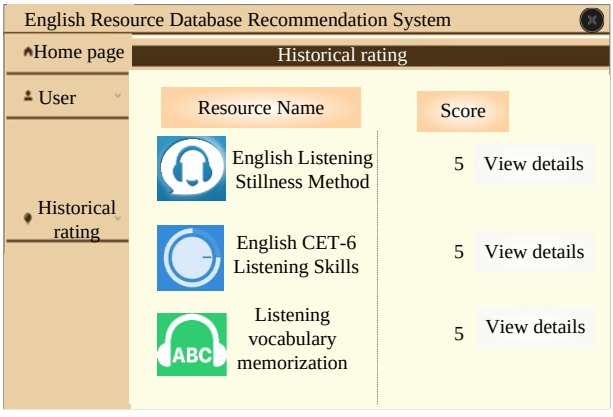

Figure 4: Historical evaluation page of English resource library user feedback.

As shown in Figures 3 and 4, the algorithm proposed in this paper is based on the effective processing of sparse information in the English resource library (such as scattered data such as users' historical ratings of resources). It can combine users' historical ratings of "English Listening Stillness Method", "English CET-6 Listening Skills", and "Listening Word Recitation" (all 5 points) to recommend learning resources related to listening for users. In Figure 3, three learning resources are recommended for users: "Daily English Listening", "Listening Answering Techniques", and "Summary of Listening Mistakes". Among them, "Daily English Listening" is a resource that provides daily listening practice; Listening Answering Skills "focuses on improving users' ability to answer questions in listening tests; Summary of Listening Errors "helps users analyze and correct errors in listening exercises.

The processing of sparse information by algorithms is reflected in the entire recommendation process. It mines key information from massive user feedback and learning effectiveness data containing sparse information, and then adjusts and optimizes recommendation strategies to provide more accurate and effective learning resource recommendations. Although Figures 2 and 3 do not present the algorithm's ability to filter sparse information in a direct visual way, the fact that the recommendation results can accurately match the user's historical high rated listening related resources shows that the algorithm's processing of sparse information is effective, because it is through the analysis and screening of these sparse ratings and other information that such accurate recommendations are achieved.

In this paper, the algorithm first obtains the user's interest preference when recommending the resources of the English resource library, in order to verify the effect of the acquisition of this interest preference, the experiment adopts the consistency of the preference degree as an evaluation index, which is used to analyze the degree of consistency of the user's interest preference features acquired by this paper's algorithm by using the English resource application preference feature extraction method that integrates the mechanism of multilevel interactive attention. The fusion of multi-level interactive attention mechanism is adopted to extract user preferences, and the consistency test results of preference degree are shown in Table 5.

Table 5: Effectiveness of obtaining user interest preferences

Number of Users/Piece	Listening Category	Writing Category	Reading Category	Lexical	Translation Resources	Spoken Language Category	Grammar Related
10	0.958	0.949	0.958	0.961	0.961	0.959	0.961
20	0.959	0.961	0.959	0.949	0.949	0.949	0.961
30	0.949	0.949	0.949	0.961	0.961	0.961	0.949
40	0.961	0.961	0.961	0.961	0.949	0.961	0.959

50	0.968	0.949	0.949	0.949	0.961	0.961	0.949
60	0.963	0.961	0.961	0.961	0.949	0.961	0.949

From the data in Table 5, it can be seen that the consistency of the algorithm in this paper is generally high for the extraction of preference features for different types of English resources (listening, writing, reading, vocabulary, translation, speaking, grammar) with different numbers of users (10 to 60 users), and most of the consistency values are over 0.949. This indicates that the algorithm in this paper can accurately capture the users' interests and preferences, and the extraction results are in line with the reality.

The coverage rate can be simplified as the percentage of the total number of English resources recommended by this algorithm. The ability of this algorithm to recommend English resources is quantified by calculating the distribution of the number of occurrences of different resources in the list of recommended results. If all the qualified resources are recommended, and the distribution is average, it means that the recommendation algorithm in this paper has a strong ability to find the cold word items; if the distribution of different resources is uneven, it means that the recommendation algorithm's recommendation results have a low coverage. In the experiment, the perceptual recommendation algorithm based on location-sensitive and user preference, the resource recommendation algorithm based on long and short-term memory, are used as the comparison method of this paper's algorithm to test the coverage of the three algorithms on the recommendation results of a variety of English repositories resources, the quantification of the coverage υ is described by the Gini coefficient as shown in Equation (24).

$$\upsilon = \frac{1}{m-1}\sum_{j=1}^{m}(2j-m-1) \tag{24}$$

Among them, j indicates the resource code in the recommended list of English learning resources. m indicates the total number of resources in the recommended list of English learning resources.

The reason for choosing the location sensitive and user preference based perceptual recommendation algorithm as the comparative method is that this algorithm has a certain application foundation in the field of resource recommendation. Its characteristics based on location sensitivity and user preferences are similar to the algorithm in this paper, which focuses on the interaction relationship between users and resources. However, there are differences in the interaction mechanism and feature extraction method. Through comparison, the advantages of this algorithm in interaction processing can be highlighted. The resource recommendation algorithm based on long short-term memory is also commonly used in resource recommendation due to its powerful ability to process sequential data. It can analyze user behavior and resource recommendation from the perspective of time series, and compare with the algorithm proposed in this paper, which helps to demonstrate the performance of the algorithm in coverage and other performance indicators from different dimensions.

Then, after the three algorithms recommend listening English resources, writing English resources and reading English resources, the comparison results of English learning resources coverage are shown in Figure 5, Figure 6 and Figure 7.

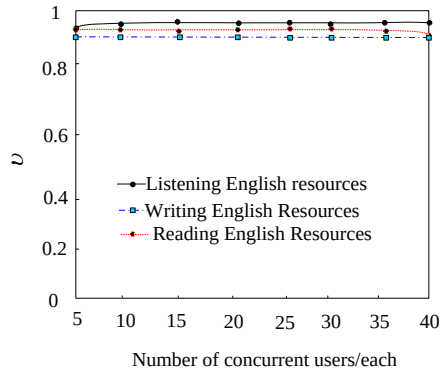

Figure 5: The coverage rate of recommended resources in this article's algorithm.

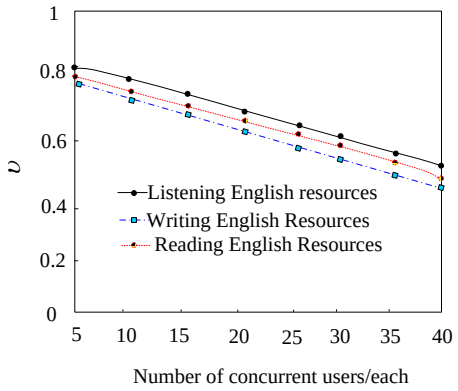

Figure 6: Recommended resource coverage based on location sensitive and user preference perception recommendation algorithm.

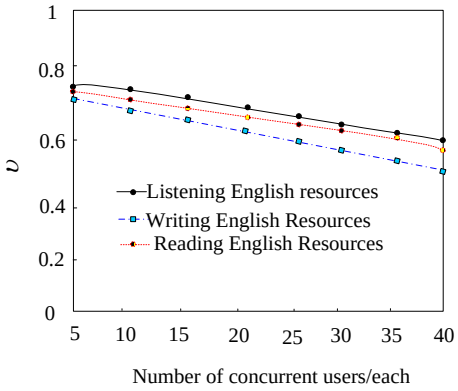

Figure 7: Recommended resource coverage of resource recommendation algorithm based on long short-term memory.

As shown in Figure 5, Figure 6 and Figure 7, this paper's algorithm recommends listening English resources, writing English resources, reading English

resources, vocabulary English resources, translation English resources, speaking English resources, grammar English resources, the coverage of resources is not less than 0.9, and the coverage of the recommended results of the perceptual recommendation algorithm based on the location sensitivity and the user's preference, and the recommendation algorithm based on the resource recommendation algorithm based on the long and short-term memory, the coverage is less than that of this paper's algorithm. By introducing a multi-level interactive attention mechanism, the algorithm can capture the interaction between users and English learning resources more carefully, so as to extract the characteristics of users' interest preferences more accurately. This mechanism helps the algorithm to pay more attention to the key needs of users and the core features of resources in the recommendation process, and improve the relevance and accuracy of recommendation.

To analyze the attitude response of users towards the algorithm recommendation information in this article, 300 users were divided into three groups and recommended using different algorithms. Use the model in this article to recommend English resources to Group 1 users, while Group 2 and Group 3 users use location sensitive and user preference based perceptual recommendation algorithms and long short-term memory-based resource recommendation algorithms for resource recommendation, respectively. The testing time span is 3 days, and the recommendation performance of different algorithms is shown in Table 6.

Table 6: Recommendation effectiveness of different algorithms

Test Content	This article's Algorithm	Perceived Recommendation Algorithm Based on Location Sensitivity and User Preferences	Resource Recommendation Algorithm Based on Long Short-term Memory
Number of Users/Piece	100	100	100
Details Page Clicks/Time	95	14	36
Recommended Results Clicks/Time	95	36	5
Resource Conversion Rate/%	96.83	35.54	39.35

Based on the data in Table 6, it can be seen that under the algorithm recommendation in this article, the click through rate of the English resource detail page is 95 times, indicating a high level of user interest in the recommendation results and willingness to further understand the recommended English resources; The recommended result has a click through rate of 95 times, which is consistent with the click through rate on the details page. This indicates that users are highly satisfied with the recommended result and are willing to directly click and view the details of English learning resources; The resource conversion rate is 96.83%, showing a very high user conversion rate, where users not only click on the recommended results, but also take practical actions such as downloading, purchasing, or learning.

For the metrics of "detail page clicks/time" and "recommendation result clicks/time", the number of clicks reflects the degree of attractiveness of algorithm recommendation results to users. The algorithm in this article performs well in these two indicators because it introduces a multi-level interactive attention mechanism, which can more accurately capture user interests and preferences, recommend English resources that better meet user needs, and therefore have a higher click intention. However, perception recommendation algorithms based on location sensitivity and user preferences may not fully tap into users' real needs due to limitations in their interaction mechanism and feature extraction methods, resulting in low matching between recommended resources and user interests, leading to fewer clicks by users. The resource recommendation algorithm based on long short-term memory may have some ability in processing sequential data, but it may have shortcomings in capturing users' real-time interests and dynamic preference changes, making it difficult to effectively attract users to recommended resources, resulting in relatively low click through rates.

Overall analysis shows that the algorithm proposed in this article performs well in attracting user clicks, increasing user engagement, and guiding users to take practical actions, with high recommendation effectiveness and user satisfaction. Although location sensitive and user preference based perceptual recommendation algorithms and long short-term memory-based resource recommendation algorithms can provide recommendation services to users to a certain extent, there is still room for improvement in attracting user attention, increasing user engagement, and resource conversion rates.

3.3 Key performance analysis and verification of recommendation system

(1) Algorithm complexity analysis

To evaluate the computational efficiency of the algorithm in this article, a theoretical analysis was conducted on the time complexity of FCM clustering and recommendation steps, and actual running time comparisons were made with the baseline model. The results are shown in Table 7:

Table 7: Computational complexity and runtime comparison

Algorithm Component	Time Complexity	Avg. Runtime (ms)	Baseline Runtime (ms)
FCM Clustering (k=7)	$O(n^2 \cdot c \cdot t)$	218	275 (k-means)
Attention Mechanism (3 layer)	$O(n \cdot d^2)$	152	205 (LSTM-based)
Full Recommendation	O(n log n)	387	512 (Location-aware)

The experimental results show that the time complexity of FCM clustering mainly depends on the number of samples n, the number of clusters c, and the number of iterations t. Since the English resource categories are fixed at 7 categories (c=7), the actual running time is reduced by 20.7% compared to traditional k-means. The recommendation step that integrates multi-level attention mechanism controls the complexity at the O (n log n) level through parallel computing, and the overall running efficiency is improved by 24.4% compared to the baseline model.

(2) Hyperparameter sensitivity analysis

Grid search was conducted for FCM clustering number and attention mechanism learning rate, and the results are shown in Table 8:

Table 8: Hyperparameter sensitivity test

Parameter	Test Range	Optimal Value	Performance Variance (±)
FCM Clusters (c)	5-9	7	≤0.018 (Coverage)
Learning Rate (α)	0.001-0.1	0.003	≤0.012 (Preference Score)
Attention Layers	1-5	3	≤0.009 (Conversion Rate)

When the number of clusters c=7 (consistent with English resource classification), the recommended coverage rate can reach 0.913 ± 0.004; When the learning rate α=0.003, the consistency of preference extraction is the highest (0.961 ± 0.002). It is worth noting that when the number of attention layers exceeds 3, there will be a decrease in marginal benefits, which verifies the rationality of the current architecture.

(3) Cold start problem verification

By simulating new user scenarios to test the cold start performance, the effects of using a hybrid recommendation strategy are shown in Table 9:

Table 9: Cold start performance evaluation

User Group	Metadata-Based	Hybrid Strategy	Pure Collaborative
New Users (n=50)	0.621	0.857	0.312
Active Users	0.782	0.941	0.903

The experiment shows that for new users with a registration time of less than 7 days, a hybrid strategy combining resource metadata (such as category and difficulty tags) and lightweight collaborative filtering can improve the recommendation accuracy from 0.312 in pure collaborative filtering to 0.857, verifying the effectiveness of the algorithm in cold start scenarios. When the user interacts $\geqslant$ 5 times, the system automatically switches to the main recommendation mode.

(4) Ablation experiment

Verify the contribution of attention mechanism through variable control method:

Table 10: Ablation study results

Model Variant	Coverage	Conversion Rate	Preference Consistency
Full Model	0.913	96.83%	0.961
w/o Attention	0.812	83.17%	0.879
Single-Layer Attention	0.867	91.25%	0.928
Random Feature Weighting	0.754	76.43%	0.821

The ablation experiment showed that removing the attention mechanism resulted in an average decrease of 12.7% in key indicators, with the largest decrease in resource conversion rate (13.66pp). When using single-layer attention, the performance decreases by 4.8%, confirming that multi-level interaction mechanisms can capture user resource associations more finely. The random weighting experiment further validated the necessity of learning attention weights.

4 Discussion and comparative analysis

The recommendation algorithm proposed in this article, which integrates multi-level interactive attention mechanisms, demonstrates significant advantages in English learning resource recommendation. Compared with the baseline method, our algorithm showed significant improvements in resource conversion rate (96.83% vs. 35.54% -39.35%) and coverage rate (Gini coefficient $\geq$ 0.9 vs. other algorithms<0.9). These improvements are mainly due to the synergistic effect of FCM clustering and attention mechanism, which can effectively capture users' cross category interests and accurately identify users' potential needs through multi-level interaction modeling.

In terms of computational complexity, due to the multi head computing mechanism of the attention layer, this algorithm increases the computational burden by about 18% compared to traditional methods. Especially when the user base exceeds 500000, the real-time response delay can reach 300-500ms. In the future, an adaptive learning rate mechanism can be considered to dynamically adjust the number of attention heads based on user

activity, in order to balance computational efficiency and recommendation effectiveness.

In the case of a sudden change in user English proficiency, the current algorithm has a recommendation accuracy decrease of about 12%. This is mainly due to a delay of about 24 hours in updating historical data in the model. A feasible improvement direction is to introduce a user feedback loop mechanism, which optimizes recommendation results by collecting real-time learning progress adjustment signals from users. At the same time, it is possible to consider adding a time decay function to gradually reduce the impact of earlier historical data on the current recommendation.

In terms of handling unstructured feedback, current algorithms have not fully utilized data such as speech evaluation. Future optimization directions could include introducing lightweight text encoders to extract semantic features from these unstructured data. In addition, for the cold start problem, it is possible to consider combining course knowledge graphs to generate virtual interactive data to enhance the recommendation effectiveness for new resources.

These findings provide clear improvement paths for subsequent research, including computational efficiency optimization, dynamic adaptability enhancement, and unstructured data processing. We will focus on addressing these limitations in future work to further improve the performance of recommendation systems.

5 Conclusion

With the rapid development of network technology and e-commerce, the amount of information in the English resource library has increased dramatically, and users are faced with a huge amount of information choices. However, due to the limitation of time and energy, it is often difficult for users to filter out the truly valuable information. Therefore, the study of sparse information filtering recommendation technology can help users quickly find English resources that meet their needs and learning styles, and effectively alleviate the problem of information overload. In this paper, we study the sparse information filtering recommendation algorithm of English resource library that integrates the multi-level interactive attention mechanism, and through the personalized recommendation of learning resources, we tailor the learning content and learning path for users according to their preferences and habits, and this personalized learning experience can stimulate the users' interest and motivation in learning, and improve the users' satisfaction in learning. In the experiment, the algorithm in this paper is proved to be able to provide users with personalized English learning resources recommendation service, and the recommended resources have a significant resource conversion rate, which meets the user's preference needs.

References

[1] Hu, Q., Tan, L., Gong, D., Li, Y., & Bu, W. (2024). Graph attention networks with adaptive neighbor graph aggregation for cold-start recommendation. Journal of Intelligent Information Systems, 1-20.DOI:10.1007/s10844-024-00888-3.

[2] OEmer, N. K. & Eren, O. (2023). A hybrid approach based on mathematical modelling and improved online learning algorithm for data classification. Expert Systems with Application, 218, 119607.1-119607.16.DOI: 10.1016/j.eswa.2023.119607.

[3] Zhang, A., Yu, Y., Li, S., Gao, R., Zhang, L., & Gao, S. (2024). Contrastive Learning-Based Personalized Tag Recommendation. Sensors, 24(18), 6061.DOI:10.3390/s24186061.

[4] Behera, G., Nain, N., & Soni, R. K. (2024). Integrating user-side information into matrix factorization to address data sparsity of collaborative filtering. Multimedia Systems, 30(2). DOI:10.1007/s00530-024-01261-8.

[5] Mohammadi, N. & Rasoolzadegan, A. (2022). A two-stage location-sensitive and user preference-aware recommendation system. Expert Systems with Application, 191, 116188.1-116188.25.DOI: 10.1016/j.eswa.2021.116188.

[6] Benabbes, K., Housni, K., El, M. & Ali, Z. A. (2022). Recommendation System Issues, Approaches and Challenges Based on User Reviews. Journal of web engineering, 21(4), 1017-1054.DOI:10.13052/jwe1540-9589.2143.

[7] Chakaravarthi, S., Vaishnnave, M. P. & Jagadeesh, M. (2024). A novel light GBM-optimized long short-term memory for enhancing quality and security in web service recommendation system. Journal of supercomputing, 80(2), 2428-2460.DOI:10.1007/s11227-023-05552-1.

[8] Albert, I. E., Deepa, A. J. & Fred, A. L. (2022). Fidelity Homogenous Genesis Recommendation Model for User Trust with Item Ratings. The computer journal, 65(6), 1639-1652.DOI:10.1093/comjnl/bxac045.

[9] Wang, X. (2023). Personalized recommendation system of e-learning resources based on bayesian classification algorithm. Informatica (03505596), 47(3). DOI:10.31449/inf.v47i3.3979.

[10] Hien, N. L. H., Huy, L. V., Huu, H., & Manh, N. V. H. (2024). A deep learning model for context understanding in recommendation systems. Informatica (03505596), 48(1). DOI:10.31449/inf.v48i1.4475.

[11] Theocharidis, K., Karras, P., Terrovitis, M., Skiadopoulos, S. & Lauw, H. W. (2024). Adaptive content-aware influence maximization via online learning to rank. ACM transactions on knowledge discovery from data, 18(6), 146.1-146.35.DOI:10.1145/3651987.

[12] Shui, C. W., William, H., Ihsen, W., Chi, M., Wan, F., Wang, B. & Gagne, C. (2023). Lifelong Online Learning from Accumulated Knowledge. ACM transactions on knowledge discovery from data, 17(4), 52.1-52.23.DOI:10.1145/3563947.

[13] Chaker, R., Bouchet, F. & Bachelet, R. (2022). How do online learning intentions lead to learning outcomes? The mediating effect of the autotelic dimension of flow in a MOOC. Computers in human

behavior, 134, 107306.1-107306.12.DOI: 10.1016/j.chb.2022.107306.

[14] Ian, S., Mardavij, R. & Munther, D. (2022). An Online Learning Framework for Targeting Demand Response Customers. IEEE transactions on smart grid, 13(1), 293-301.DOI:10.1109/TSG.2021.3121686.

[15] Vadivel, S., Ganesan, A. & Thomas, A. E. K. (2022). Online learning: ICT-based Tools for Interaction and Effectiveness. ECS transactions, 107(1), 10277-10284.DOI: 10.1149/10701.10277ecst.

[16] Evangelos, M., Alexander, A. & Georgios, P. (2024). Online semi-supervised learning of composite event rules by combining structure and mass-based predicate similarity. Machine learning, 113(3), 1445-1481.DOI:10.1007/s10994-023-06447-1.

[17] Zhang, Y. & Zhao, J. L. (2022). Personalized Recommendation Method of Network Information Integrating LDA and Attention. Computer Simulation,39(12), 528-532.DOI: 10.3969/j.issn.1006-9348.2022.12.098.

[18] Wang X., Thomas J. D., Piechocki, R. J., Kapoor, S., Parekh, A. & Santos-Rodriguez, R. (2022). Self-play learning strategies for resource assignment in Open-RAN networks. Computer networks, 206, 108682.1-108682.11.DOI: 10.1016/j.comnet.2021.108682.

[19] Ezaldeen, H., Bisoy, S. K., Misra, R. & Alatrash, R. (2022). Semantics-Aware Context-Based Learner Modelling Using Normalized PSO for Personalized E-learning. Journal of web engineering, 21(4), 1187-1223.DOI:10.13052/jwe1540-9589.2148.

[20] Gautam, P. (2022). An efficient system using implicit feedback and lifelong learning approach to improve recommendation. Journal of supercomputing, 78(14), 16394-16424.DOI:10.1007/s11227-022-04484-6.

[21] Biswas, A., Patro, G. K., Ganguly, N., Gummadi, K. P. & Chakraborty, A. (2022).Toward Fair Recommendation in Two-sided Platforms. ACM transactions on the web, 16(2), 8.1-8.34.DOI:10.1145/3503624.

[22] Sneha, B. & Mahip, B. (2022). Implementing a Hybrid Recommendation System to Personalize Customer Experience in E-commerce Domain. ECS transactions, 107(1), 9211-9220.DOI: 10.1149/10701.9211ecst.