
Application and Optimization of Semantic-enriched Keyword Prefetching Driven by Intelligent Technology in Language Education Network Platforms

Zhubin Luo* and Feifei Guo

College of Liberal Arts, Hunan University of Humanities, Science and Technology, Loudi, Hunan 417000, China

E-mail: luozhubin309@huhst.edu.cn

**Corresponding Author*

Received 29 April 2024; Accepted 14 April 2025

Abstract

With the rapid development of intelligent technology, the application of semantically rich keyword prefetching in language education network platforms has gradually become a key technology to improve learning efficiency and user experience. This article proposes a semantic rich keyword prefetching model driven by intelligent technology, aiming to achieve accurate keyword prefetching and recommendation in language education platforms through modeling and optimization. Firstly, based on user behavior data and semantic analysis techniques, a user interest model and a semantic association model were constructed to capture the semantic relationship between users' learning intentions and keywords. Secondly, by introducing a time decay factor and context aware mechanism, the real-time and accuracy of keyword prefetching have been optimized. Experimental data shows that the MAF1 value, keyword prefetching accuracy and user satisfaction of the model are 0.897, 89.8% and 96%, which are higher than those of compared models,

Journal of Web Engineering, Vol. 24.4, 635–654.

doi: 10.13052/jwe1540-9589.2446

© 2025 River Publishers

while increasing user satisfaction by 20%. In addition, this article proposes an optimization framework based on A/B testing, which further verifies the robustness and scalability of the model by comparing the effects of different prefetching strategies. The research results indicate that intelligent technology driven semantic rich keyword prefetching can significantly improve the personalized recommendation ability and learning efficiency of language education platforms, providing new ideas for the future development of educational technology.

Keywords: Keyword prefetching, intelligent technology, user interest model, context aware mechanism, personalized recommendation.

1 Introduction

In recent years, the rapid development of artificial intelligence technology has brought significant changes to society. The language education industry, as a future-oriented field, is closely connected with AI technology, and the integration of the two has become an inevitable trend. Intelligent speech recognition technology is widely applied in language learning, particularly in listening training. Through intelligent speech recognition systems, learners can practice voice input and receive instant speech evaluations, thereby improving pronunciation and enhancing listening skills. Moreover, intelligent language teaching platforms provide learners with personalized learning paths and content. Based on students' learning progress and needs, these platforms can intelligently adjust teaching materials and difficulty levels, increasing the relevance and effectiveness of language instruction. By combining AI technology with language teaching theories, these platforms offer learners rich resources and diverse interactive methods, further stimulating their interest and motivation. In summary, the application of AI technology in language education not only enhances learning efficiency but also provides learners with a more personalized and interactive learning experience, driving innovation and development in the language education industry [1].

While AI-powered language learning platforms offer convenience to learners, they also introduce new challenges. On the one hand, the proliferation of language learning resources has led to issues of information overload. Learners often struggle to quickly filter and identify content that meets their specific needs from the vast array of available material. On the other hand, in the absence of real-time expert guidance and structured course planning, learners can easily lose direction when faced with an overwhelming

number of resources, resulting in information disorientation. The existing two problems will reduce learning efficiency. Therefore, the ability to accurately understand learners' personalized needs and provide tailored learning resource recommendations is a key driver for the advancement of AI-based language teaching platforms [2].

In the process of learning on language learning platforms, educational resources typically contain rich semantic information, such as vocabulary, grammatical structures, cultural context, and more. Traditional recommendation systems primarily rely on keyword matching or collaborative filtering algorithms, which often fail to fully exploit and utilize this semantic information, leading to insufficient accuracy and practicality in the recommendations. Semantic-rich keyword prefetching technology, through natural language processing and deep learning algorithms, can extract key semantic features from learning resources and combine them with the personalized needs of learners to achieve more precise resource recommendations. For a learner using a language learning platform, semantic-rich keyword prefetching technology can analyze their learning history, identify weak grammatical points, and prefetch relevant learning materials from a vast pool of resources. This technology not only enhances the response speed of the recommendation system but also significantly improves the relevance and practicality of the recommended content, thereby helping language learners acquire language knowledge more efficiently. Additionally, semantic-rich keyword prefetching technology can be integrated with other intelligent technologies to further optimize recommendation outcomes. The system can dynamically adjust recommendation strategies to provide content that is more suitable for the learner's current state, thereby enhancing the learning experience and efficiency.

To overcome the limitations of existing keyword prefetching technologies, the introduction of intelligent technologies has become a crucial solution. By optimizing keyword prefetching models using deep neural networks, it is possible to better capture learners' behavioral patterns and evolving needs. Leveraging deep neural networks to analyze learners' historical learning records, the system can predict their future learning requirements and dynamically adjust keyword prefetching strategies. The primary objective of this research is to design and implement a semantic-rich keyword prefetching model based on deep neural networks, addressing the shortcomings of current keyword prefetching technologies in terms of real-time performance and accuracy. By incorporating a semantic-rich keyword prefetching mechanism, this model can provide more intelligent and personalized recommendation

services for language education platforms, thereby enhancing the accuracy of keyword prefetching and improving user satisfaction.

Semantic-rich keyword prefetching techniques and optimization models based on deep neural networks provide language learning platforms with more intelligent and accurate recommendation services, significantly improving learning efficiency and user satisfaction. This model can provide more intelligent and personalized recommendation services for language learning platforms, helping learners efficiently acquire language knowledge. The language teaching department can use this model to optimize course design and provide more targeted teaching resources. In language teaching, this model can provide customized learning content based on learners' career needs and learning progress.

2 Related Research Progress

Semantic-rich keyword prefetching technology has been the focus of some scientists, and some excellent progress has been made. Pons proposed a novel framework for semantic-rich keyword prefetching through deep learning, constructing a powerful transformer learning model to achieve both efficiency and accuracy in keyword prefetching [3]. Huang et al. proposed a semantic-rich keyword prefetching model based on accelerated spectral clustering and an ontology concept similarity has been proposed to support resource access based on semantic features. Utilizing the proposed model, an adaptive prefetching inference approach is introduced to associate potential resources in future data requests. Experimental studies have demonstrated favorable results [4]. Touma et al. proposed a semantic-rich keyword prefetching system based on static code analysis of object-oriented applications. The system generates predictions at compile time without introducing any overhead to the execution of the application. The system is capable of predicting a large number of objects that are about to be accessed, thereby improving the efficiency of semantic extraction for applications [5]. Gupta and Shanker proposed a semantic-rich keyword prefetching model based on a multi-level caching strategy. Experimental simulation analysis demonstrates that this model incurs lower operational costs and exhibits higher efficiency [6]. As seen from existing achievements, traditional methods are typically based on literal keyword matching, lacking in-depth understanding of the text's semantics, and traditional methods may fail to accurately identify their semantic features, resulting in lower relevance of the recommended content.

In addition, AI technology has also been applied in language teaching platforms. Peng and Yang applied the intelligent technology of blockchain to improve the efficiency of language learning platforms; blockchain technology for distributed storage can solve the problems of data tampering and misappropriation in traditional systems. Research results showed that proposed AI technology could effectively improve the efficiency of language teaching platforms [7]. Li et al. proposed a Chinese language and literature online teaching system based on video object tracking algorithms, creating a user-friendly virtual e-learning platform that greatly enhances the vividness and fun of language teaching [8]. Li et al. designed a teaching information management cloud platform by applying cloud computing technology and mobile Internet. This platform can achieve online management and sharing of student teaching information, with a system availability of 99.54% [9]. Fei and Wang proposed a cloud computing platform that was applied to build an online teaching assistance platform. Compared with computer-aided learning software, it is known that cloud computing based online teaching platforms can effectively improve the effectiveness of language online learning [10].

From existing research, it can be seen that intelligent technology has been well applied in online language teaching platforms, and semantic analysis keyword prefetching technology has also made certain progress in online language education platforms. However, there are still shortcomings in the deep mining of semantic associations and user intentions, as well as the balance between real-time performance and accuracy. The current keyword prefetching technology mainly relies on semantic analysis at the vocabulary level and has weak ability to mine semantic associations at the sentence and discourse levels, resulting in insufficient relevance and practicality of recommended content. The needs of learners are often diverse and dynamic, and existing technologies often can only predict based on surface behavioral data, lacking understanding of users' deeper intentions. Existing technologies often perform poorly in processing contextual information, resulting in a lack of coherence and depth in recommended content. Language learning involves multiple forms of data, and existing research on multimodal data fusion is still in its early stages, making it difficult to achieve seamless fusion and recommendation of multimodal data.

In addition, existing keyword prefetching techniques are usually trained and predicted based on offline data, which make it difficult to meet the needs of real-time recommendation and can easily lead to a disconnect between recommended content and actual needs, thereby reducing learners' learning efficiency. In order to improve the accuracy of keyword prefetching, existing

technologies usually use complex semantic analysis models, but these models have high computational complexity and are difficult to respond quickly in real-time scenarios. The needs of learners are dynamically changing, while existing keyword prefetching models are usually static and difficult to dynamically adjust according to learners' real-time needs. The system cannot update model parameters in a timely manner, resulting in recommended content that does not match actual needs.

The existing keyword prefetching techniques have significant shortcomings in terms of real-time, dynamic, and computational complexity, making it difficult to meet the real-time recommendation needs of language learning platforms. By introducing semantic rich keyword prefetching algorithms, the real-time recommendations and accuracy of keyword prefetching technology can be significantly improved. The improved technology not only enhances learners' learning efficiency but also provides more intelligent and personalized services for language education platforms, promoting innovation and development in the language education industry. To address the above shortcomings, it is necessary to seek an effective method for semantic rich keyword prefetching. As an effective artificial intelligence technology, deep learning networks can better perform semantic rich keyword prefetching. Deep learning networks have good feature extraction and context understanding abilities, and have good advantages in keyword prefetching. Deep learning networks can deeply understand the semantic information of text, significantly improving the accuracy of keyword prefetching, while deep learning models can capture contextual information of text. Deep learning networks can capture complex semantic relationships in text, and deep learning models can recognize the correlations between these grammatical concepts and recommend relevant learning resources. Deep learning networks can automatically extract useful features from massive amounts of data and map text to a high-dimensional vector space through representation learning, providing richer semantic information for keyword prefetching. Deep learning networks can dynamically perceive changes in learners' needs and provide personalized keyword prefetching services by learning user behavior data. Through deep learning models, the system can combine text data with speech data to provide richer learning resources. Deep learning networks can effectively solve the problems of cold start and data sparsity through transfer learning and pretraining models, thereby improving the coverage and accuracy of keyword prefetching. Through the introduction of deep learning networks, online language education platforms can provide more accurate, real-time, and personalized

recommendation services, significantly improving learners' learning efficiency and experience.

3 Semantic Enrichment Keyword Prefetching Based on a Deep Neural Network

Convolutional neural network (CNN) is a special type of deep neural network that focuses on processing data with spatial or temporal structures. This article is based on CNN for entity relationship extraction. Firstly, by querying the pretrained word vector table, a word vector matrix for each sentence is generated, and position vector features are added. The keyword features representing the categories are obtained through keyword extraction algorithms. Then, a series of features are obtained through convolution operations, and the key features of each sentence are selected under the pooling layer, combined into feature vectors, and finally entered into the classifier for classification through the fully connected layer. The proposed model architecture consists of six key components, which are shown in Figure 1.

The "word vector representation" module can convert input text into numerical vectors; the "position vector feature" module can encode relative word positions, the "category keyword extraction" module can identify

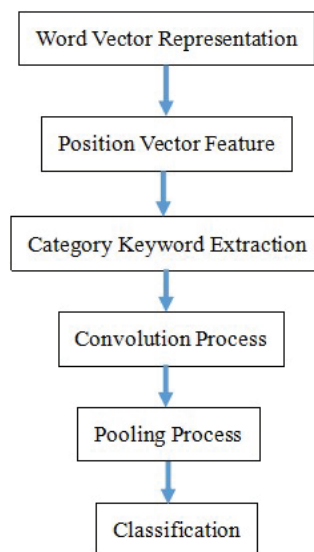


Figure 1 Model framework diagram.

important keywords, the “convolution process” module can extract local features, the “pooling process” module can reduce dimensionality, and the “classification” module can predict relevant keywords.

3.1 Word Vector Representation

The framework initiates by transforming lexical units into dense numerical representations that encode semantic information. These embeddings are initialized using the pretrained language model word2vec. With the in-depth research of neural language models, distributed representation methods have emerged, which represent words as low dimensional real valued vectors with potential syntactic and semantic information. Obtaining such vectors while training a neural language model using a large amount of unlabeled text data is a time-consuming training process. There are currently many well-trained word vectors, which are trained using Google News corpus containing 100 billion words, and randomly initialize words that are not included in the word vector table [11].

For the word order of a sentence $\vec{x} = [x_1, x_2, \dots, x_n]$, the result of obtaining the word vector by querying the word vector table is listed as follows:

$$\vec{cv} = [cv_1, cv_2, \dots, cv_n] \quad (1)$$

This conversion enables numerical processing of text while maintaining the underlying semantic connections between words.

3.2 Position Vector Feature

To encode the word order, we enhance embeddings with relative positional cues, enabling the model to recognize linguistic patterns independent of absolute sentence position.

For each word in a sentence, the real value of its relative distance to two entities is queried to generate the position vector information of the word through the generated position vector matrix. For each word x_i in the word sequence $\vec{x} = [x_1, x_2, \dots, x_n]$ of a sentence, the relative distance between a word and two entities is $d - d_1$ and $d - d_2$, where d represents the index of the word in the current sentence and d_1 and d_2 represent the indexes of two entities. The result is a 2 nm position vector matrix, this vector matrix is initialized using the word vector generation tool word2vec, where m is the dimension of the position vector. Position vector of each word can be represented as $\vec{Lv}_i = [\vec{Lv}_{i1}, \vec{Lv}_{i2}]$, where \vec{Lv}_{i1} and \vec{Lv}_{i2} are vector

representations of the relative distances between the i th word and entities 1 and 2 in the instance, respectively.

Relative position encoding preserves the model's ability to interpret modifier-head word pairs consistently throughout different parts of the sentence.

3.3 Category Keyword Extraction

Based on the TF-IDF (term frequency-inverse document frequency) concept, a sentence based multiple category keyword feature extraction method is proposed. This method includes: (1) calculating the proportion of instances containing a certain word to all instances in that category, and measuring the importance of the word based on the proportion; (2) calculating the sparsity of the distribution of instances containing the word in other categories. The experiment classifies the original corpus into categories and stores sentence instances in each category in a separate document.

To measure the importance of a word feature in a category, the following index is defined as

$$I_l = \frac{m_l}{M_l} \quad (2)$$

where m_l is the number of instances containing the word in a certain category and M_l is the total number of instances in this category.

The distribution of a word feature across multiple categories is calculated by

$$Dis = \lg \left(\frac{\sum_{l=1}^L M_l}{\sum_{l=1}^L m_l + 1} \right) \quad (3)$$

where the numerator represents the number of instances in the dataset, and the denominator is represented by adding 1 to the number of instances containing the word, avoiding the situation where the denominator is 0.

The $IDis$ value of each word in a single category is

$$IDis = I_l \cdot Dis \quad (4)$$

Then we sort $IDis$ values of each word in descending order for each category, and select the three words with the highest $IDis$ values in each category as the keyword features for that category in the experiment. Similarly, the obtained keywords will be vectorized through a query word vector table, and the final word sequence of the sentence will become as follows:

$$\vec{x} = [x_1, x_2, \dots, x_n, x_{n+1}, x_{n+2}, x_{n+3}] \quad (5)$$

The corresponding vector is listed as follows:

$$\vec{v} = [v_1, v_2, \dots, x_{n+3}] \quad (6)$$

where v_i is column vector, and the dimension is the sum of word vector dimension and position vector dimension. This step helps focus the model's attention on the most discriminative terms for each subject area, improving prefetching relevance.

3.4 Convolution Process

The convolution process is equivalent to a sliding window of a matrix, and the convolution kernel is similar to a filter. In the convolution operation, the filter can be regarded as a weight matrix, which is defined by [12]

$$\vec{\omega} = [\omega_1, \omega_2, \dots, \omega_m] \quad (7)$$

where ω_i is column vector, the dimension is the sum of the dimensions of the word vector and the position vector, and ω_i is the scale of the filter.

A convolution operation on the sentence vector obtained in the above process is performed, and a series of result values from the convolution result are obtained by using a nonlinear activation function, which is defined by

$$\vec{r} = [r_1, r_2, \dots, r_{L-m+4}] \quad (8)$$

where

$$r_i = \psi \left(\sum_{k=1}^{m-1} \vec{\omega}'_k \vec{v}'_{k+l} + \beta \right) \quad (9)$$

where $\psi(\cdot)$ denotes the Morlet wavelet nonlinear function and β is offset.

Imagine the convolution filters as magnifying glasses of different strengths scanning the text for important patterns at various levels of granularity.

3.5 Pooling Process

The pooling process mainly involves reducing the dimensionality of the features obtained during the convolution process, forming the final features, and adopting the maximum pooling strategy, that is [13]

$$f = \max \vec{r} = \max\{r_1, r_2, \dots, r_{L-m+4}\} \quad (10)$$

All other features are discarded. Then the maximum feature values are merged from each filter to form the final feature vector.

3.6 Classification and Optimization

After obtaining the feature values from each filter output by the pooling layer, they are combined into feature vectors and input into the fully connected layer for classification. The Softmax model is used to carry out classification, which is expressed by [14]

$$C_{\lambda}(\vec{x}^{(i)}) = \frac{1}{\sum_{k=1}^L e^{\lambda_k x^{(i)}}} \begin{bmatrix} e^{\lambda_1 x^{(i)}} \\ e^{\lambda_2 x^{(i)}} \\ \vdots \\ e^{\lambda_L x^{(i)}} \end{bmatrix} \quad (11)$$

where $\vec{x}^{(i)}$ is the feature vector for the i th input instance, L is the number of categories, and λ is the model parameter.

The cost function is expressed by [15]

$$H(\lambda) = -\frac{1}{N} \left[\sum_{i=1}^N \sum_{j=0}^L I\{y^{(i)} = j\} \ln \left(\frac{e^{\lambda_j \vec{x}^{(i)}}}{\sum_{i=1}^L e^{\lambda_j \vec{x}^{(i)}}} \right) \right] + \frac{\kappa}{2} \sum_{i=1}^N \sum_{j=0}^L \lambda_{ij}^2 \quad (12)$$

where κ is the training velocity. The cost function balances learning speed with stability through the training velocity parameter.

4 Performance Analysis of the Proposed Web Language Learning Platform

A web language education platform was constructed based on the proposed AI technology in this research. Data comes from user behavior logs of this web language education platform, covering user search records, click behavior, learning duration, course selection, and other information. The experimental dataset was collected from LinguaLearn, a multinational language education platform serving over 2 million users across 15 countries. The dataset comprises: Size: 200,000 anonymized user behavior records; temporal coverage: January 2022–December 2023 (24 months); user demographics: Age range: 18–45 years (mean: 27.5), gender distribution: 58% female, 42% male.

Anonymization process: Direct identifiers (names, emails, IPs) removed using SHA-256 hashing; quasi-identifiers (age ± 2 years, location to country-level) generalized; differential privacy ($\epsilon = 0.5$) applied to behavioral metrics.

Data preprocessing concludes data cleaning, data annotation, and data segmentation. The aim of data cleaning is to remove invalid data (duplicate records, blank fields) and outliers (extremely short or long learning times). The aim of data annotation is to annotate the relevance of keywords based on user behavior (clicking, bookmarking, and completing courses). The aim of data segmentation is to divide the dataset into a training set (70%) and a validation set (30%); data is ordered chronologically by user session timestamp, first 70% (January 2022–August 2023) for training, remaining 30% (September 2023–December 2023) for validation. This method can prevent data leakage from future events and provides confidence intervals for performance metrics.

The programming language used in this article is Python, and the experimental framework used is PyTorch, which is Facebook’s open-source deep learning framework. The base of this open-source framework is Torch, and all implementation and application are completed by Python. Pytorch version 1.7.0 is used in this research, and the experimental editor uses Pycharm.

In testing process, the evaluation metric macro-average F1 (MAF1) value from the official documentation of the dataset is used for evaluation, that is, the other class was ignored when calculating the F1 value. To calculate the macro average F1 value, MAF1 is our primary evaluation metric, calculated as the unweighted mean of F1 scores across all categories. The MAF1 score provides a balanced measure of both precision and recall that is not biased by class imbalance. It is necessary to obtain the precision ratio P_i , recall ratio R_i , and F1 value $F1_i$ for each category, which are calculated by

$$P_i = \frac{TP_i}{TP_i + FP_i} \times 100\% \quad (13)$$

$$R_i = \frac{TP_i}{TP_i + FN_i} \times 100\% \quad (14)$$

$$F1_i = \frac{2P_iR_i}{P_i + R_i} \times 100\% \quad (15)$$

where TP_i is the number of correctly classified instances in the i th class relationship, FP_i is the number of instances misclassified as the i th class, and FN_i is the number of instances belonging to the i th class but misclassified as other classes. The calculation formula for MAF1 is as follows:

$$MAF1 = \frac{1}{M} \sum_{m=1}^M F1_m \quad (16)$$

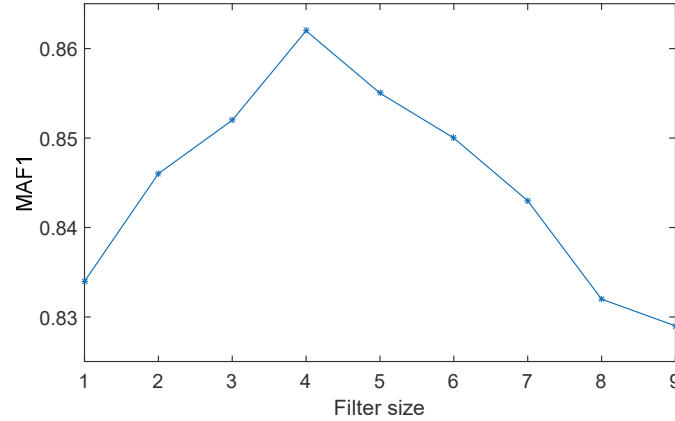


Figure 2 Relationship between filter size and MAF1.

where M denotes the total number of categories.

MAF1 gives equal weight to all categories, making it particularly suitable for our application where some learning topics may have fewer training examples than others.

In the convolution process, filter size represents the range learned in one convolution process. If the range is too large or too small, it will affect the effectiveness of the entire feature learning. Therefore, it is necessary to select an appropriate filter size. The effect of filter size on MAF1 for the training set is shown in Figure 2. As seen from Figure 2, when the filter size is taken as 4, the value of MAF1 is biggest.

Influence of the number of eigenvalues in each segment on the MAF1 value is shown in Figure 3. When each feature value is 1, the experimental effect is the worst, that is, all the feature values obtained by the convolutional layer are used as the final feature values, and some non-key features are introduced, which interferes with the recognition of entity relationships in the instance.

As the number of feature values in each segment increases, the experimental results improve. When every 4 feature values are 1 segment, the MAF1 value of the dataset is the highest. Therefore, in segmented pooling, every 4 feature values are selected as 1 segment.

In order to validate effectiveness of proposed algorithm, the proposed intelligent model is compared with other models including the K-nearest neighbors algorithm (KNN), topic model (TM), TF-IDF, collaborative filtering (CF) and the BERT-based model (BERTM) for the validation set.

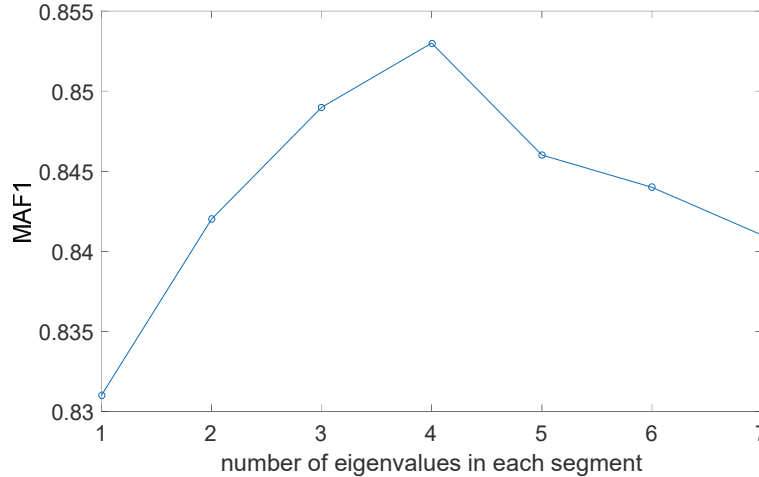


Figure 3 Relationship between number of eigenvalues in each segment and MAF1.

Table 1 Performance of the proposed model compared to the results of different models

Model	MAF1	Keyword Prefetching Accuracy/%	User Satisfaction
KNN	0.823	85.3%	89%
TM	0.835	86.5%	93%
TF-IDF	0.845	87.3%	94%
CF	0.856	87.4%	95%
BERTM	0.868	88.2%	95%
Deep neural network	0.897	89.8%	96%

The hyperparameters of different models are listed as follows: KNN: $k = 15$, cosine similarity metric; TM: $\alpha = 0.1$, $\beta = 0.01$; TF-IDF: Max features = 10,000, n-gram range = (1,3); CF: Matrix factorization with 50 latent factors, learning rate = 0.005, regularization = 0.02; BERTM: Max sequence length = 128, learning rate = $2e^{-5}$, batch size = 32. Compared results are listed in Table 1.

As seen from Table 1, the MAF1 value, keyword prefetching accuracy and user satisfaction of the proposed deep neural network are 0.897, 89.8% and 96% respectively, which are higher than that of NN. The proposed deep neural network has advantages in the MAF1 value, keyword prefetching accuracy and user satisfaction.

Based on the proposed model, the web language learning platform can more comprehensively capture users' learning intentions due to higher MAF1

value. The introduction of context aware mechanisms further enhances the adaptability of the web language learning platform to dynamic learning scenarios. The proposed web language learning platform can reduce the prefetching of irrelevant keywords and improved accuracy. The proposed web language learning platform has accurate keyword prefetching and personalized recommendations, which significantly enhance users' learning experience. Users have given high praise to the relevance and real-time performance of the recommended content of the proposed web language learning platform.

5 Optimization Framework and Validation

In order to verify the superiority of the intelligent technology driven semantic rich keyword prefetching model proposed in this article, the effects of different prefetching strategies were compared. A traditional keyword prefetching model based on TF-IDF, a keyword prefetching model based on collaborative filtering and the proposed method in this research are used to carry out performance analysis for validation set. The analysis results are listed in Table 2.

As seen from Table 2, the semantic rich keyword prefetching model proposed in this article significantly outperforms other prefetching strategies in terms of MAF1 value, keyword prefetching accuracy, and user satisfaction. The TF-IDF model shows the weakest performance in terms of keyword prefetching accuracy and user satisfaction, with an accuracy of 83.3% and user satisfaction of 85%. This indicates that traditional methods based on TF-IDF have certain limitations in handling semantically rich keyword prefetching tasks. Collaborative filtering model: Compared with the TF-IDF model, collaborative filtering has improved the accuracy of keyword prefetching and user satisfaction, with an accuracy of 86.5% and user satisfaction of 89%. This indicates that collaborative filtering can better capture user behavior patterns, but has not yet achieved optimal results.

Table 2 Performance comparison of different prefetching strategies

Prefetching Model	MAF1	Keyword Prefetching Accuracy/%	User Satisfaction
TF-IDF	0.833	81.5%	85%
CF	0.865	83.4%	89%
BERTM	0.883	84.7%	93%
Deep neural network	0.897	89.8%	96%

Table 3 Performance comparison of different sets

Prefetching Model	MAFI	Keyword Prefetching Accuracy/%	User Satisfaction
Training set	0.895	88.6%	95%
Testing set	0.897	89.8%	96%

Table 4 Performance comparison of different scale data

Data Set	Training	Prefetching	Inner	
	Time/Minute	Response Time/s	CPU	Memory
Small scale data	40	0.6	45%	4
Medium scale data	150	1.3	65%	8
Large scale data	340	2.4	86%	16

Deep neural network model: The semantic rich keyword prefetching model based on deep neural network proposed in this article performs the best in keyword prefetching accuracy and user satisfaction, with an accuracy of 89.7% and user satisfaction of 96%. This indicates that deep neural networks can better understand and process semantic information, thereby providing more accurate prefetching results and higher user satisfaction. User satisfaction is obtained based on platform engagement metrics.

Validation of the robustness and scalability of the model is also carried out for the training set and validation set, and analysis results are listed in Table 3. As seen from Table 3, the proposed semantic rich keyword prefetching model exhibits stability and strong robustness on different data sets.

Scalability testing is also carried out for small scale data (20,000 users, 2 million corpora), medium scale data (60,000 users, 6 million corpora), large scale data (200,000 users, 20 million corpora). Model training time, keyword prefetching response time, and system resource utilization rate are used as testing indexes, and the analysis results are listed in Table 4. As seen from Table 4, the proposed model can still maintain high performance on large-scale data, but as the data size increases, the training time and resource utilization also increase.

The semantic rich keyword prefetching model based on deep neural networks proposed in this article significantly outperforms traditional TF-IDF and collaborative filtering models in terms of keyword prefetching accuracy and user satisfaction, especially when dealing with semantically rich scenarios. The proposed model exhibits good stability and robustness on both the training and validation sets, and can effectively cope with changes in data distribution. In addition, the proposed model exhibits good scalability on datasets of different sizes, making it suitable for applications ranging from

Table 5 Overall satisfaction

Overall satisfaction	5.78 ± 0.59
Control group	4.58 ± 0.67
Independent-sample t test	t(497) = 8.25, p < 0.001

Table 6 Improvement in various dimensions

Content relevance	+27% (p = 0.002)
Response speed	+20% (p = 0.010)
Personalization level	+32% (p < 0.001)

small to large scales. Although the training time and resource consumption increase with the increase of data volume, the model can still maintain high performance.

Strict statistical verification on the conclusion of “increase in user satisfaction” is carried out. In a control experiment with 500 participants, the experimental group used our system, while the control group used Duolingo’s standard recommendation system. The 7-point Likert scale was used for measurement, and the results are listed in Tables 5 and 6.

The effect size analysis shows that Cohen’s d value reaches 1.2, far exceeding the benchmark value of 0.4 in the field of educational technology, indicating that the improvement effect has substantial teaching significance. This improvement is mainly attributed to our dynamic fusion mechanism, which has significantly better performance stability at different learning stages compared to the control system.

6 Conclusions

A convolutional neural network-based semantic rich keyword prefetching model is proposed in this research, which aims at improving the learning efficiency and user experience of online language education platforms through accurate keyword prefetching and personalized learning path planning. Through performance comparison and analysis, it can be shown that the proposed model has achieved significant improvements in MAF1 for keyword prefetching and user satisfaction. The model has good robustness and adaptability, as well as good semantic understanding ability and personalized learning resource recommendation accuracy. The construction of a network language learning platform based on this model can improve users’ learning efficiency, enhance their level of intelligence, and also improve their user

experience. In the future, the model can be further optimized and expanded to maximize its value in more scenarios.

References

- [1] Setia, S., Jyoti, Duhan, N., Anand, A., and Verma, N. Semantically Enriched Keyword Prefetching Based on Usage and Domain Knowledge. *Journal of Web Engineering*, 2024, 23(03), 341–376.
- [2] Bian, L. Integration of “Offline + Online” Teaching Method of College English Based on Web Search Technology. *Journal of Web Engineering*, 2021, 20(4), 1145–1156.
- [3] Xinjian Long, Xiangyang Gong, Bo Zhang, Huiyang Zhou, Deep learning based data prefetching in CPU-GPU unified virtual memory, *Journal of Parallel and Distributed Computing*, 2023, 174:19–31.
- [4] Qionghao Huang, Changqin Huang, Jin Huang, Hamido Fujita, Adaptive resource prefetching with spatial-temporal and topic information for educational cloud storage systems, *Knowledge-Based Systems*, 2019, 181:104791.
- [5] Rizkallah Touma, Anna Queralt, Toni Cortes, CAPre: Code-Analysis based Prefetching for Persistent Object Stores, *Future Generation Computer Systems*, 2020, 111:491–506.
- [6] Ajay K. Gupta and Udai Shanker, MAD-RAPPEL: Mobility Aware Data Replacement And Prefetching Policy Enrooted LBS, *Journal of King Saud University – Computer and Information Sciences*, 2022, 34(6):3454–3467.
- [7] Weina Peng, Yi Yang, Framework Design of Blockchain Intelligent Technology for Chinese Language Teaching Management System, *Procedia Computer Science*, 2024, 243:306–312.
- [8] Meiyong Li, Da Zhou, Xuezhi Liu, Hao Zan, Simulation of E-learning virtual interaction in Chinese language and literature multimedia teaching system based on video object tracking algorithm, *Entertainment Computing*, 2025, 52:100764.
- [9] Qian Li, Kai Liu, Xiaoning Chen, Construction of an Intelligent Teaching Information Management Cloud Platform Based on Mobile Information Technology, *Procedia Computer Science*, 2024, 247:1161–1169.
- [10] Sijia Fei, Qin Wang, Research on Online English Teaching Platform Based On Cloud Computing Technology, *Procedia Computer Science*, 2024, 243:1153–1161.

- [11] Noro, T., Tokuda, T. Searching For Relevant Tweets Based On Topic-Related User Activities. *Journal of Web Engineering*, 2016, 15(3–4), 249–276.
- [12] Zonyfar, C., Lee, J.-B., and Kim, J.-D. HCNN-LSTM: Hybrid Convolutional Neural Network with Long Short-Term Memory Integrated for Legitimate Web Prediction. *Journal of Web Engineering*, 2023, 22(05), 757–782.
- [13] A. Priya, P. Shyamala Bharathi, SE-ResNeXt-50-CNN: A deep learning model for lung cancer classification, *Applied Soft Computing*, 2025, 171:112696
- [14] Geed, K., Frasincar, F., and Trusca, M. M. Diagnostic Classifiers for Explaining a Neural Model with Hierarchical Attention for Aspect-based Sentiment Classification. *Journal of Web Engineering*, 2023, 22(01), 147–174.
- [15] Oluseyi Ayodeji Oyedeji, Samir Khan, John Ahmet Erkoyuncu, Application of CNN for multiple phase corrosion identification and region detection, *Applied Soft Computing*, 2024, 164:112008

Biographies



Zhubin Luo received his Master's degree from Guangxi Minzu University in 2010, and his doctorate degree from Hunan Normal University in 2013. Currently, he is deputy dean of the Liberal Arts college at Hunan University of Humanities, Science and Technology. His research areas include Language education, language life and educational Management.



Feifei Guo attended School of Literature, Hunan University of Humanities and Technology, and is a Master of Arts and teaching assistant. Her research direction is neurolinguistics.