Hindawi
Mathematical Problems in Engineering
Volume 2022, Article ID 9883201, 13 pages
https://doi.org/10.1155/2022/9883201

Research Article

Numerical Analysis and Optimization of Feature Extraction-Oriented English Reading Corpus

Yue Li

Department of Teacher Education, Zhumadian Vocational and Technical College, Zhumadian 463000, China

Correspondence should be addressed to Yue Li; 164904224@stu.cuz.edu.cn

Received 7 May 2022; Accepted 3 June 2022; Published 29 June 2022

Academic Editor: Xiantao Jiang

English is a universal language in the world. It has become the consensus of society as a subject of education in primary and secondary schools and even universities. Therefore, how to improve English reading ability has also become a focus area of school education and students. The current research on English reading is mainly based on the sense of reading questions, reading patterns, answering skills, etc. and lacks the analysis of English reading corpus. In view of this, this paper used a self-built English reading corpus, adopts the feature extraction method, and combines the convolutional neural network (CNN) to build a model to carry out numerical analysis on the self-built English reading corpus, optimized the model, and compared and analyzed the results obtained. The optimal dropout rate and iteration times were obtained by updating experimental parameters. In order to make the experimental results more convincing, the W2V-SVM and W2V-CNN models that combine different feature extraction and classification methods are designed. Compared with the optimized CNN model, the accuracy rate, recall rate, and $F1$ value of the optimized CNN model were 89.81%, 92.39%, and 92.8%, respectively. The accuracy, recall, and $F1$ value of the W2V-SVM model are 81.31%, 82.09%, and 81.25%, respectively. The accuracy, recall, and $F1$ value of the W2V-CNN model are 85.24%, 84.98%, and 85.12%, respectively. It shows that the optimized CNN feature classification model has better feature classification effect on the self-built English reading corpus. The experimental results meet the expected value.

1. Introduction

The corpus is the language material stored by the computer, and its language material is composed of the language material that actually exists in practice. Therefore, the corpus contains a wealth of linguistic knowledge such as language, vocabulary, grammar, etc., which has a certain impact on students' English learning, the improvement of teachers' teaching methods, and even the research of linguistic researchers. However, as a resource for language expression, it requires learners to process and analyze it by themselves in the process of using it. As one of the main languages used in the world, English has a position that cannot be ignored in school teaching. English reading ability is an important indicator for evaluating English learning. In the actual English learning process, the English examination mainly examines the students' English reading ability, and improving the students' English reading ability has also become a major difficulty for teachers in teaching. The emergence of corpus can help teachers cultivate students' good English reading habits and can also analyze students' mistakes in learning English through examples of corpus and promote the realization of English teaching goals. By analyzing the characteristics of the English reading corpus, the composition of the vocabulary and sentences in the actual English reading can be obtained, which can provide some data support for English reading teaching. At the same time, it can also help students choose a more effective learning method in English reading and provide a reference basis. To sum up, the study of English reading corpus has certain practical significance for the improvement of English reading ability.

In view of the importance of English corpus in English learning, many researchers have done research on English corpus. Yanez-Bouza and Gonzalez-Diaz [1] conducted related research on the content and compilation process of

APU writing and English reading corpora through the operation process of children's language comprehension [1]. Starting from the English reading corpus, Lee et al. [2] studied the correlation between the English corpus and learners' mastery of vocabulary and solved the problem of vocabulary learning by encouraging learners to self-analyze and construct an English reading corpus [2]. Shatz [3] analyzed the role of capitalization in language processing and writing assessment during reading from the capitalization error patterns in an English reading corpus constructed from 133,000 texts of 38,000 foreign learners [3]. Guziurová [4] constructed an English reading corpus from English-type articles written by nonnative English-speaking research scholars and compared it with an English reading corpus constructed based on articles written by English writers designed by SciELF [4]. Oveshkova [5] developed a task and activity system based on the English corpus and took the students of a foreign language college as the experimental object to observe the effect of the language learning method based on the corpus [5]. Ryu et al. [6] investigated the textual difficulty of reading materials of English textbooks in Korean middle schools and analyzed the linguistic and psycholinguistic characteristics of English texts and textbooks by constructing an English reading corpus [6]. From the above studies, we can see that the current research direction of the English reading corpus is basically the research on its function, and the basic framework of the English reading corpus, such as vocabulary and word attributes, is rarely researched and analyzed. However, this qualitative analysis cannot fully meet the needs of modern learners, so it is necessary to consider refining the problem and conduct related research from the composition of the succession of the English reading corpus.

In recent years, CNN has become an algorithm used in research in many disciplines. Evo and Avramovi [7] built a classification model for aviation and target image detection through CNN [7]. Based on the combination of CNN and fault detection and classification of semiconductor manufacturing process, Lee et al. [8] extract fault features for multivariate sensor signals [8]. Chen and Jahanshahi [9] proposed a model framework based on CNN and Naive Bayes for analyzing single video frame numbers for crack detection [9]. Palsson et al.'s [10] method is using CNN to fuse multispectral and hyperspectral images together to obtain high-resolution images [10]. Murillo et al. [11] extract image features by improving the training of CNN in NATLAB [11]. El-Sawy et al. [12] trained and tested a database of handwritten Arabic characters using CNN [12]. The use of CNN in current research is mostly image recognition and feature extraction, but since the weight and parameter settings of CNN in the convolution process will affect the accuracy of the data, only using CNN to train on research data may not achieve the expected results in research. Depending on the data attributes that need to be analyzed, the characteristics of the research data need to be considered when training, and the English reading corpus obviously has text characteristics. When using the CNN to analyze the English reading corpus, it is necessary to combine the characteristics of the text and integrate with other methods in order to obtain more accurate data.

The role of CNN in the field of feature recognition has been reflected in different disciplines. In this paper, CNN is combined with some common feature extraction methods to build a classification model and optimize the model. Combined with the numerical analysis of the English reading corpus created by tem-8 reading sections over the years, the classification accuracy of the model was obtained. The results show that, compared with TF-IDF, NNLM, and Word2Vec models, the feature classification model constructed in this paper is better. By comparing it with the CNN feature classification model before optimization, W2V-SVM, and W2V-CNN, it confirms that the optimized model is more practical. In this paper, the self-built English reading corpus is analyzed by the feature extraction method. In the use of method tools, based on the model structure of CNN, an experimental model integrating multiple methods is constructed, and in the design of the model, word vectors with different meanings of words are proposed according to the semantic features of the text, which is the innovation of this paper.

2. English Text Feature Extraction Method and Classification

2.1. Feature Extraction Method and Text Classification. Methods and tools to discover deep knowledge and certain pattern features from a large number of documents are called text mining [13]. Text mining is related to techniques in multiple fields, such as pattern recognition, statistics, data mining, and informatics [14]. As shown in Figure 1, it also involves Internet fields such as machine learning, artificial intelligence, and computer science.

Figure 1 shows some of the subject areas involved in text mining techniques. It can be seen that the tool carrier of text mining is a computer, so when performing text mining, the expression and classification of text should conform to the processing thinking of computer [15]. At present, the selection of text features is mainly to select a feature word from the text and quantify the feature word to express text information. Based on this, which algorithm to choose to process feature words has become the main research direction of text representation. When selecting feature words, the selected feature words not only need to be able to clearly identify the text information on the basis of being distinguishable from other texts, but also need to pay attention that the number should not be too many. At present, the main methods of feature selection are as follows: directly select from original features or use mapping transformation and select representative items according to expert knowledge, or use mathematical statistics to find the most obvious features. It selects directly from raw features or transforms with mapping and selects representative items based on expert knowledge. In addition to feature selection, the overall function of text classification also includes text preprocessing, statistics, and other steps. The general training process is shown in Figure 2.

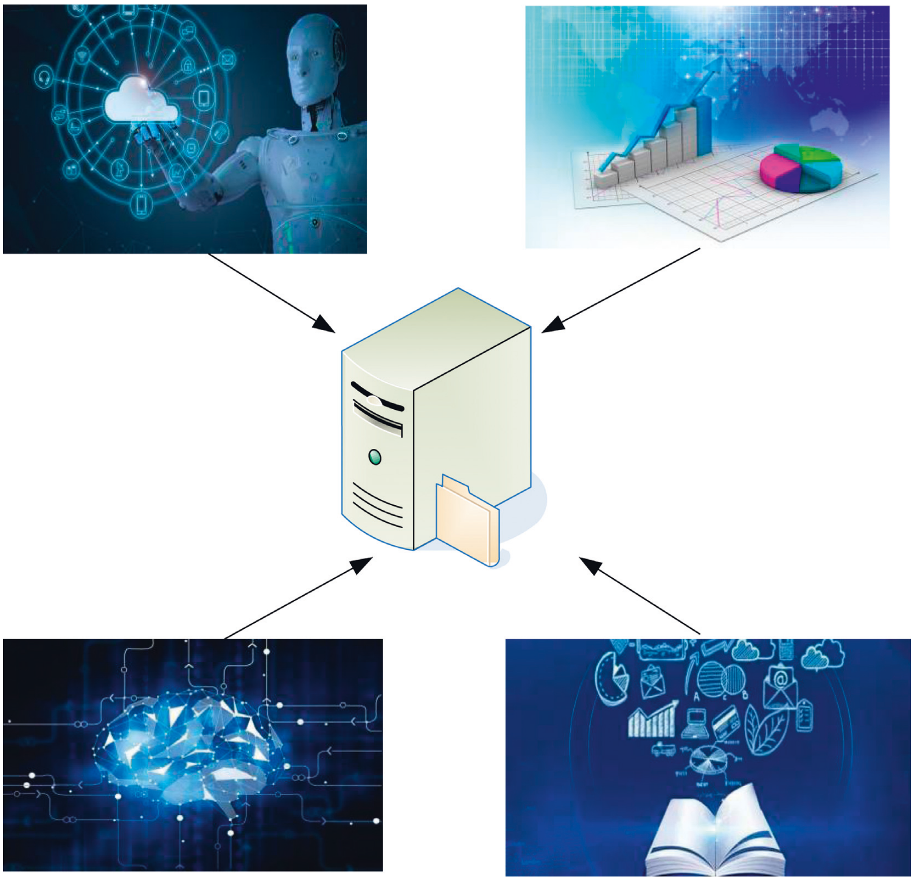

Figure 1: The text mining part covers the fields.

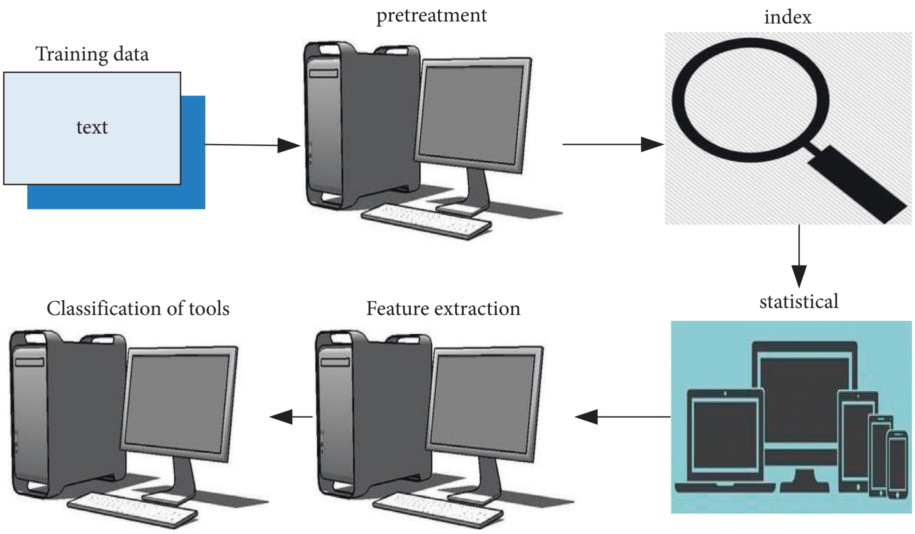

Figure 2: Basic process of text classification training.

As can be seen from the schematic diagram in Figure 2, after inputting the training set, the system first formats the original text data into the same format, which is the preprocessing process. The initial document is then decomposed into individual units, and the probabilities of items in the data associated with the classification are counted, which is the index and statistics of the system. After completing these basic steps, feature extraction is performed, and then a classifier is selected for classification.

Text classification methods mainly include word matching method, knowledge engineering, and statistical learning [16], of which word matching method is the earliest proposed method. The classification processing principle of word matching method is to classify according to the class name in the document. The same class name is the same class, and the different class name is not. The classification is too simple, so this method is rarely used at present. Knowledge engineering is to add the inference rules of human judgment into the classification system for classification. The labor cost of this method is high and the reasoning rules in each field need to be reconstructed, which is relatively cumbersome to use, so this method is also used less. The classification method of statistical learning is to first use the training set for model training and mining features for classification, and the techniques used have a solid theoretical basis [17], so this classification method is widely used. Statistical learning includes many algorithms.

Naive Bayesian classification method belongs to a class of Bayesian learning methods; its basic principle is to calculate the probability that a document in the text can be classified in a certain text category [18]. Assuming that A_x is a certain category in the document category, and B_y is any document in the document, the probability of document B_y belonging to each document category is calculated, and document B_y is classified as one of the document categories with high probability. The probability calculation formula is

$$p\left(A_x|B_y\right) = \frac{p\left(A_x\right)p\left(B_y|A_x\right)}{\sum_{x=1} p\left(A_x\right)p\left(B_y|A_x\right)}. \tag{1}$$

In formula (1), the precondition for calculating the probability is that the features of the document are independent of each other, which is unlikely to hold in actual text training. However, as long as the probability error is within the allowable range, the Naive Bayes classification method can still maintain a good classification effect.

Conditional random field is a conditional probability distribution model, which is mainly used for lexical analysis work such as part-of-speech tagging. The main feature is that when classifying data, the classification information of adjacent data can be considered. Therefore, problems such as classification bias can be well solved in use, and in this algorithm, all features will be normalized, and the global optimal solution will finally be obtained.

SVM mainly uses the kernel function to project the original sample space into a higher dimensional plane of space and then looks for a super-large plane in this high-order space. This superplane can split the two data planes and ultimately maximize the separation between the two split data planes [19].

This article will use these methods for comparative analysis. Since there is no guarantee of 100% classification effect in the classification, mainly because the classification effect cannot be guaranteed to be 100% in the classification, some data are misclassified, and some data are correctly classified, and this information can be represented by a binary classification mixture matrix, as shown in Table 1.

Table 1 is the binary classification mixture matrix. In this matrix, assuming that the training samples are divided into positive cases, where the number represents 1, and the number represents 0, then A, B, C, and D in the table represent the classification of these samples.

The accuracy rate can be expressed as follows according to the data in the table:

$$R = \frac{A + D}{A + B + C + D}. \tag{2}$$

According to the data in the table, the accuracy can be expressed as

$$R = \frac{A}{A + C}. \tag{3}$$

Precision represents the proportion of correctly classified items among all positive examples.

The formula for calculating recall is

$$R(\text{recall}) = \frac{A}{A + D}. \tag{4}$$

The recall rate represents the ratio of correctly classified positive examples in the correct classification, that is, how many positive examples in the training sample are correctly predicted.

The formula for calculating the $F1$ value is

$$F1 = \frac{2A}{2A + B + C}. \tag{5}$$

This paper will use the above indicators to measure the feature extraction-oriented numerical analysis model.

Common feature extraction methods are based on statistics and based on semantics. The former mainly constructs an evaluation function, independently evaluates and scores the feature sets in the training samples to independently evaluate and score the feature set in the training sample, and sorts them according to the size of the scores and finally extracts the optimal features.

Supposing that the training sample set is Q, m is the text in the set Q, and x is the feature of m.

The document frequency calculation formula is

$$F(x) = \{q|\text{if minQ}, q \in Q\}. \tag{6}$$

The calculation of document frequency is mainly to calculate the number of samples with a certain feature from all the training samples.

Information gain represents the average information of a feature in the text. Assuming that the text category is g_w, $p(x|g_w)$ represents the probability that x appears in this text category, and $p(x'|g_w)$ represents the probability that the text does not have this type of feature but belongs to this category, then the information gain expression formula is

$$\begin{aligned} \text{IG}(x) = p(x) &\sum_{w=1} p(x|g_w)\log\, p(x|g_w) \\ &+ p(x') \sum_{w=1} p(x'|g_w)\log\, p(x'|g_w) \\ &- \sum_{w=1} p(g_w)\log\, p(g_w). \end{aligned} \tag{7}$$

The expected cross entropy is similar to the information gain. The difference between it and the information gain is that the expected cross entropy does not consider the fact that the text does not exist, and its calculation formula is

Table 1: Binary classification mixture matrix.

–		Predictive value	
		+	–
Real value	+	Number of positive examples correctly classified (A)	Number of misclassified negatives (B)
	–	Number of misclassified positives (C)	Number of correctly classified negatives (D)

$$F(x) = p(m)\sum_{w=1} p(g_w)\log\frac{p(m|g_w)}{p(m)}. \tag{8}$$

Mutual information is a variable used to represent the correlation between features and text categories. The calculation formula is

$$G(x) = \log\frac{p(x,g)}{p(x)p(g)}, \tag{9}$$

where g represents the text category.

The word frequency is literally the frequency of a word, and the expression formula is

$$F(x) = P(m|Q). \tag{10}$$

The above is a feature extraction method from a statistical point of view.

Traditional text semantic feature extraction methods commonly used one-hot discrete model, bag-of-words model, and TF-IDF model.

One-hot discrete text semantic feature extraction is mainly trained according to word characteristics. It first builds a dictionary of words in the text and then uses 0 or 1 for each identical word in the dictionary. Assuming that there are 7 words in the dictionary, each word is recorded as 1 if it appears, and it is recorded as 0 if it does not appear. In this way, the words are digitally marked in turn. For example "Mark likes to swim at the natatorium."; build the dictionary ("Mark": 1, "likes": 2, "to": 3, "swim": 4, "at": 5, "the": 6, "natatorium": 7); the word representation is

Mark: [1, 0, 0, 0, 0, 0, 0,]

Likes: [0, 1, 0, 0, 0, 0, 0,]

Swim: [0, 0, 1, 0, 0, 0, 0,]

By analogy, each word in the constructed dictionary is marked in this way, and the vector form of all words constitutes a text matrix. And this method only extracts a single word and cannot combine the semantic understanding to preserve the relationship between the upper and lower words.

The training steps of the bag-of-words model are similar to those of the one-hot discrete model. The difference between the two is that the bag-of-words model marks the frequency of word occurrences. This extraction method has the same disadvantages as the one-hot discrete model.

Tf-IDF model simply means that if a certain word appears more times in one of the texts, it will appear less times in the training data.

One is the neural network language model (NNLM) based on the N-gram grammar, and the other is the word vector Word2Vec semantic feature extraction model based on the neural network language model. NNLM is shown in Figure 3.

Figure 3 is a schematic diagram of the NNLM structure. From the sketch of the structure, as in the NNLM, a feature vector is first established for each word, and these feature vectors are combined into a high-dimensional vector matrix C. It sets the probability model, inputs a sequence of vectors, and finally calculates the joint probability of this sequence. n is the number label; the mathematical expression of the neural network model is

$$\begin{aligned} &f(\zeta(\delta),\zeta(\delta-1),\ldots,\zeta(\delta-n+2),\zeta(\delta-n+1)) \\ &\quad = p\left(\frac{\zeta(\delta)}{\zeta_1(\delta-1)}\right). \end{aligned} \tag{11}$$

Among them, $\zeta_1(\delta-1)$ represents the vector sequence from the first word to the δ word. The above expression model needs to meet the following conditions:

$$\begin{gathered} f(\zeta(\delta),\zeta(\delta-1),\ldots,\zeta(\delta-n+2),\zeta(\delta-n+1)) > 0, \\ \sum_{1}{}_{v=1}^{Q} f(v,\zeta(\delta),w(\delta-1),\ldots,\zeta(\delta-n+2),\zeta(\delta-n+1)) = 1. \end{gathered} \tag{12}$$

The input vector sequence is converted into a probability distribution through a feedforward or recurrent neural network, and the probability is calculated as

$$\begin{aligned} &f(v,\zeta(\delta),\zeta(\delta-1),\ldots,\zeta(\delta-n+2),\zeta(\delta-n+1)) \\ &\quad = g(v,C(\zeta(\delta-n+1)),\ldots,C(\zeta(\delta-1)), \end{aligned} \tag{13}$$

where $g(x)$ represents a feedforward or recurrent neural network. After the model training is completed, the weight parameters and word vectors of the network are obtained. The model solves the problem of text representation and the probability distribution of word vectors.

Another semantic feature extraction model is Word2Vec updated under the neural network language model. The difference between it and the neural network language model is that Word2Vec directly connects the word vector to an embedding layer, regardless of the contextual relationship, but uses the next word of the current word as the contextual information. The Word2Vec model mainly includes two models, CBoW and Skip-gram. The Word2Vec model is shown in Figure 4.

Figure 4 is the structure diagram of Word2Vec model. As can be seen from the structure diagram, CBoW judges the semantic features of the current word according to the semantic relationship between the training texts, while Skpp-gram judges the relationship between the antecedents and

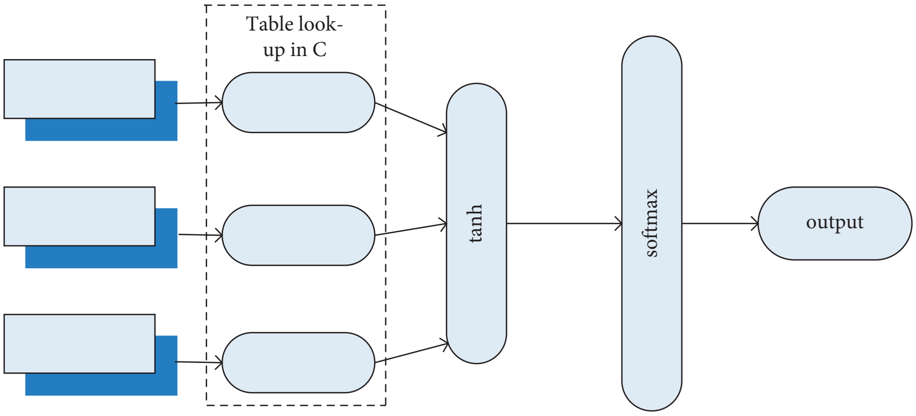

Figure 3: Simplified NNLM.

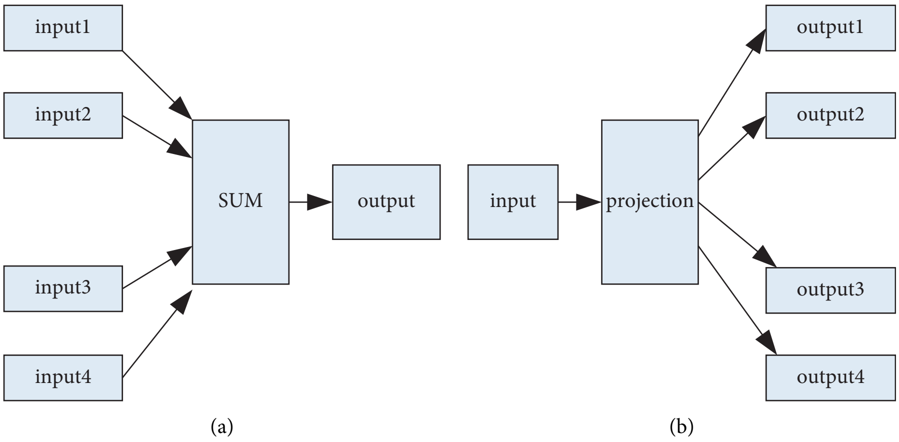

Figure 4: Word2Vec model diagram. (a) CBoW and (b) Skip-gram.

contexts based on the semantic characteristics of the current word. Therefore, the Word2Vec model can perform feature extraction in combination with the context of the context.

2.2. CNN-Based English Reading Corpus Text Feature Extraction Model. CNN first inputs the original features from the input layer for convolution processing, and after the convolution processing, it becomes a feature map through the calculation of the next layer to continue processing. Then the feature map is weighted and then biased, and finally the output is processed by the activation function of the output layer [20]. The CNN structure is shown in Figure 5.

Figure 5 is a schematic diagram of the CNN structure. From the structure diagram, CNN selects features through convolution operation. The expression of the convolution layer is as follows:

$$c_j^i = f\left(\sum_{x\in M_j} c_x^{i-1} r_{xj}^i + s_j^i\right). \tag{14}$$

Among them, i represents the number of layers of the convolutional layer, s represents the convolution kernel, j represents the previous layer, M represents the feature, r is the bias, and f represents the activation function.

The training process of the convolutional layer of CNN is shown in Figure 6.

Figure 6 shows the training process of convolution kernel. Based on the structural processes shown, the feature extraction is performed by the convolution kernel.

Based on the language features of English, attributes are assigned to the meaning of each word in different sentence contexts, and a semantic model of meaning attributes is constructed according to the principle of feature extraction. Each meaning has a corresponding attribute word vector. During the training process, the words in the training sample are extracted and the meaning is estimated in the model according to the semantic model. Assuming that the word w belongs to the meaning h_i, the calculation formula of the probability that the word w corresponds to the meaning h_i is $p(w|h_i)$:

$$p(w|h_\mathrm{i}) = \frac{\mathrm{times}(w, h_i)}{\mathrm{times}(w, h)}. \tag{15}$$

Among them, $\mathrm{times}(w, h_i)$ represents the frequency of occurrence of the word and its meaning in the training set; $\mathrm{times}(w, h)$ represents the sum of the frequency of occurrence of the word and all its meanings in the training set.

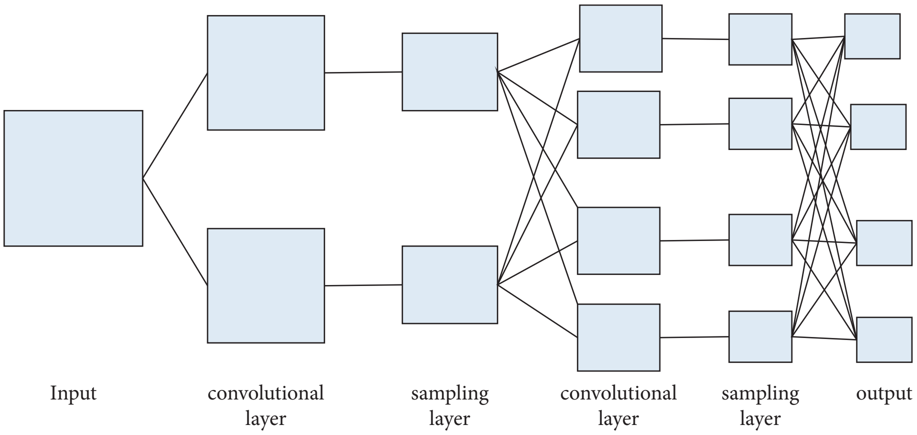

FIGURE 5: CNN structure diagram.

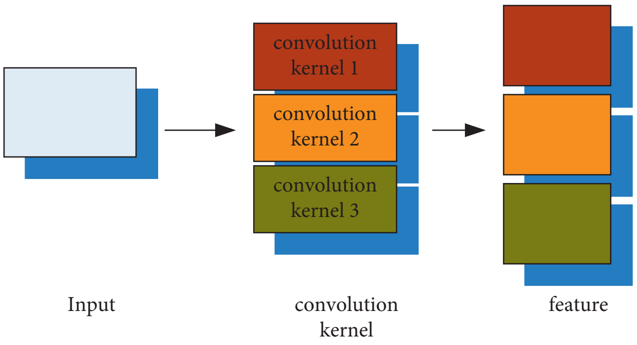

FIGURE 6: Convolutional layer training process.

Combined with relevant theoretical knowledge, the constructed text feature extraction model is shown in Figure 7.

Figure 7 is a structural diagram of a text feature extraction model based on CNN. According to the structure of Figure 7, after the data is processed by the convolution layer and the pooling layer, it is processed by the connection layer, and finally the features are output.

Assuming that the text is represented as Q, which contains n words, w represents words, and z represents word vectors, the hidden layer state function is

$$H = W\left(z_w + z_h\right) + \varepsilon, \tag{16}$$

where W represents the weight matrix and ε is the decoding bias vector.

In the decoder part, the updated hidden state is obtained from the previous hidden state and the current input:

$$\widehat{H} = CNN\left(\left[Y_{\mathrm{i}}; H\right], I\right), \tag{17}$$

where Y is the output vector of decoder, i is the number of layers reached by training, and I represents the current input.

Using K to represent the bias vector, the probability distribution of the final output through the hidden state is

$$p = \operatorname{softmax}\left(W\widehat{H} + K\right). \tag{18}$$

The meaning category Y of the final output is

$$Y = \widehat{W}\widehat{H} + \widehat{\varepsilon}. \tag{19}$$

Among them, $\widehat{H}$ represents the updated feature after the pooling layer, and $\widehat{W}$ represents the updated weight matrix.

3. Experiments and Results

3.1. CNN-Based English Reading Corpus Text Feature Classification Experiment. This paper selects the content of the National English Major Level 8 reading comprehension section from 1999 to 2014 to build a self-built English reading corpus. The experiment is divided into test data and training data, and the specific number is shown in Table 2.

Table 2 shows the distribution of experimental data sets, dividing the self-built English reading corpus into three types: words, parts of speech, and sentences. The parts of speech are divided into nouns, verbs, adjectives, and adverbs. The specific proportions are shown in Figure 8.

According to the data shown in Figure 8, the nouns in the special eight reading sections have a higher proportion, which is also due to the large number of nouns in the English sentence structure.

The experiment will use the data sets in the above table and figure as the object to evaluate the effect of the training model constructed above.

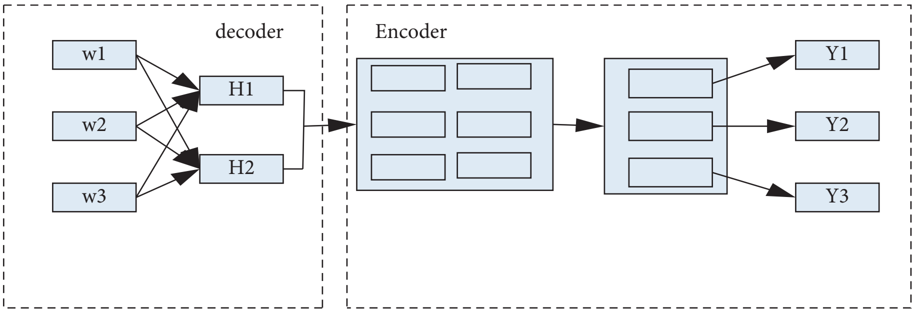

Figure 7: Model structure of text feature extraction based on CNN.

Table 2: Experimental dataset allocation table.

–	Test set	Training set
Number of words	2361	5512
Number of sentences	646	1510
Number of word attribute	12500	29167

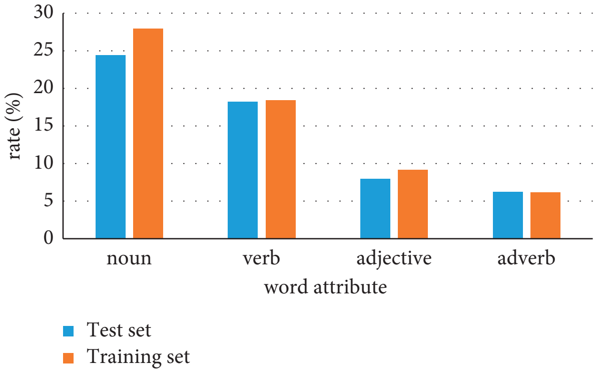

Figure 8: Proportion of part-of-speech dataset.

3.2. Model Optimization. The dropout layer is added to the model constructed in this paper based on CNN, and a loss function is added to the training. The binary cross entropy loss function is used to optimize the model.

The optimized CNN feature classification model is shown in Figure 9.

Figure 9 shows the optimized text feature extraction model. It can be seen from the model structure diagram that the optimized model only adds a dropout layer on the original basis.

In this paper, the shape of the convolution kernel is updated in the experiment, which are 2width, 3width, 4width, 5width, and 6width, respectively, so as to consider the semantic features of two connected words, three words, and so on until 6 words. Three ways are combined into a group, and different combinations of convolution kernel experiments are designed. The specific allocation is shown in Table 3.

Table 3 shows the three convolution kernel combination methods, and the experiment will select three convolution kernel combination methods for experimental verification. In the experiment, three combinations of convolution kernels will be selected for experimental verification.

4. Results

The first experiment in this paper is a text semantic feature extraction experiment, using the TF-IDF, NNLM, and Word2Vec mentioned above as a comparison experiment. The experiment uses the previous experimental data set and part-of-speech data set for experiments, and the experimental results are analyzed using the binary classification mixture matrix data mentioned above. For the training set, the final results are shown in Figures 10 and 11.

Figures 10 and 11 are comparison charts of precision-recall rates for processing experimental datasets and part-of-speech datasets using TF-IDF, NNLM, Word2Vec, and the feature extraction model constructed based on CNN in this paper. From the data in the figure, the feature extraction model based on CNN has the best performance. From the data in the two figures, it can be seen that the traditional method performs slightly better when dealing with the experimental dataset. The reason is that the experimental dataset is a long text, while the part-of-speech dataset is a single word, and the semantic features of a single word are not as obvious as long texts. The traditional feature extraction method model is relatively simple, and it is difficult to extract semantic information through meaning, and the new model constructed in this paper takes into account the different meaning attributes of words. Therefore, whether it is processing the experimental data set or the part-of-speech data set, the effect is better.

The second experiment is to analyze the classification effect of feature extraction. According to the three types of methods mentioned above, SVM, conditional random field, and Naive Bayes, comparative experiments are carried out. The experimental results are analyzed using the binary classification mixture matrix data mentioned above. The training set data obtained are shown in Figure 12.

Figure 12 shows the classification results of the experimental data set and the part-of-speech data set by three classification methods. From the data in the figure, the classification effect of NBM is the best. The reason may be that the classification of NBM is based on the assumption of conditional independence, so it has certain advantages in this actual text classification.

The accuracy of the verbs, nouns, adjectives, and adverbs in the test set and training set is the test index, and the classification effect is shown in Figure 13.

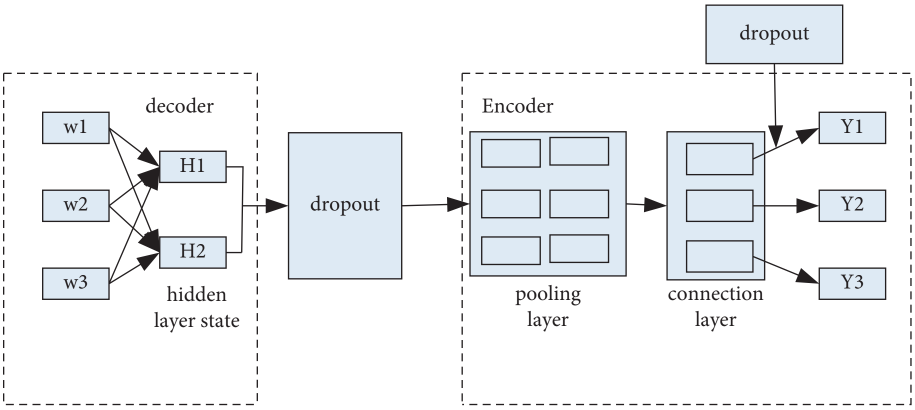

Figure 9: CNN-based text feature extraction optimization model structure.

Table 3: Combination of convolution kernels.

Combination	Convolution kernel shape (width)		
Combination 1	2	3	4
Combination 2	5	4	3
Combination 3	4	5	6

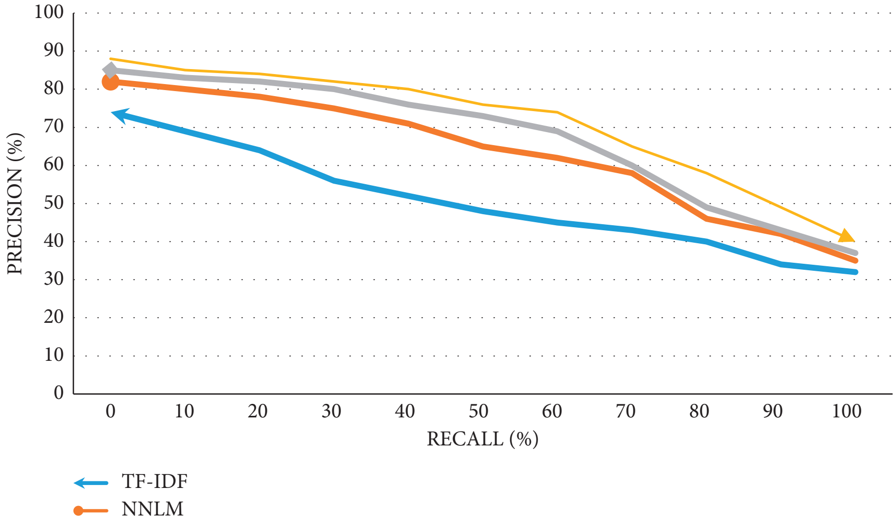

Figure 10: Precision-recall comparison plot for the experimental dataset.

Figure 13 shows the classification results of gerunds, adverbs, and adjectives in the test set and training set. From the data in the figure, the classification accuracy of verbs is the highest. This paper believes that the main reason is that verbs have obvious meanings in sentences, so they can be better identified in classification.

The third experiment in this paper is a model optimization experiment, which is first tested according to the three convolution kernel methods in Table 3.

Tables 4 and 5 are the test results of the convolution kernel. From the test results, when the shape of the convolution kernel is 2, 3, and 4, the classification accuracy of the test set is the highest, which is consistent with the real language features. When the number of words is small, it is easier to obtain the correlation between the two.

The results of the dropout experiment are shown in Figure 14.

Figure 14 shows the effect of using different dropout at different layers on the model. From the data in the figure, when the dropout rate of the convolutional layer is about 10%, the training effect is the best; when the dropout rate of the connection layer is about 50%, the training effect is the best. Therefore, the optimization model in this paper will set these two values in the corresponding positions.

Then it is necessary to adjust the number of iterations to observe the experimental effect. The results are shown in Figure 15.

Figure 15 is a comparison chart of the effects of the experiments when the number of iterations is different. From the results in the figure, when the number of iterations

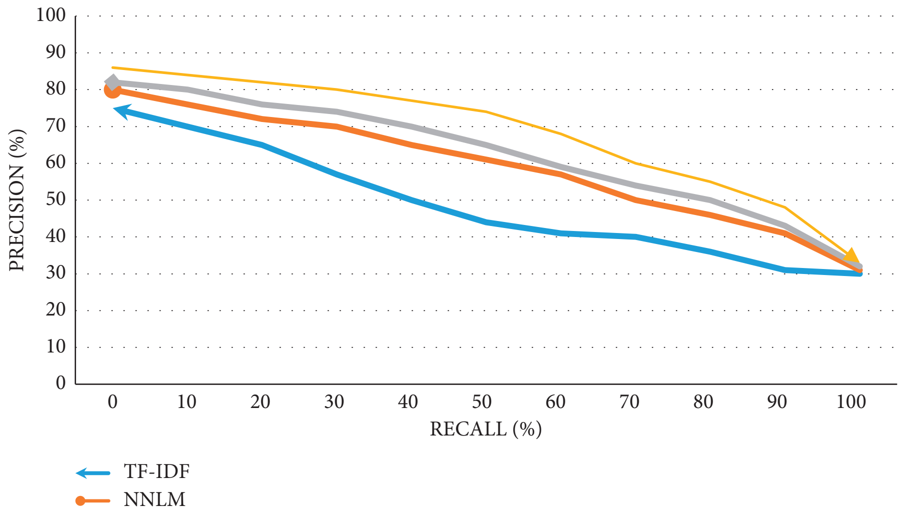

FIGURE 11: Precision-recall comparison plot for part-of-speech datasets.

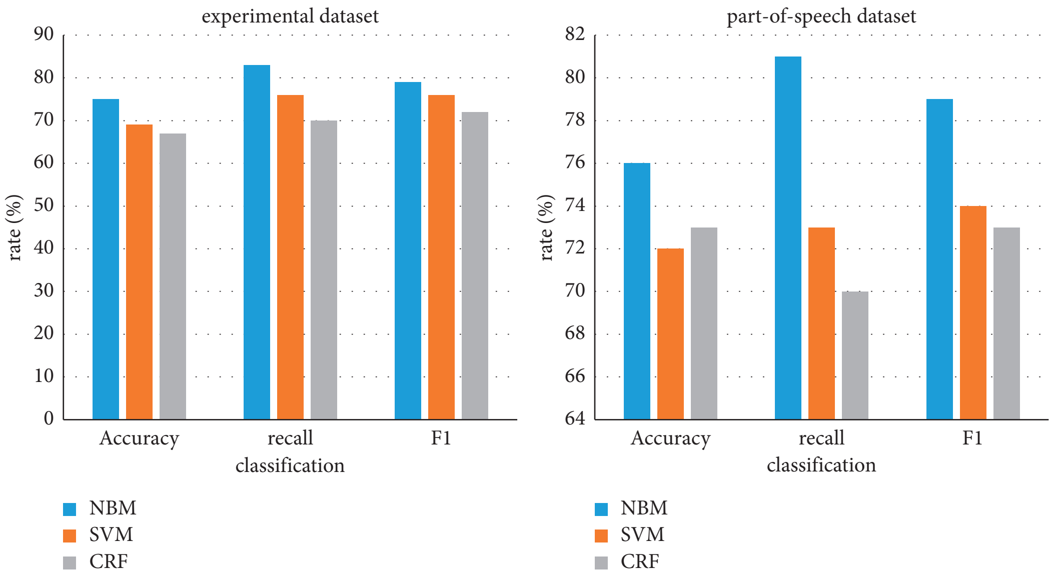

FIGURE 12: Comparison of feature classification methods.

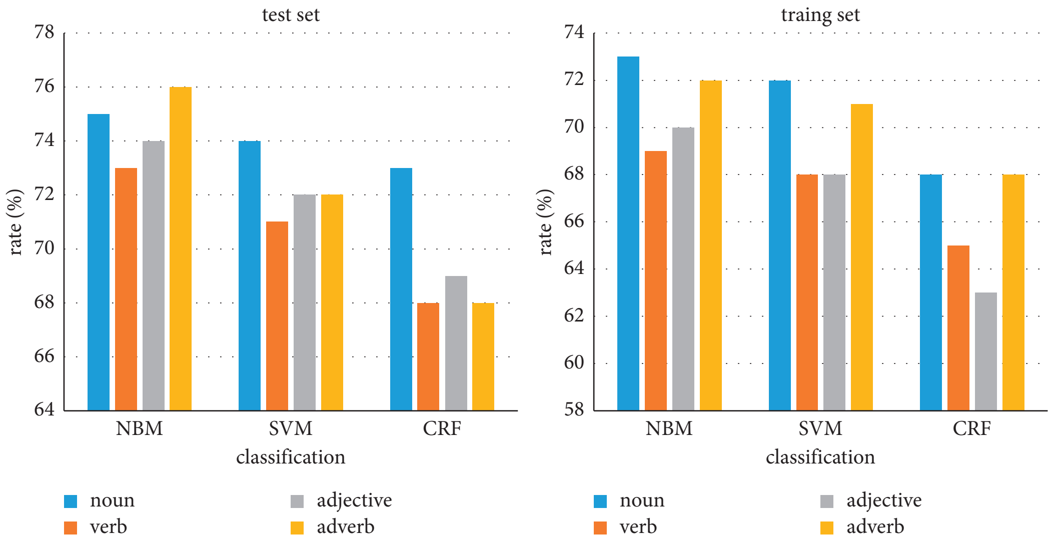

FIGURE 13: Part-of-speech specific classification results.

Table 4: Experimental dataset convolution kernel test results.

Convolution kernel shape	Test set accuracy	Training set accuracy	*F*1
2,3,4	0.9864	0.8975	0.9135
5,4,3	0.9783	0.8695	0.9127
4,5,6	0.9932	0.8896	0.9123

Table 5: Part-of-speech dataset convolution kernel test results.

Convolution kernel shape	Test set accuracy	Training set accuracy	*F*1
2,3,4	0.9875	0.9014	0.9029
5,4,3	0.9817	0.8765	0.9056
4,5,6	0.9897	0.8895	0.9104

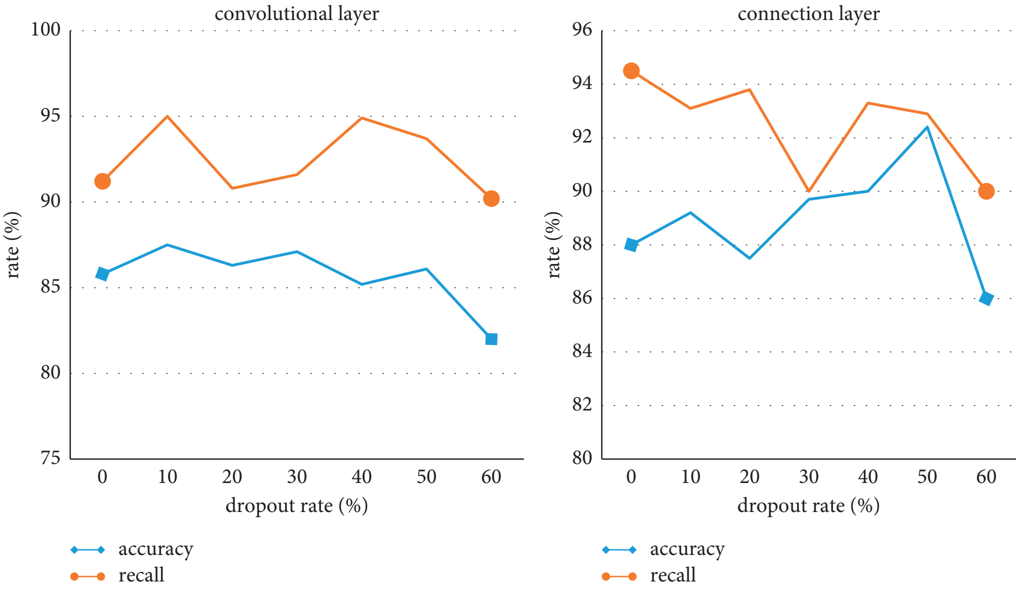

Figure 14: The effect of dropout with different convolutional and connected layers.

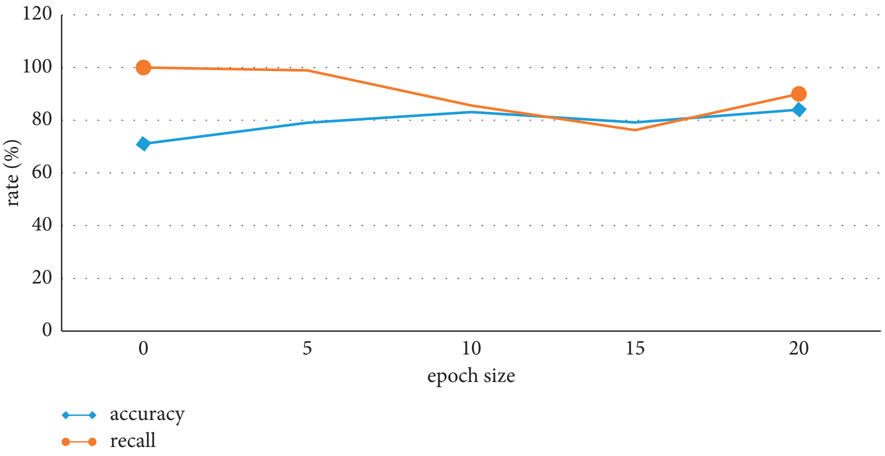

Figure 15: Effects at different iterations.

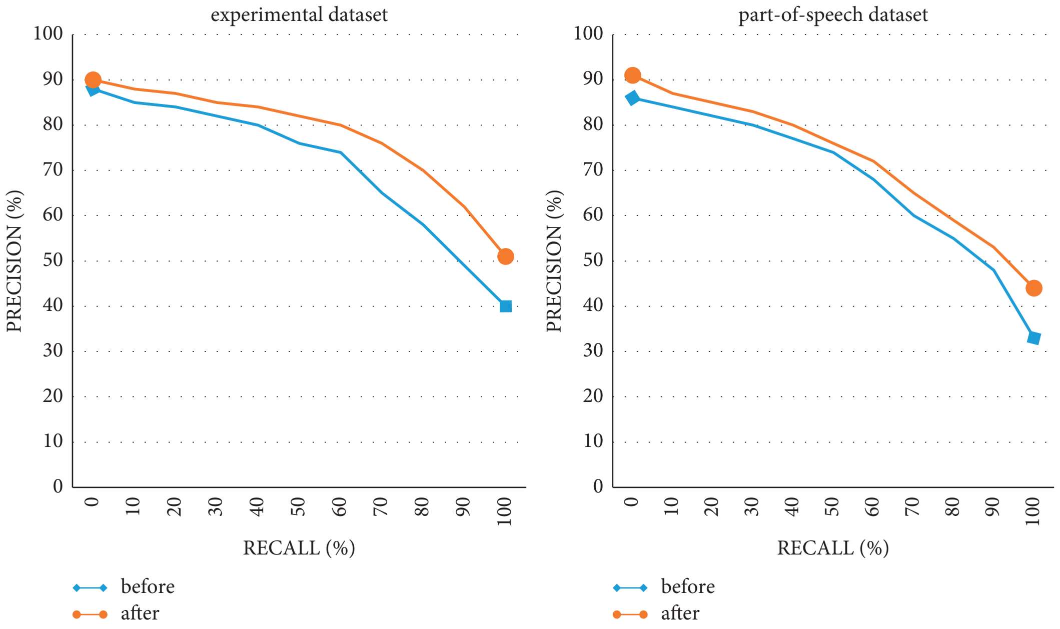

Figure 16: Precision-recall comparison before and after model optimization.

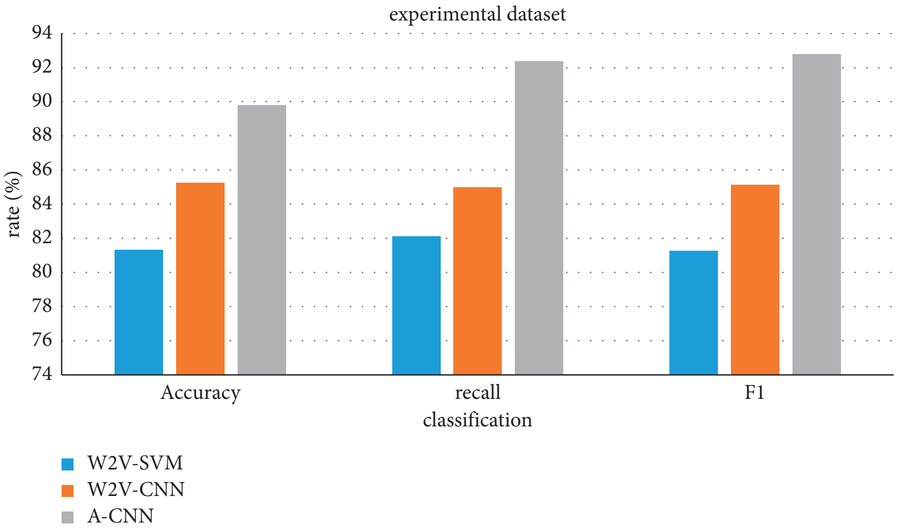

Figure 17: Effect comparison of different models.

is about 20, the comprehensive index of the model is the best, so the variable of the optimization model in this paper will be set to 20 in the training process.

According to the optimized model, experiments are carried out on the experimental data set and part-of-speech data set and compared with the experimental results before optimization. The results are shown in Figure 16.

Figure 16 is a comparison chart of the precision-recall rate of the experimental dataset and the part-of-speech dataset before and after optimization of the CNN-based feature extraction model. Figure 16 clearly shows that the feature extraction effect after optimization is more accurate, which is partly because the optimized model optimizes the loss function. And this experiment added dropout to extract more refined data.

To better analyze the results, combined experiments were next performed. The W2V-SVM group used Word2-Vec to transform text words into word vectors and then extracted features through SVM. The W2V-CNN group uses Word2Vec to convert text vocabulary into word vectors and then uses the constructed CNN model for feature classification. Compared with the optimized CNN (A-CNN) feature extraction experiment, the experimental results are shown in Figure 17.

Figure 17 shows the effects of different models on feature extraction and classification experiments. From the results in the figure, the classification effect after optimization is better than that of other combination methods set in this paper. The main reason is that the optimized model actually combines the advantages of various models and performs word vector processing on the meaning of words. And it optimizes the number of iterations of CNN, which can improve the training effect of the model to a certain extent. In addition, the dropout layer added by the optimized CNN was also tested, and the optimal solution of the dropout rate in the model was obtained. Therefore, the training effect of the optimized model is naturally stronger than that of other models.

5. Conclusions

By discussing the importance of English reading ability in today's society, this paper proposes a numerical analysis of the English reading corpus and constructs a feature classification model based on the CNN model structure and English semantic features to analyze the self-built English corpus. The classification effects of TF-IDF, NNLM, Word2Vec, and the model constructed in this paper are compared and analyzed based on the experimental indicators of precision-recall rate; the experimental results show that the feature extraction classification model constructed in this paper has the best effect. Then compare the commonly used text classification methods NBM, SVM, and CRF, and get the best classification effect of NBM. Then, the constructed feature classification model is optimized to obtain a more accurate classification effect of the optimized model. Finally, a combination of several models is designed to analyze the feature classification effect; according to the classification accuracy, recall and F1 value of the optimized CNN model are higher than other models, which proves the availability of the optimized model. All experiments show the importance of word vector representation for different meanings of words in the process of text semantic feature extraction.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] N. Yanez-Bouza and V. Gonzalez-Diaz, "'He liked to read, write, and whatch televishon'-The APU Writing and Reading Corpus (1979-1988)," *Literary and Linguistic Computing*, vol. 34, no. 2, pp. 449–469, 2019.

[2] H. Lee, M. Warschauer, and J. H. Lee, "Toward the establishment of a data-driven learning model: role of learner factors in corpus-based second language vocabulary learning," *The Modern Language Journal*, vol. 104, no. 2, pp. 345–362, 2020.

[3] I. Shatz, "How native language and L2 proficiency affect EFL learners' capitalisation abilities: a large-scale corpus study," *Corpora*, vol. 14, no. 2, pp. 173–202, 2019.

[4] T. Guziurová, "Discourse reflexivity IN written academic English as lingua franca: code glosses IN research articles," *Discourse and Interaction*, vol. 13, no. 2, pp. 36–54, 2020.

[5] A. N. Oveshkova, "Work with English corpora as a means of promoting learner autonomy," *The Education and science journal*, vol. 20, no. 8, pp. 66–87, 2018.

[6] J. Ryu, M. Jeon, and M. Jeon, "An analysis of text difficulty across grades in Korean middle school English textbooks using coh-metrix," *The Journal of AsiaTEFL*, vol. 17, no. 3, pp. 921–936, 2020.

[7] I. Evo and A. Avramovi, "Convolutional neural network based automatic object detection on aerial images," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 5, pp. 740–744, 2017.

[8] K. B. Lee, S. Cheon, and C. O. Kim, "A convolutional neural network for fault classification and diagnosis in semiconductor manufacturing processes," *IEEE Transactions on Semiconductor Manufacturing*, vol. 30, no. 2, pp. 135–142, 2017.

[9] F.-C. Chen and M. R. Jahanshahi, "NB-CNN: deep learning-based crack detection using convolutional neural network and naïve Bayes data fusion," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 5, pp. 4392–4400, 2018.

[10] F. Palsson, J. R. Sveinsson, and M. O. Ulfarsson, "Multispectral and hyperspectral image fusion using a 3-D-convolutional neural network," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 5, pp. 639–643, 2017.

[11] P. C. U. Murillo, J. Arenas, and R. J. Moreno, "Implementation of a data augmentation algorithm validated by means of the accuracy of a convolutional neural network," *Journal of Engineering and Applied Sciences*, vol. 12, no. 20, pp. 5323–5331, 2017.

[12] A. El-Sawy, M. Loey, and H. M. El-Bakry, "Arabic handwritten characters recognition using convolutional neural network," *WSEAS Transactions on Computer Research*, vol. 5, no. 1, pp. 11–19, 2017.

[13] J. Atkinson-Abutridy, C. Mellish, and S. Aitken, "Combining information extraction with genetic algorithms for text mining," *IEEE Intelligent Systems*, vol. 19, no. 3, pp. 22–30, 2004.

[14] S. A. Salloum, M. Al-Emran, A. A. Monem, and K. Shaalan, "A survey of text mining in social media: facebook and twitter perspectives," *Advances in Science, Technology and Engineering Systems Journal*, vol. 2, no. 1, pp. 127–133, 2017.

[15] N. Sharma and Nidhi, "Text extraction and recognition from the normal images using MSER feature extraction and text segmentation methods," *Indian Journal of Science and Technology*, vol. 10, no. 17, pp. 1–12, 2017.

[16] M. García, S. Maldonado, and C. Vairetti, "Efficient n-gram construction for text categorization using feature selection techniques," *Intelligent Data Analysis*, vol. 25, no. 3, pp. 509–525, 2021.

[17] E. F. Ayetiran, "An index-based joint multilingual/cross-lingual text categorization using topic expansion via BabelNet," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 28, no. 1, pp. 224–237, 2020.

[18] A. Rozaimee, M. Bashir, and M. Wan, "Automatic Hausa LanguageText summarization based on feature extraction using nave Bayes model," *World Applied Sciences Journal*, vol. 35, no. 9, pp. 2074–2080, 2017.

[19] S. Sahoo, B. Kanungo, S. Behera, and S. Sabut, "Multiresolution wavelet transform based feature extraction and ECG classification to detect cardiac abnormalities," *Measurement*, vol. 108, no. 108, pp. 55–66, 2017.

[20] C. A. M. Strfer, J. Wu, H. Xiao, and E. Paterson, "Data-driven, physics-based feature extraction from fluid flow fields," *Communications in Computational Physics*, vol. 25, no. 3, pp. 625–650, 2018.