Hindawi

Research Article

Identification of Grammatical Errors of English Language Based on Intelligent Translational Model

Hongli Chen (ID)

School of International Business and Management, Shanghai Sipo Polytechnic, Shanghai 201300, China

Correspondence should be addressed to Hongli Chen; 201901125214@stu.zjsru.edu.cn

To address the problem of low kappa, precision and recall values, and high misjudgment rate in traditional methods, this study proposes an English grammatical error identification method based on a machine translation model. For this purpose, a bi-directional long short-term memory (Bi-LSTM) model is established to diagnose English grammatical errors. A machine learning (ML) model, i.e., Naive Bayes is used for the result classification of the English grammatical error diagnosis, and the N-gram model is utilized to effectively point out the location of the error. According to the preprocessing results, a grammatical error generation model is designed, a parallel corpus is built from which a training dataset for the model training is generated, and different types of grammatical errors are also checked. The overall architecture of the machine translation model is given, and the model parameters are trained on a large-scale modification of the wrong learner corpus, which greatly improves the accuracy of grammatical error identification. The experimental outcomes reveal that the model used in this study significantly improves the kappa value, the precision and recall values, and the misjudgment rate remains below 1.0, which clearly demonstrates that the detection effect is superior.

1. Introduction

In the field of English teaching and testing, grammatical error detection is an important branch of natural language processing (NLP) and is also an important indicator of detecting the language ability of English learners. In simple terms, the task of grammatical error detection is to utilize a computer for the identification of grammatical errors along with their location and to classify or correct these errors [1]. It has a wide range of applications, including automated correction of language learner mistakes, content proofreading, and grammatical correction [2]. At present, the detection of English grammar errors usually relies on manual review by teachers or graders. This process requires a lot of manpower and other resources, and it is a challenging task to assure the test results' reliability and validity. To overcome the abovementioned drawbacks, scholars at home and abroad have begun to use the power of NLP in recent years to use computers for the automatic evaluation of English application quality. Among them, grammatical error

detection is an important part of English quality evaluation, which can provide learners with written error correction feedback and improve students' autonomous learning awareness.

With the rapid development of artificial intelligence (AI) technology, people are progressively gaining the ability to utilize machines to detect grammatical faults in English. Scholars at home and abroad have already had many successful research results, which have been applied to the actual English examination papers. For instance, Fu et al. [3] proposed a grammar detection algorithm for text information hiding. Through the investigation and structural analysis of a large number of sentences, the rewriting template with keywords as the core was extracted to rewrite the sentences with certain structural characteristics, but the language consistency of sentences may be damaged after synonym replacement. Because of this challenge, a grammar detection algorithm is proposed. Firstly, the optional collocations of part of speech are counted according to the grammar library to judge whether the detection of part of

speech collocation is reasonable, then the collocation of word attributes is detected, and finally whether to detect the word itself is determined. Numerical experiments on the C platform illustrate that the identification model can detect syntax errors in an effective way. Tan et al. [4] proposed a corpus-based technique for checking and correcting grammatical errors in English articles, built a corpus, and utilized the finite fallback algorithm to check and correct grammatical faults in the corpus. The simulation outcomes depict that this approach is promising in the error detection of articles and nouns. In addition, some scholars have proposed grammar error checking methods in automatic text proofreading. The grammar errors of text are divided into collocation errors and errors related to sentence pattern components. Pattern matching methods and sentence pattern component analysis are used to check them, respectively. The combination of these two methods can consider local and global grammar restriction information, and at the same time, it also reduces the complexity of syntax checking. Through the analysis and evaluation of the experimental results, it is proved that the method is feasible.

Although the above methods realize the detection of grammatical errors and reduce the workload for teachers and other relevant personnel. At present, English grammar detection is not only oriented to traditional paper homework but also involves a large number of network homework, which leads to the increase of detection workload. In this context, the abovementioned traditions cannot effectively detect English grammar accurately because there are problems of low kappa value, precision and recall value, and high misjudgment rate. Therefore, this paper proposes an English grammar error detection method based on a machine translation model. This work utilizes the Bi-LSTM model for English grammatical error diagnoses, the Naive Bayes algorithm for their results classification, and the N-gram method for finding the exact location of the error. The primary aim of the proposed method is to properly detect collocation problems, word errors, and writing faults in the English homework of students.

The remaining structure of the paper is laid down as follows: Section 2 describes the grammatical error preprocessing. This section further explained the Bi-LSTM-based English grammar error diagnosis, classification of English grammatical errors based on the Naive Bayes algorithm, and English grammar error detection method based on the machine-translation model. Section 3 is about the English grammar error detection method based on the machine-translation model. It further describes the syntax error generation model, model training dataset generation, different types of syntax error checking, implementation of English grammar error detection based on the machine-translation model, and English grammar error detection process. The simulation results and analysis are presented in Section 4. Finally, Section 5 summarizes the overall theme of the paper.

2. Grammatical Error Preprocessing

The goal of designing an English grammatical error detection method is to accurately detect collocation errors, word errors, and writing errors in the learners' English homework. Learners can use this method to check their own English grammatical errors and understand what they have committed. The exact type of error detection may help them correct the errors. With the help of the proposed method, learners will be able to know their own mistakes without the manual help of the teacher and dictionary to improve their English application level in a targeted manner.

2.1. English Grammar Error Diagnosis Based on the Bi-LSTM Model. In the diagnosis of English grammar errors, the traditional method only uses the neural network to process the error sentence information. For a specific word, it can only be inferred whether it is correct or not by using the information before it. But for this kind of contextually related grammatical error diagnosis problem, it is far from enough to use the forward information only, and it is necessary to combine the latter information to judge whether the grammar is correct [5]. Hence, this study proposes a Bi-LSTM neural network-based English grammar error diagnosis.

The Bi-LSTM neural network structure includes a fully connected layer, an encoding layer, a decoding layer, and an output layer. The decoding layer then takes the output of the encoding layer as an input, which is another fully connected layer with 1 output size [6]. The expression of the output layer is as follows:

$$O_a = \mu \left(D_a + D_e + D_f \right)^2, \tag{1}$$

$$g(r_i) = O_a \times F(z). \tag{2}$$

Among them, O_a represents the training set, D_a, D_e, and D_f represents different error types, $g(r_i)$ represents whether the word r_i has errors, and $F(z)$ represents the loss function. Compared with words, there are many more correct words in the sentences than the words containing errors, so the Bi-LSTM neural network structure always tends to mark 0, which means that if there is no balance, the sentence is wrong. Therefore, weights are assigned to the loss function [7], in order to rebalance the correct and incorrect labels. The regularized loss function is calculated via the following formula:

$$F(z) = \sum_{i,j=1}^{N} \left(\tau_{ij}^{a} + \eta_{ij}^{b} \right). \tag{3}$$

Among the variables, τ_{ij}^{a} represents the correct type, η_{ij}^{b} represents the wrong type, and a and b represents the positive coefficient and the negative coefficient, respectively.

Through the above steps, it can be judged whether there are three errors of D_a, D_e, and D_f in the input content. Then, the correct and the incorrect types are separated according to the continuous length of errors in the test phase. If the continuous length is 1, it is the correct type, and if the length is greater than 1, it is the wrong type, thereby realizing English grammar error diagnosis.

2.2. Classification of English Grammatical Errors Based on the Naive Bayes Algorithm. The ML model, i.e., Naive Bayes [8], is a classification model based on the Bayesian model and the independence assumption of feature conditions. This Bayesian classification model has a simple concept, but it is difficult to calculate the posterior probability, so the following independence assumptions are introduced.

For the scenario of a given category K, all the features are independent of one another, as shown in the following equation:

$$D_{\min} = \sum_{i,j=1}^{N} (y_i - y_j)(y_i - y_j)^T. \tag{4}$$

Among the variables, $D_{\min}$ represents the minimum value of the error type, y_i represents a grammatical error in a long sentence, y_j represents a local grammatical error, and T represents the window size.

Based on this assumption, the Naive Bayes classification algorithm is obtained, as shown in the following equation:

$$D_{ij}^{'} = \sum_{i,j=1}^{N} A^T y_i - \sum_{i,j=1}^{N} A^T y_j, \tag{5}$$

where A^T represents the attribute of the syntax error type.

The English grammatical error classification method based on the Naive Bayes model usually focuses on three components: first, it calculates the conditional probability $P(A^T|K)$ of each attribute A^T under the category attribute K, and then calculates the probability of each attribute A^T, as the value is a constant term, so, the ϑ, which is a normalization factor can be used in the place of constant. Finally, it calculates the probability of each category based on the dataset. It calculates the posterior probability $P(K)$ of the sample for every category and takes the category with the maximum probability as the classification result.

The Naive Bayes method has numerous advantages: its performance is consistent, and it does not change much when classifying different datasets. The algorithm's logic is simple, and it takes less time and space.

2.3. English Grammatical Error Preprocessing Based on the N-Gram Model. The field of English grammatical error preprocessing includes a variety of commonly used models, such as the N-gram model, Markov model, and maximum entropy model. Among them, the most commonly used is the N-gram model, which can reflect the context relationship better than that of the rest. In principle, the larger the order of each fragment, the stronger its ability to reflect the context relationship. If the sparsity of the corpus is taken into consideration, the high order is not good. Therefore, in actual use, this article adopts the N-gram model for English grammatical error preprocessing.

The N-gram model is usually divided into two stages: the training stage and the inspection stage [9, 10]. According to the needs of the model, the corpus information is counted and saved. In the preprocessing stage, the information obtained from the input sentence is counted to determine

the grammatical error [11]. Under different circumstances, the way of using the N-gram model for grammar checking is also different. Generally, it can be divided into two types.

In the first step, the model estimates the chance of the binary grammar existing in the input sentence, yielding the probability of grammatical mistakes. The following is the calculating formula:

$$R_A = \frac{1}{N} \sum_{i=1}^{N} [x - x_i]^2, \tag{6}$$

where x represents global context information and x_i represents local context information.

Find all the binary grammars for the supplied sample text, and check and compute each binary grammar one by one in the second step. If there is a binary grammar calculation result that is less than the set threshold ∂, it is judged that there is an error here, and the prompt is returned. The threshold is expressed by the following formula:

$$\partial = \frac{p(x)p(x_i)}{P(K)}. \tag{7}$$

Both the first method and the second method can judge whether the grammar is correct or wrong. The second method can effectively point out the location of the error, but the first method cannot. The first method and the second method adopt the n-ary grammar model, so the corpus largely determines the effectiveness of the two methods. A more standardized corpus can improve the effect of grammatical error detection. As a result, corpus construction will be studied further.

3. English Grammar Error Identification Approach Based on the Machine Translation Model

The diagnosis and categorization of English grammatical errors, as well as the probability of grammatical errors, are accomplished using the English grammatical error preprocessing link. Based on this, the detection of English grammatical errors is carried out.

3.1. Syntax Error Generation Model. The size of the error correction parallel corpus has always been the biggest impediment to its efficiency when it comes to English grammatical errors. In order to obtain a more complete parallel sentence pair system based on the training of the positive grammatical error correction model, the correct text is used as the carrier and moderately added noise, so as to obtain incidental noise, that is, wrong text, to establish a pseudo-parallel corpus.

The English grammar error generation model is trained through the error correction parallel corpus, the correct sentences contained in the parallel sentence pairs are inputs, and the sentences with grammatical errors in the parallel sentence pairs are outputs. The reverse English grammar error generation model selects the same network structure and training rules as the forward English grammar error

correction model. Given that the wrong sentence at the source end is $Z = \{z_1, z_2, \ldots, z_n\}$ and the corrected sentence at the target end is $V = \{v_1, v_2, \ldots, v_m\}$, the probability of applying noise when modeling the reverse model is

$$\sigma(N) = \sqrt{\sum_{l=1}^{N} \|U_l\| \times p(z_i|v_i)}, \qquad (8)$$

where U_l represents the numerical distribution of the probability of the value of the noise function, z_i represents the noise function, and v_i represents the probability distribution.

The model loss function is given in the following equation:

$$Z(t) = \left(z_{ij}^t\right)_{m \times n} \times \sigma(N), \qquad (9)$$

where z_{ij}^t represents the probability of noise at the time t.

The learning goal is the likelihood of the maximum expected model in the training data, namely:

$$J(x) = \left(z_{ij} - H_x\right)^T \cdot W, \qquad (10)$$

where H_x represents the convergence of the maximum likelihood estimation during the model training process and W represents the likelihood weighting.

According to the grammatical error generation model, the text containing English grammatical mistakes may be created effortlessly by producing the right text, resulting in a pseudo-parallel corpus.

3.2. Model Training and Dataset Generation.

A model training dataset needs to be generated before building an English grammar error detection model in order to improve the accuracy of error detection. In order to track the performance of a particular model using a dataset, it is very important to divide the dataset into two parts, i.e., training part and testing part. The training part is used to train the model while the testing part is used to evaluate the performance of the model on untrained data. The purpose of the testing part is to test whether the trained model can perform well on the unseen data or not. Many datasets perform well on the training set, but when applied to other datasets, the performance is different due to the model overfitting. Therefore, it is necessary to use the test set to observe whether the model performs consistently on both datasets. The testing part should satisfy two conditions: the first condition is that the scale is large enough to produce mathematically efficient outcomes, and the second condition is to be able to represent the entire dataset. In other words, the features of the chosen testing part should match those of the training part. In the case of ensuring that the test set meets the above conditions, the ratio of the training set to the test set is usually about $8 : 2$, which can be adjusted according to the actual situation.

In this study, there are a total of 11069 sentences in the training set, of which 7,524 sentences have grammatical errors, and there are 8961 sentences in the test set, of which

2,854 sentences have grammatical errors. Adding grammatically correct sentences can improve the model detection results to a certain extent. Therefore, a higher percentage of correct sentences were added to the training set in this study. Table 1 shows the statistical results of the model training dataset.

In order for the test set to properly evaluate the model's effect, the proportions of the training set and the test set in the distribution of different categories of grammatical errors are essentially the same, as shown in Table 2.

According to Table 2, in the training set, there is a total of 7524 grammatical errors, 1532 missing content, 2046 sequence errors, 2749 tense inconsistencies, and 1197 fixed collocation errors. Similarly, in the test set, there are a total of 2854 grammatical errors. The errors are divided into 503 missing content, 258 wrong sequences, 981 tense inconsistency, and 1112 fixed collocation errors.

3.3. Different Types of Syntax Error Checking.

If there are spelling errors, word errors, collocation errors, and tense errors in English sentences, it will have a great impact on the subsequent grammar check. This article uses a combination of rules and statistics for the spelling check of words. Word error η_i gives all candidate sets whose edit distance is less than 2, denoted by $E(\eta_i)$, and then query the most frequently occurring candidate word in the corpus as the corrected word. In order to find the candidates more accurately, three combinations of candidate words are used, as shown in formulas (11)–(13):

$$E(\eta_i) = \left\|\eta(i) - \eta_{(i)}^n\right\|^2, \qquad (11)$$

$$\eta_i = \eta_m^2(i) + \|\eta(i) - \eta(i+j)\|^2, \qquad (12)$$

$$\eta_m(i) = \frac{\left(\eta_{m+1}(i) - \eta_{m-1}(i)\right)}{\eta_m(i)}. \qquad (13)$$

Among them, i represents the position of the given word, η_i represents the given word, $E(\eta_i)$ represents the candidate set η_i, and $\eta_{m+1}(i)$ represents the candidate word with the largest sum of the two-tuples and triples in the corpus among all the candidates.

Since there are many types of words in English grammar, such as nouns, verbs, prepositions, and conjunctions, this article selects nouns and verbs as the key research objects and studies the methods for checking errors of these two types of words. The following are the specific method design steps.

3.3.1. Noun Check Module.

The noun check module is mainly for the use of wrong singular and plural nouns. Due to the inconsistency of singular and plural nouns, a list of singular and plural nouns is first established in the noun check. The expression forms of some nouns are shown in Table 3.

The specific inspection process of the noun module is described as follows:

TABLE 1: Statistical results of the model training dataset.

Type	Number of sentences/piece	Number of grammatically correct	Number of grammatical errors/piece
Training part	11069	3545	7524
Testing part	8961	6107	2854

TABLE 2: Classification of error types in training and testing parts.

Type	Content is missing	Wrong order	Inconsistent tense	Fixed collocation errors
Training part	1532	2046	2749	1197
Testing part	503	258	981	1112

TABLE 3: Noun checklist.

Noun prototype	Plural noun	Type tag
Book	Books	1
Ruler	Rulers	1
Bus	Buses	2
Dish	Dishes	2
Story	Stories	3
Baby	Babies	3
Thief	Thieves	4
. . .	. . .	. . .

TABLE 4: Verb checklist.

Verb original form	Verb tense	Type tag
Work	Works	1
Get	Gets	1
Go	Goes	2
Wash	Washes	2
Study	Studies	3
Try	Tries	3
Have	Has	4
. . .	. . .	. . .

(1) First of all, the relevant pluralization rules are determined based on the findings of part of speech tagging

(2) By querying the noun checklist if it is marked as 1, add "*s*" after the word directly; if it is marked as 2, add "es" after the word, and if it is marked as 3 or 4, it will be converted according to the change rule

(3) Using the finite backoff algorithm [12], according to the frequency and ratio of the singular and plural nouns in the corpus, it is judged whether the correction should be completed

3.3.2. Verb Check Module. The verb check module is primarily concerned with verb usage, including verb form, tense, and subject-predicate agreement. Due to the complex and diverse forms of verbs, a verb checklist is established [13], as shown in Table 4.

The specific checking process of the verb checking module is described as follows:

(1) Find out the type of the labeled word according to the results of part of speech tagging

(2) Find the inflection form of the word in the verb checklist one by one

(3) Use the limited backoff algorithm to check and correct errors [14]

3.4. Implementation of English Grammar Error Identification Based on the Machine Translation Model

3.4.1. Complete Design of Machine Translation Model. The underlying concept of building machine translation models and statistical machine translation models is identical. They both construct complicated translation models using a large-scale parallel corpus and convert source language utterances into target language sentences using this translation model [15]. Figure 1 is a schematic diagram of the machine translation model.

The machine translation model, like the standard neural machine translation model, has two parts: an encoder and a decoder. The biggest difference is that the machine translation model completely abandons the traditional neural network architecture and is completely based on the self-attention mechanism of encoding and decoding [16]. This model framework can effectively solve the problem of long-distance dependence and can be processed in parallel, which is much faster than neural networks [17].

(1) Encoder: The encoder is composed of Y identical layers. Among them, each layer contains two different sublayers, the first sublayer is a multichannel attention network, and the second sublayer is a simple fully connected network. Each sublayer in the encoder is connected by residual connection and standard normalization.

(2) Decoder: The decoder also consists of Y identical layers. Unlike the encoder, in addition to having the same two sublayers as the encoder, the decoder also has a third sublayer, which passes the final result of the encoder to the multichannel attention mechanism sublayer. Similar to the encoder, each sublayer of the decoder is also connected by residual connection and standard normalization. In order to prevent words in the target language from using foreign output word information, the decoder adds a mask to each multichannel attention sublayer. Specifically, because the generation of the first word cannot refer to the generation result of the second
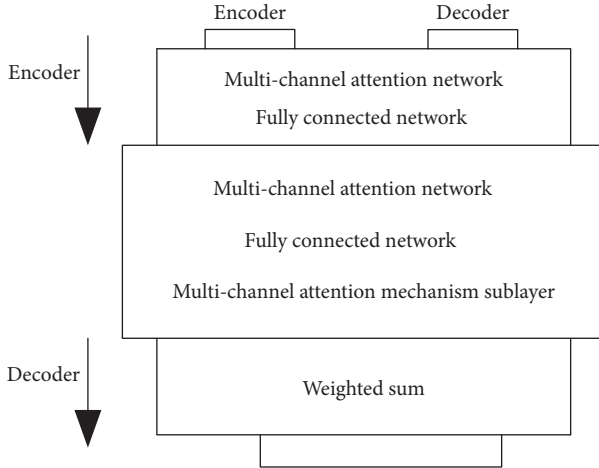
FIGURE 1: Schematic diagram of machine translation model.

word, the mask will change this information to 0 to ensure the accuracy of the detection result.

When dealing with various attention mechanisms, the general attention mechanism first creates a mapping function to obtain information relating to the provided query from the supplied key-value combination, and then takes the weighted sum of the values as the output. The machine translation model employs a particular attention technique known as scaling dot product, which efficiently solves the problem of data dimension growth. The problem of a significant rise in the amount of calculation produced by increasing the number of layers can minimize overfitting, enhance the accuracy of English grammatical fault detection, and reduce time loss.

3.4.2. Improvement of Grammatical Error Correction (GEC) System. If the subject-verb agreement module is placed before the noun singular and plural module, the traditional GEC system will not be able to obtain correct detection results. The traditional GEC system cannot effectively deal with grammatical errors in example sentences and cannot accurately detect obvious singular and plural errors of nouns and inconsistent subject-verb errors. In order to detect sentences containing multiple grammatical errors, this paper uses a machine translation model to improve the GEC system. The following formula is the expression of the machine translation model:

$$p\left(c_j|xy\right) = \frac{p\left(x|c_j\right)p\left(y|c_j\right)}{p(xy)} = \prod_{j=1}^{n} \frac{p(x|y)p\left(c_j\right)}{p(xy)}. \tag{14}$$

Among them, $p(x|c_j)$ represents the source language and $p(y|c_j)$ represents the target language. The model automatically extracts phrase-based bilingual dictionaries from the parallel corpus, calculates the translation model parameter $p(xy)$, and extracts the N-ary sequence from the parallel corpus to calculate the probability P' of the target language $p(y|c_j)$.

The GEC system, which is based on the machine translation paradigm, considers the learner's language output to be the source language and equals the text with the target language after fixing the grammatical faults. The accuracy of grammatical error detection is greatly improved by modifying the wrong learner corpus training model parameters on a large scale.

3.5. English Grammar Error Detection Process. After the training of the data in the corpus is completed, all words, part of speech tags, and their corresponding ids have been saved in the database. On the basis of these data, the machine translation model can be used to perform grammatical error-checking operations. After the inspection module receives the input sentence, it performs word segmentation, part of speech tagging, sentence analysis, and finally initialization to obtain the corresponding sentence model. It then performs grammatical error detection, processes the information obtained in the detection, and finally obtains the detection result. Figure 2 shows a flowchart of English grammar error detection.

4. Simulation Results and Analysis

In the previous section, the English grammatical error detection method based on the machine translation model was designed and analyzed. In this section, the effect of the method will be evaluated. It mainly includes four aspects, namely, kappa value, precision value, Recall value, and misjudgment rate. Four sets of experiments were done to verify the effectiveness of the method proposed in this article. The specific experimental results of each item are analyzed below.

4.1. Experimental Data Set. UICLE corpus is the first publicly available corpus of learners with native background/$L1$ and error types. The corpus contains up to 75 error types in total. In practical applications, the error codes are usually merged and used as training or test set. Table 5 is a specific experimental dataset.

4.2. Experimental Indicators. This article uses different evaluation standards for the evaluation of test results. The different indicators are introduced below.

(1) Kappa coefficient: It is a statistical value used to evaluate the consistency of English grammatical error detection results. The larger the kappa value, the higher the consistency.

(2) Precision: Refers to the correctness of the English grammatical error detection result. Its definition formula is

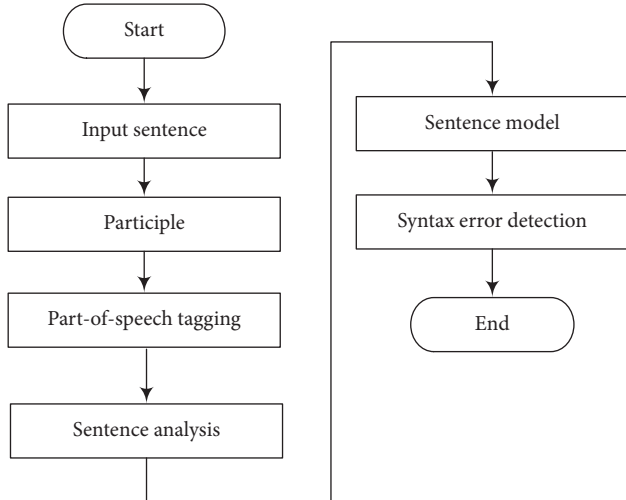$$\text{Precision} = \frac{\sum_{i=1}^{N}\left(q_i \cap x_i\right)}{\sum_{i=1}^{N} x_i}. \tag{15}$$

FIGURE 2: English grammar error detection process.

FIGURE 3: Consistency test results of different methods.

TABLE 5: Experimental dataset.

Dataset	Number of texts/piece	Number of error types/piece	Number of words/10,000
1	1954	9	3.5
2	5312	21	13
3	3307	15	8.9
4	989	3	0.8
5	2216	12	6.7

(3) Recall: It is the detection ratio of all English grammatical errors by the evaluation method, which is the recall rate. Its definition formula is

$$\text{Precision} = \frac{\sum_{i=1}^{N}\left(q_i \cap x_i\right)}{\sum_{i=1}^{N} q_i}. \tag{16}$$

In the above formulas, the numerator is the number of actual errors in the sentence that the detection result matches where the denominator of formula (15) is the number of detected English grammatical errors, and the denominator of formula (16) is the number of actual errors.

(4) Misjudgment rate: Misjudgment occurs in the experiment due to different reasons, however, there are three main reasons for it: firstly, it is difficult to check special words. Special words mean some proper nouns that are used less frequently, such as place names, people's names, school names, and specific conjunctions. Because such terms are rarely used, they may not be included in the training corpus, resulting in misjudgment owing to a lack of knowledge. Secondly, it is difficult to check the use of punctuation because there are many punctuation positions such as quotation marks and question marks, especially quotations marks. What is said in quotation marks can be any word, which is difficult to count. Thirdly, errors in the test corpus itself. There will be some input errors in the test corpus, such as word errors and English
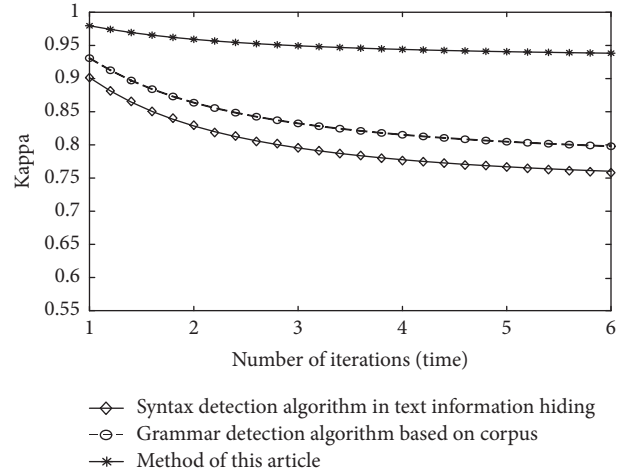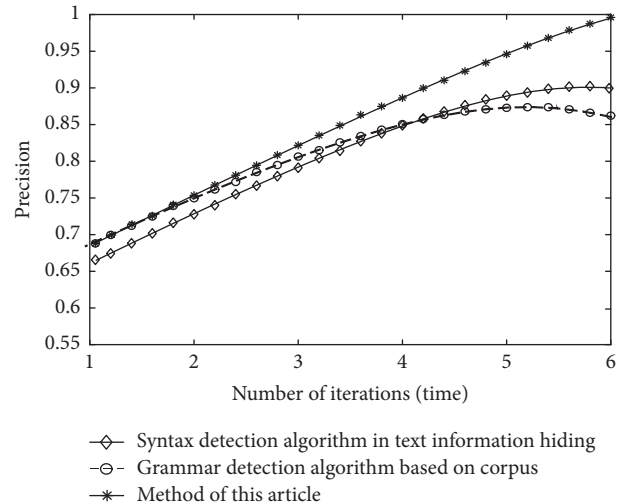

FIGURE 4: Comparison of results of precision values of different methods.

punctuation errors. These errors are recorded in the test corpus as correct, resulting in misjudgment.

4.3. Experimental Results.
The grammar detection algorithm in text information hiding and the checking and correcting method of grammatical errors in English articles based on corpus are used as a contrast method to compare with the method in this article. The following are the specific experimental results.

4.3.1. Consistency Inspection.
Figure 3 shows the consistency test results of different methods, that is, the comparison result of the kappa coefficient.

It can be seen from Figure 3 that as the number of iterations increases, the consistency of the English grammatical error detection results of the three methods shows a downward trend. Among them, the method in this paper has a slower downward trend, and the grammar detection in text
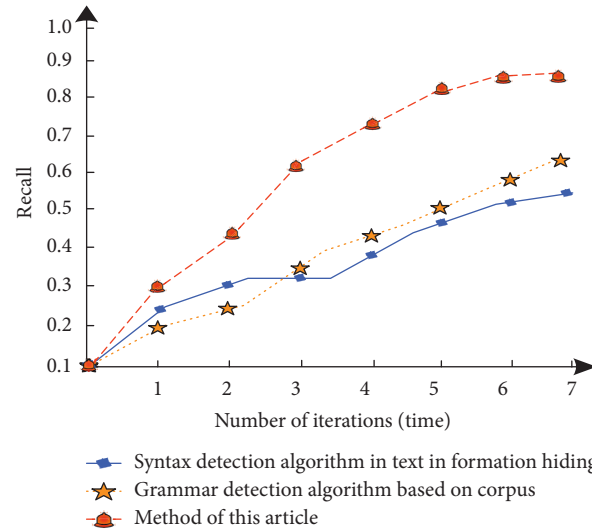
FIGURE 5: Comparison of results of recall values of different methods.

TABLE 6: Comparison of results of misjudgment rates of different methods.

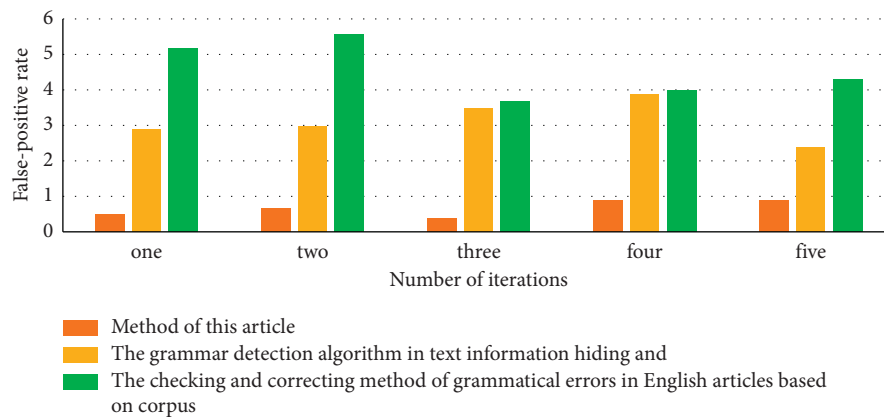| Number of iterations/time	Method of this article	The grammar detection algorithm in text information hiding	The checking and correcting method of grammatical errors in English articles based on corpus
1	0.5	2.9	5.2
2	0.7	3.0	5.6
3	0.4	3.5	3.7
4	0.9	3.9	4.0
5	0.9	2.4	4.3
6	0.8	2.7	5.2

FIGURE 6: Comparitive result of misjudgment.

information hiding algorithms and the checking and correcting method of grammatical errors in English articles based on corpus have a clear downward trend. Through comparison, it can be seen that the kappa coefficient of the method in this paper is higher, indicating that the gap between the detection result and the actual grammatical error is small, and the detection result is closer to the true value.

4.3.2. Precision Value. In order to compare the accuracy of the detection results of different methods more objectively,

the precision value comparison is carried out. The comparison result is shown in Figure 4.

Analyzing Figure 4, it can be seen that at the beginning of the experiment, the difference between the precision value of the method in this paper and the two traditional methods is not obvious. When the number of iterations reaches 3 times, the gap gradually becomes obvious. The precision value of this method shows a linear growth trend. The highest precision value is close to 1, indicating that the English grammar error detection result of this method is more accurate. From the experimental results, in terms of detecting English

grammatical errors, the detection results of the method in this paper are more accurate.

4.3.3. Recall Value.
Comparing the recall value of different methods, the comparison result is shown in Figure 5.

The recall values of the two traditional methods are low, as shown in Figure 5, indicating that the detection proportion obtained in the process of syntax error detection using the traditional method is low, i.e., the recall rate is low, which is far lower than the detection ability of the method in this paper. This is mainly because the traditional method has no pertinence in the process of training. When training, only the correct corpus is used instead of the corpus containing errors, and the recognition effect is poor, which is just corrected by this method.

4.3.4. Misjudgment Rate.
The misjudgment rates of different methods are compared, and the comparison results are shown in Table 6 and Figure 6.

Figure 6 demonstrates that, when compared to the conventional approach, this method has a substantially lower misjudgment rate, with a minimum misjudgment rate of just 0.4, which is significantly lower than the traditional method. It shows that the overall performance of the method in this paper is improved compared with the traditional method, which shows that the improvement has achieved certain results.

5. Conclusion

The traditional methods used for the English grammatical error detection suffered from several problems including the problems of low kappa value, precision value, and recall value. Apart from these problems, the misjudgment rate in the conventional system was also high. Taking these challenges into consideration, this paper proposes an English grammar error detection method based on a machine translation model. To do so, a Bi-LSTM model was created to diagnose English grammatical faults, the Naive Bayes method was used to classify the results of the English grammatical error diagnosis, and the N-gram model was used to effectively point out the error's location. The experimental verification shows that the detection result of this method is better and the accuracy rate is higher, which can provide a more reliable reference for English learners. At present, this article only studies the type and location of the detected errors, and does not correct the errors, so future research will start from the following two aspects: (1) Obtain more training data to enhance the model. (2) The model will not only identify errors but will directly correct grammatical errors as well.

Data Availability

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

Conflicts of Interest

The author declares no conflicts of interest.

References

[1] J. Zhu, X. Shi, and S. Zhang, "Machine learning-based grammar error detection method in English composition," *Scientific Programming*, vol. 2021, pp. 1–10, 2021.

[2] C. Hongli, "Design and application of English grammar error correction system based on deep learning," *Security and Communication Networks 2021*, vol. 2021, Article ID 4920461, 9 pages, 2021.

[3] M. Fu, Z. X. Dai, and W. T. Hu, "A syntax checking algorithm for information-hiding," *Science Technology and Engineering*, vol. 15, no. 21, pp. 142–145, 2015.

[4] Y. M. Tan, X. H. Wang, and Y. X. Yang, "Grammatical error correction based on corpus," *Journal of Beijing University of Posts and Telecommunications*, vol. 39, no. 04, pp. 92–97, 2016.

[5] Z. He, "English Grammar Error Detection Using Recurrent Neural Networks," *Scientific Programming*, vol. 2021, Article ID 7058723, 8 pages, 2021.

[6] R. Catelli, V. Casola, G. De Pietro, H. Fujita, and M. Esposito, "Combining contextualized word representation and sub-document level analysis through Bi-LSTM+CRF architecture for clinical de-identification," *Knowledge-Based Systems*, vol. 213, no. 1, 106649 pages, 2021.

[7] S. Zhou and W. Liu, "English Grammar Error Correction Algorithm Based on Classification Model," *Complexity*, vol. 2021, Article ID 6687337, 11 pages, 2021.

[8] A. Khajenezhad, M. A. Bashiri, and H. Beigy, "A distributed density estimation algorithm and its application to naive Bayes classification," *Applied Soft Computing*, vol. 98, no. 6, 106837 pages, 2021.

[9] C. Hongli, "Design and Application of English Grammar Error Correction System Based on Deep Learning," *Security and Communication Networks*, vol. 2021, Article ID 4920461, 9 pages, 2021.

[10] F. Long, "The Construction of Machine Translation Model and its Application in English Grammar Error Detection," *Security and Communication Networks*, vol. 2021, Article ID 2731914, 11 pages, 2021.

[11] H. K. Hamarashid, S. A. Saeed, and T. A. Rashid, "Next word prediction based on the N-gram model for Kurdish Sorani and Kurmanji," *Neural Computing & Applications*, vol. 33, no. 9, pp. 4547–4566, 2020.

[12] X. Wang and W. Zhong, "Research and Implementation of English Grammar Check and Error Correction Based on Deep Learning," *Scientific Programming*, vol. 2022, Article ID 4082082, 10 pages, 2022.

[13] J. Nie, C. Zheng, P. Zeng, B. Zhou, L. Lei, and P. Wang, "Using the theory of planned behavior and the role of social image to understand mobile English learning check-in behavior," *Computers & Education*, vol. 156, no. 10, 103942 pages, 2020.

[14] R. . Zhang, "Modeling and simulation of English speech rationality optimization," *Computer Simulation*, vol. 34, no. 2, pp. 289–292, 2017.

[15] N. S. Khan, A. Abid, and K. Abid, "A novel natural language processing (NLP)–Based machine translation model for English to Pakistan sign language translation," *Cognitive Computation*, vol. 12, no. 2, pp. 1–18, 2020.

[16] Q. Ren, L. S. Yi, and W. L. Wan, "Research on the LSTM Mongolian and Chinese machine translation based on morpheme encoding," *Neural Computing & Applications*, vol. 32, no. 1, pp. 41–49, 2020.

[17] U. V. Ucak, T. Kang, J. Ko, and J. Lee, "Substructure-based neural machine translation for retrosynthetic prediction," *Journal of Cheminformatics*, vol. 13, no. 1, pp. 4–15, 2021.