An Improved Classification Model for English Syntax Error Correction Design of DL Algorithm

Xue Han
College of English and Tourism, Henan Institute of Economics and Trade, China
hanxue24@outlook.com

Abstract: *To better respond to the slogan of smart teaching in universities and fully integrate various artificial intelligence technologies with educational learning, many scholars have conducted research on teaching methods and models in universities. Traditional English teaching often uses manual verification to correct grammar errors. In view of the shortcomings of the traditional manual English grammar correction methods, such as low efficiency, time-consuming, this paper combines the deep learning technology to design an English Syntax Error Correction (ESEC) model based on the Transformer structure. The paper first introduces the working principle of traditional neural networks in syntax error correction, and then studies the Generative Adversarial Network (GAN) and Transformer structure. Finally, the Transformer structure is integrated with the GAN and the ESEC model to create the final syntax error correction model. The results of testing the performance of the model showed that the designed model had good performance. The recognition accuracy, recognition recall, and F1 values on the test dataset CoNLL-2020 were 0.98, 0.96, and 0.97, respectively. The three values on the JFLEG test set dataset were 0.96, 0.98, and 0.97, respectively. In conclusion, the English grammar error correction model proposed in this paper demonstrates satisfactory performance, and its implementation in practical English grammar error correction tasks yields similarly positive results.*

Keywords: *Universities, English, syntax error, classification model, neural network, attention mechanism.*

Received January 25, 2024; accepted June 2, 2024
https://doi.org/10.34028/iajit/21/4/2

1. Introduction

English grammar is an important component of English learning and is crucial for understanding and applying English. However, in actual writing and translation, syntax error are still a common problem [14, 22]. For example, syntax error may lead to incorrect or difficult sentences to understand, which may affect translation and text quality. Therefore, how to effectively correct syntax error has become an important research topic in the Natural Language Processing (NLP). In the past few years, deep learning achieved great achievements in the field of NLP [6]. Among them, deep learning technologies such as Generative Adversarial Network (GAN) and Transformer structure are extensively utilized in NLP tasks [1]. These techniques can effectively generate natural language text and correct syntax error on this basis.

However, the traditional GAN and Transformer structures are not suitable for the correction of English syntax error [12]. The traditional GAN usually adopts supervised learning, while the Transformer structure requires unsupervised learning to correct syntax error. Unsupervised learning often faces problems that are difficult to train and predict [8]. Therefore, this paper attempts to combine GAN and Transformer structures to address the aforementioned issues. Based on this background, the paper first introduces the relevant technologies of the traditional neural network model in speech processing, and then uses Transformer structure and GAN to build an English Syntax Error Correction (ESEC) model. Its purpose is to deal with the matters of low detection accuracy and poor detection performance of the current grammar correction model. This study is segmented into five parts, starting with an overview of the entire research content. The second part is a summary of the current research, followed by the design of the construction method of ESEC model. The fourth part is to verify the performance of the model in the result analysis section. The final part provides a summary of the entire article in the conclusion part and proposes directions for future work.

2. Related Work

The Transformer model, as a type of deep learning model, is widely adopted in lots of fields. To address the serious impact of incorrect labels on remote supervised relationship reminders in practical applications, Xiao *et al*. [17] proposed a Transformer module using a mixed attention mechanism for remote supervised relationship reminders in multi-instance learning. The designed method outperformed the most advanced algorithms in remote supervised relationship reminder tasks. In addition, the Transformer model had also been applied in the mechanical processing industry. Zhang *et al*. [21] found that traditional automatic speech error detection methods cannot fully utilize the prior knowledge of the target text, so they proposed applying the Transformer model to it. This method could achieve a relative improvement of 8.4% on the F-1 scoring index, which had significant implications for optimizing automatic language error detection methods. Li *et al*. [11] believed that the existing anomaly detection methods in the power industry had not fully explored the potential value of data, so they proposed an anomaly detection model based on graph attention and Transformer. They designed experiments based on electricity data from a certain region in China and found that this anomaly detection method could effectively detect anomalies. Li *et al*. [10] believed that the current neural Text-to-Speech (TTS) model had robustness issues that lead to audio anomalies. To construct a neural network model that could simultaneously synthesize natural audio and stable audio, a Transformer-based TTS model called RoboTrans was proposed by Li *et al*. [10]. The experiment found that the model solved the robustness problem of the TSS model. Xiao *et al*. [18] found that existing entity and relationship extraction algorithms cannot recognize the correspondence between relationships and sentences, resulting in noise labeling issues. Therefore, a hybrid deep neural network model based on models such as Transformer has been proposed. Through different experiments, it has been shown that the model has achieved good results in entity and relationship extraction, and has the ability to filter noisy sentences.

Yang *et al*. [19] hoped to validate the practicality of deep learning algorithms in the medical field by successfully identifying previous spinal implants through the application of the algorithms. This experiment demonstrated that deep learning algorithms were effective and practical for spinal implant recognition. Zhang *et al*. [20] proposed a hybrid wind speed prediction model built on multivariate data binary decomposition method and deep learning algorithm to solve the problems of data decomposition-based prediction models in wind power generation systems. For accuracy and effectiveness, it was greater than other wind speed prediction models. Fan *et al*. [5] designed a novel end-to-end unsupervised deep learning video anomaly detection model using perceptual GAN. The proposed model achieved the purpose of training by classifying network videos, and adding perceptual GAN could assist the model in better video detection, thereby helping the model identify videos with abnormal states. Testing the performance of the model on multiple popular benchmarks has shown that the adopted model had good detection performance. Li and Mao [9] raised a GAN-based prediction model to accurately predict the temperature of molten steel during the heating stage of an electric arc furnace in real time. This experiment found through alternating training of the discriminator and generator that the model had higher accuracy and effectiveness in predicting the temperature of molten steel. Sajjad *et al*. [15] proposed a new method using deep convolutional GAN to generate three different stages of Alzheimer's disease to integrate data and improve the accuracy of disease diagnosis models. This was more excellent than others in synthesizing brain Positron emission tomography images of all three-stage of Alzheimer's disease.

In summary, many scholars have conducted a series of studies on Transformer structure and GAN. Among them, research on Transformer structure is mainly focused on the detection of various abnormal signals and data, while research on GAN is mainly focused on the fields of signal prediction and image processing. Based on this situation, this paper innovatively combines the two to better extract the features of English syntax error, and identify and detect their wrong grammar.

3. A Classification Model of ESEC Based on Neural Network

ESEC, as an essential field in NLP, has attracted many experts' attention in recent years. Aiming at a series of shortcomings of traditional machine translation methods in grammar error correction, this paper uses the idea of encoder decoder to build an ESEC model through the Transformer structure. To further address the issue of low recognition and detection accuracy of open syntax error, this paper integrates GAN and Transformer structures and designs the final syntax error correction model.

3.1. Construction of ESEC Model Based on Transformer Structure

ESEC is a method of using computer technology to help students improve their English grammar knowledge and skills in the process of language learning. This method helps students better understand and master English grammar rules and improve the accuracy and standardization of language use by analyzing and correcting syntax error that students may make in the learning process [3]. Currently, with the continuous innovation and optimization of artificial intelligence technology, many scholars have built English grammar correction models using various intelligent learning methods. The earliest grammar error correction methods mainly used manual correction. Although this method had high detection accuracy, it also consumed a lot of manpower and material resources, so it was not suitable for large-scale promotion. In the current research on syntax error correction model, automatic syntax error correction is a popular research direction, whose goal is to complete the automatic correction function of syntax error correction model.

The common Recurrent Neural Network (RNN) is expanded according to the time step. When using RNN to correct English grammar errors, its working principle is to automatically correct syntax error by transforming the input natural language text into the output of RNN. In RNN, each neuron is connected to input and output

[16]. Among them, input includes lexical, syntactic, and semantic information of the text. During the training process, RNN will learn patterns and patterns from natural language texts and apply them to the output. In English grammar correction, RNN can learn the knowledge of morphology, syntax and semantic information in English text, and automatically correct syntax error. Using RNN to identify syntax error avoids manual parsing and manually designed rules. It's working principle is to use machine learning algorithms and NLP technology to automatically identify and correct syntax error. Figure 1 shows the unfolding structure of RNN.

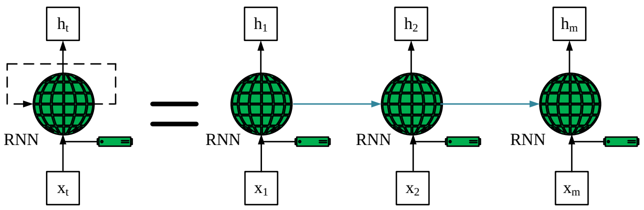


Figure 1. RNN expansion structure.

In Figure 1, x_t represents the input of time step at time t, and t=[1, 2, …m]. h_t represents the implicit state of the time step output at time t. The standard expressions for RNN are Equations (1) and (2) [4].

$$h_t = \varphi_h \left(W_h x_t + U_h h_{t-1} + b_h \right) \tag{1}$$

Equation (1) is the expression for the implicit state output of the time step at time t. W_h, U_h and b_h are the relevant parameters of the output implicit state, respectively. φ_h represents the nonlinear activation function of the output implicit state.

$$y_t = \varphi_y \left(W_y h_t + b_y \right) \tag{2}$$

Equation (2) is the calculation formula for the network output at t. y_t represents the output of the network at t. W_y and b_y are the relevant parameters of the network output state. φ_y represents the nonlinear activation function of the network output. Due to the original RNN is prone to the problems of gradient disappearance and gradient explosion when updating parameters, the paper further uses Transformer structure to build a syntax error correction model on the basis of RNN. Both encoder and decoder are neural network structures used for sequence to sequence learning. The Transformer structure is a type of encoder structure, which can convert word or other feature representations in the input sequence into vector representations, and then use these vectors for sequence to sequence learning. The Transformer structure combines the encoder and decoder to form a complete learning system that can better handle sequence data, especially long text problems. Figure 2 is an encoder decoder model diagram.

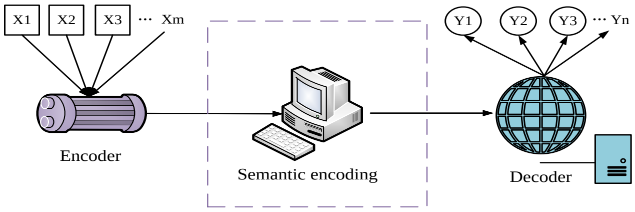


Figure 2. Encoder-decoder model.

In Figure 2, the entire model has three parts: encoder, semantic encoding module, and decoder. Attention model is a neural network model that can not only recognize and capture important information, but also not be disturbed by other information during the recognition process. In deep learning, attention model is widely used in various tasks, such as speech recognition, machine translation, text generation, image generation, etc. Attention models can more accurately capture and analyze important information in data, thereby improving the prediction accuracy and efficiency of the model. This paper combines the attention model with the encoder decoder model, resulting in an encoder decoder model with attention mechanism as shown in Figure 3.

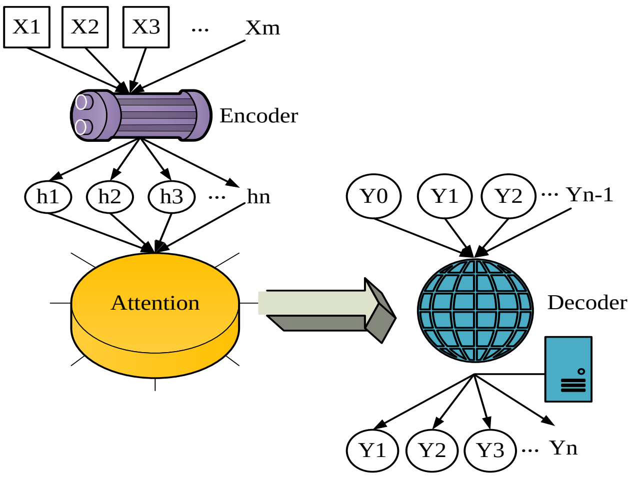


Figure 3. Encoder-decoder model with fused attention mechanism.

In Figure 3, the entire model has added an additional attention mechanism module on the top of the encoder and decoder to enhance the model's ability to process text information. The calculation process of encoding decoding is Equations (3) and (4) [2].

$$e_1, e_2, \cdots, e_m = encoder\left(X1, X2, \cdots, Xm \right) \tag{3}$$

In Equation (3), e_1, e_2 , …, e_m represents a sequence of input text. *encoder*($X1$, $X2$, …, Xm) represents encoder encoding.

$$Y_t = decoder\left(e_1, e_2, \cdots, e_m, Y1, Y2, \cdots Y_{t-1} \right) \tag{4}$$

In Equation (4), Y_t represents the probability distribution vector of the decoded data, which is calculated using the Softmax function. *decoder*(e_1, e_2 , …, e_m, $Y1$, $Y2$, …, Y_{t-1}) represents decoding the probability distribution vector.

Transformer is a neural network structure based on attention mechanism, used for modeling and generating input sequences. Transformer models employ Self-Attention Mechanism (SAM) to generate distinct vectors at various positions within the input sequence, thereby enabling the representation of different parts of the sequence. Compared to traditional RNN and Convolutional Neural Network (CNN), Transformer has stronger ability to process long sequence information. In NLP, Transformer structure is widely used in various NLP tasks, e.g., text classification, emotion analysis, machine translation, language generation, etc. Based on this background, this study first utilizes the Transformer structure in neural networks as an encoder decoder, and then fuses attention mechanisms to generate an optimized encoder decoder model. Thus, the ESEC model is further built, aiming to analyze the text information in English sentences through the model, so as to detect the wrong grammatical information and correct it. Figure 4 shows the SAM in the Transformer structure.

In the Transformer of Figure 4, SAM refers to a mechanism used to capture key information in sequence data. SAM can represent different vectors of input as different relationships, thereby better understanding the internal structure and patterns of input data. SAM is used as the attention mechanism module in the Transformer model, and the Transformer model is built as displayed in Figure 5 for the recognition and detection of English incorrect grammar.

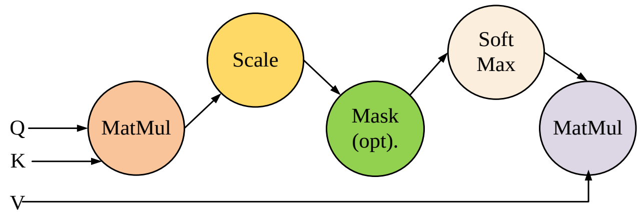


Figure 4. SAM structure diagram.

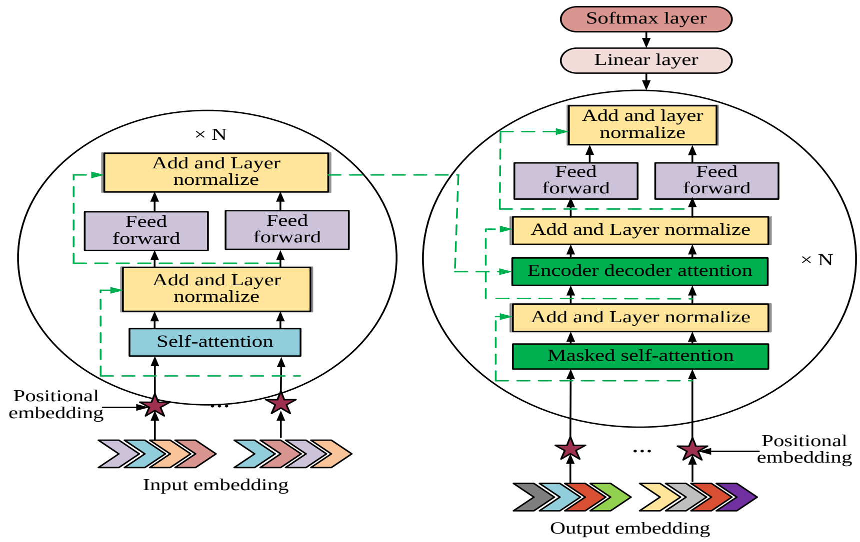


Figure 5. Transformer model structure diagram.

In Figure 5, if the traditional Transformer performs attention mechanism training on each word once, it will obtain a vector representation containing all words, which is the feature vector of that word. If the SAM is introduced in the Transformer, the model will pre train each word and then fine tune each word. In this way, the Transformer model combined with SAM can better capture the internal structure and pattern of the word, thereby improving the prediction accuracy and efficiency of the model.

3.2. An ESEC Model with Improved GAN and Transformer Structure

With the continuous optimization of neural networks, the error syntax detection model is no longer limited to syntax error between local contexts, but more targeted at the recognition and detection of open syntax error. Based on this background, this paper is no longer limited to using the Transformer model to classify the context grammar and diagnose the wrong grammar, but regards syntax error correction as a separate language translation task. It achieves the purpose of correction by translating sentences containing syntax error into sentences written normally. Traditional Transformer models face issues such as exposure bias during the training process, making it easy for the model to output incorrect prediction results within a certain time step, resulting in errors in subsequent detection results.

Adversarial learning is a machine learning method that involves two competing learning processes, one is GAN and the other is Classification GAN (CGAN). GAN is a deep learning model used for image generation, consisting of two parts: a generator and a discriminator. The generator goal is to generate new images by predicting a series of images, while the goal of the discriminator is to identify these images by verifying their authenticity and trustworthiness [7]. To cope with the complexity of open syntax error recognition and improve the detection accuracy, this study innovatively integrates GAN and Transformer structure. The fused GAN generator is capable of generating grammatically correct sentences, while the discriminator evaluates the accuracy of grammatical correction. This process improves the model's ability to learn and correct complex grammatical structures. In addition, by introducing GAN, the fusion model can effectively narrow the difference between generated sentences and real sentences in the continuous learning process, thus improving the effect of grammar error detection. Figure 6 is the basic framework of generative adversarial learning.

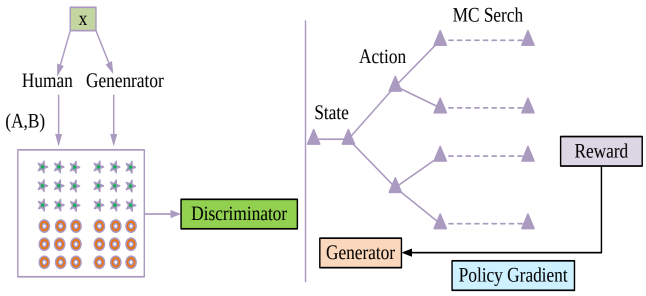


Figure 6. Basic structure of generative adversarial learning.

In Figure 6, the entire adversarial framework consists of a generator and a discriminator. The generator is mainly used for the neural network syntax correction model, which is an encoder decoder model using a sequence to sequence framework. This paper selects the Transformer structure as the generator. The discriminator is a CNN-based binary classification model, whose main responsibility is to distinguish whether a grammar correction action is completed manually or by a generator. The generator is taken as a parameterized random strategy that generates countless time steps when correcting grammar sentences [13]. The intelligent model can take corrective actions based on random strategies, using incorrect sentences as input to the discriminator, and then identifying error types through the discriminator, and outputting specific probability values as reward feedback to the generator. Continuous reinforcement can ultimately maximize the expected reward for the model, thereby improving its accuracy in identifying incorrect syntax. In Figure 6, assuming *a* "error correction" statement is (A, B), and a generator is given as G, with its parameters represented by θ, the expression for the initial source side error sentence is obtained as Equation (5).

$$a = \left(a_1, a_2, \cdots, a_m\right) \quad a_m \in A' \tag{5}$$

In Equation (5), A' represents the source vocabulary. $a=(a_1, a_2, \ldots, a_m)$ represents the incorrect sentence at the initial source.

$$b = \left(b_1, b_2, \cdots, b_n\right) \quad b_n \in B' \tag{6}$$

In Equation (6), B' represents the target vocabulary. $b=(b_1, b_2, \ldots, b_n)$ represents the corrected sentence on the target end. The definition of state is Equations (7).

$$s = \left(b_1, b_2, \cdots b_{t-1}\right) \tag{7}$$

In Equation (7), state s represents the prefix sequence currently generated by the generator, represented by (b_1, b_2, …, b_{t-1}). The definition of action l is Equations (8).

$$l = b_t \tag{8}$$

In Equation (8), action l is defined as generating the next word b_t and treating the generator as a random policy model. In the generated adversarial model mentioned above, an additional discriminator D is added to guide the learning of the generator. The discriminator uses a strategy gradient method for parameter updates. The expected reward that the model can obtain during adversarial training is the ultimate optimization goal of the training model. Given the satisfactory performance of the CNN in the classification task of this paper, the CNN is selected as the fundamental structure of the discriminator. The relationships between the features of incorrect sentences and those of corrected sentences are extracted through multi-layer convolution and pooling operations. Figure 7 shows the basic structure of the model using CNN structure as the discriminator.

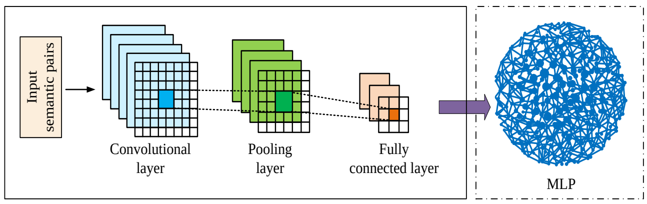


Figure 7. Basic structure of the discriminator model.

In Figure 7, the whole discriminator model is mainly composed of CNN and Multi-Level Recurrent Processing (MLP). The CNN part mainly extracts features through operations such as convolution and pooling. MLP layer can realize MLP of data through multiple perceptron, thus realizing more complex tasks. Due to the ability of MLP to process and analyze data at different levels, more complex tasks can be achieved in neural networks, and the patterns and features of data can be more accurately understood. Therefore, this paper combines CNN with MLP to ensure that the feature relationship between incorrect sentences and corrected sentences can be fully extracted, thus obtaining the expected output. Assuming that the English sentence pair in the input network is (E, F), the discriminator will first concatenate the word vectors corresponding to each word in E and F to construct a two-dimensional image input. Next, the height and width of the two-dimensional image are set to the E and F to obtain the i-th and the j-th word in E F, recording them as the (i, j)-th position in the corresponding image matrix. The feature mapping expression of position (i, j) is Equation (9).

$$Z_{i,j} = \left[E_i, F_i\right] \tag{9}$$

In Equation (9), $Z_{i,j}$ represents the feature mapping of the image matrix at position (i, j). Relu is taken as the nonlinear activation function, and its expression is Equation (10).

$$\text{Relu}=\begin{cases} x & x>0 \\ 0 & x\leq 0 \end{cases} \tag{10}$$

In Equation (10), x represents the parameter in the Relu function. Through the feature mapping in Equation (9), the convolution operations of E and F can be obtained as Equation (11).

$$Z'_{i,j}=\sigma\left(W'\text{Conv.}\left[E_i,F_i\right]+b'\right) \tag{11}$$

In Equation (11), $Z'_{i,j}$ represents the convolution operation on the E and F words at position (i, j). σ is the non-linear activation function, namely Relu. W' and b' represent the weights and thresholds of the convolutional layer, respectively. Conv.[E_i, F_i] represents the convolution of English sentences with (E, F).

$$Z''_{i,j}=\max\left(W''\text{Pooling}\left[E_i,F_i\right]+b''\right) \tag{12}$$

In Equation (12), $Z''_{i,j}$ represents pooling the E and F words at position (i, j). W'' and b'' represent the weights and thresholds of the pooling layer, respectively. Pooling [E_i, F_i] represents pooling of (E, F) in English sentences. By multiple convolution and pooling operations, the feature relationship between incorrect sentences and corrected sentences can be extracted to the maximum extent. Finally, the discriminator model is used for output, and the output content is compared with the actual results to determine the performance.

4. Performance Analysis of ESEC Based on Neural Network

To test the performance of the ESEC model built by the above research, the result analysis part first tests the performance of the ESEC model under the Transformer structure. It is found that its performance in detection accuracy, recall rate, and F1 value is superior to other comparative models. In addition, the study further contrasts the performance between the improved Transformer ESEC model integrating GAN and the ESEC model under the traditional Transformer structure. The results show that the improved Transformer ESEC model combined with GAN has better error detection and recognition ability.

4.1. Performance Analysis of ESEC Model Based on Transformer

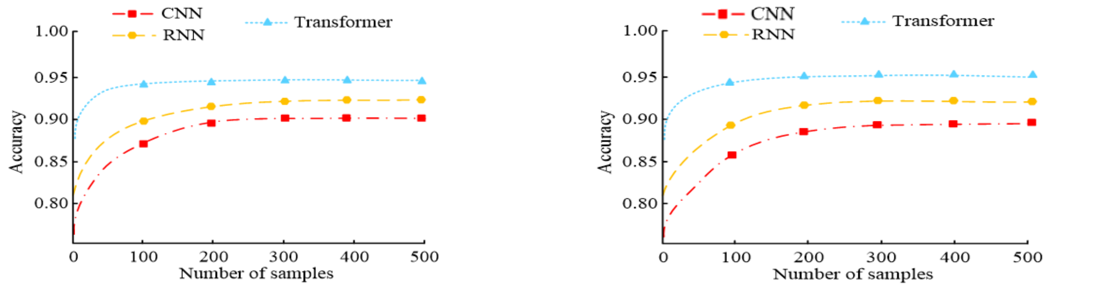
For testing the ESEC performance of under the Transformer structure, the paper selects Nucle and Lang-8 as training data sets, and trains the model several times. When the model has a certain degree of stability, the study selects CoNLL-2020 and JFLEG Test Set as test datasets to test the performance. Table 1 shows the logarithmic information of statements contained in the four datasets.

Table 1. Data set information table.

Data set type	Name	Total number of pairs of sentences	Number of available pairs of sentences
Training data set	Nucle	9530	9216
	Lang-8	8465	8155
Testing data set	CoNLL-2020 Test Set	845	818
	JFLEG Test Set	1027	994

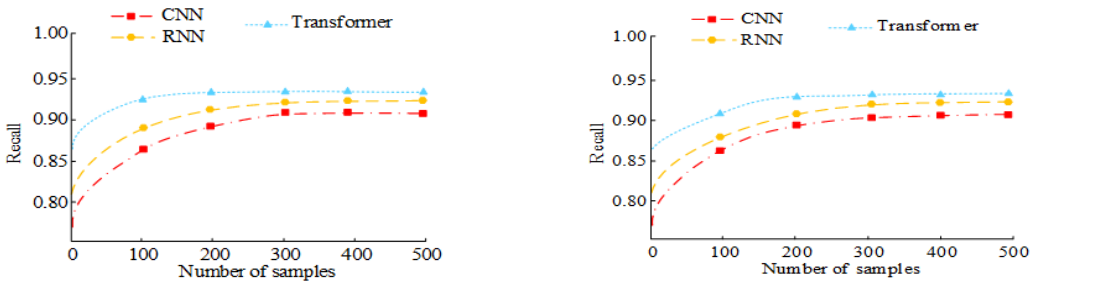
Table 1 shows the basic information of the two datasets. From Table 1, the total number of sentence pairs for the two training datasets, Nucleus and Lang-8, are 9530 and 8465, respectively. Excluding non-English interference sentence pairs, 9216 and 8150 sentence pairs are ultimately used for model training. In addition, the total sentence pairs of the CoNLL-2020 and JFLEG are 845 and 1027, respectively. Excluding non-English interference-sentence pairs, 818 and 995 sentence pairs are ultimately used for model testing. When GAN is trained, the generator and discriminator are trained alternately. First, the discriminator is fixed to train the generator and gradient descent method is used to optimize the loss function of the generator, with the goal of maximizing the miss-classification rate of the discriminator. Then, the fixed generator trains the discriminator to optimize its parameters by improving the discriminant accuracy. The Adam optimizer is then used and the learning rate is set to 0.0001 to ensure the stability and efficiency of the training process. After each training cycle, the performance of the model is evaluated using validation datasets and over-fitting is prevented by an early stop strategy. Furthermore, the parameters of the generator and discriminator are initially calibrated at the outset of the training process. This is achieved by utilizing a reduced learning rate during the initial epochs, which is subsequently increased to a standard level to enhance the resilience of the model training. The above datasets are used for model performance detection, Figure 8 shows the detection accuracy values of three different syntax correction models in the training and testing datasets.

In Figure 8, three different syntax error correction models built by neural networks are selected for comparison, namely CNN, RNN and Transformer. Figure 8-a) represents the error grammar detection accuracy values of the three models in the training dataset. As the detection sample numbers continue to increase, the detection accuracy values of the three models will gradually increase and eventually stabilize. When the model reaches a stable state, the detection accuracy value of CNN is 0.88, RNN is 0.91, and Transformer is 0.94. Figure 8-b) represents the error syntax detection accuracy values of the three models in the test dataset. When the model reaches a stable state, the detection accuracy value of RNN is 0.91, CNN is 0.87, and Transformer is 0.95. In summary, compared to CNN and RNN, Transformer can achieve a stable state faster and achieve higher detection accuracy values during the detection process.



a) Error grammar detection accuracy of different models in the training dataset. b) Error grammar detection accuracy of different models in the testing dataset.

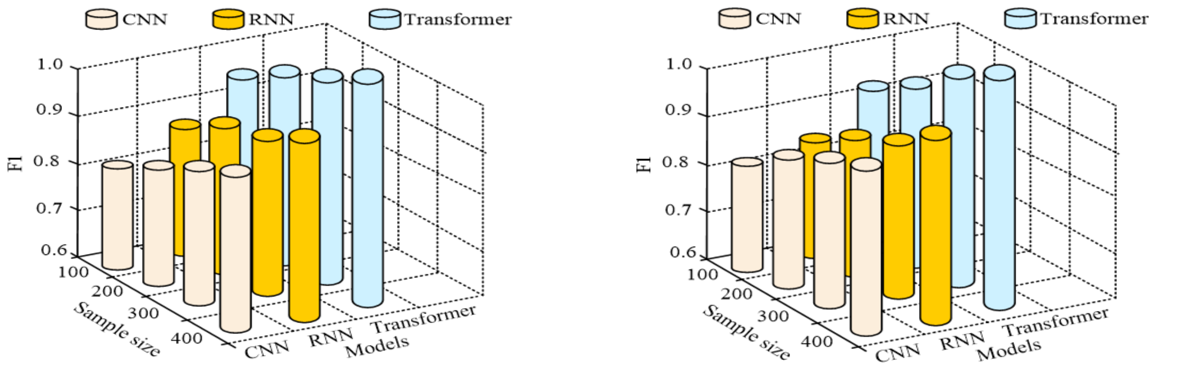
Figure 8. Accuracy values of error grammar detection for different models.



a) Error grammar detection recall of different models in the training dataset. b) Error grammar detection recall of different models in the testing dataset.

Figure 9. Recall values of error grammar detection for different models.

Figure 9-a) shows the recall values of error grammar detection for the three models in the training dataset. As the quantity of detection samples increases, the recall values of the three models also continue to increase. In the end, the Transformer is able to reach a stable state first, with a recall rate value of 0.93 in this state. Secondly, the recall rates of CNN and RNN under stable conditions are 0.91 and 0.89, respectively. Figure 9-b) represents the error syntax detection recall values of the three models in the test dataset. When the model reaches a stable state, the detection recall values for CNN, RNN, and Transformer are 0.88, 0.91, and 0.93. In summary, compared to CNN and RNN, Transformer can have better recall performance during the detection process.



a) F1 values of error grammar detecting for different models in the training dataset. b) F1 values of error grammar detecting for different models in the testing dataset.

Figure 10. F1 values of error grammar detection for different models.

Figure 10-a) shows the F1 values of error grammar detection for three models in the training dataset. When the sample size is between 100 and 400, the F1 values of CNN are between 0.8 and 0.85, RNN is between 0.85 and 0.9, and Transformer is above 0.95. Figure 10-b) is the F1 value of error syntax detection of three models in the test data set. When the amount of samples is between 100 and 400, the syntax error correction model built with Transformer structure has better F1 value performance, and its F1 value is stable above 0.95. Compared to CNN and RNN, the Transformer model has better detection performance.

4.2. Performance Analysis of Improved Transformer's ESEC Model

The above experimental results can find that the ESEC model built by using the Transformer structure has better performance. To further test the performance of the improved Transformer ESEC model integrating

GAN, this study compares the loss curve changes of the traditional Transformer ESEC model with that of the GAN-Transformer optimized by GAN, as listed in Figure 11.

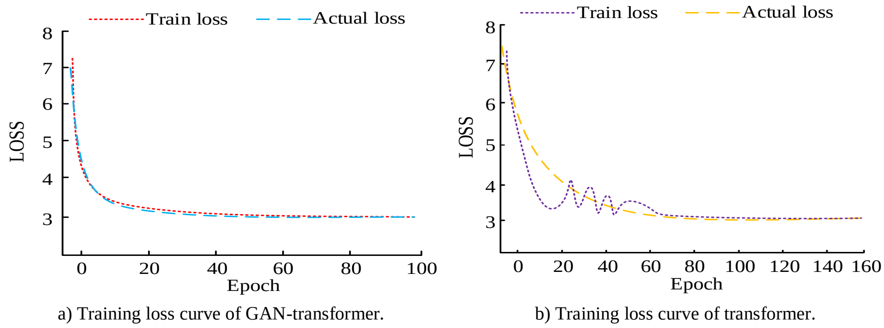


Figure 11. Loss curves for two Transformer models.

Figure 11-a) shows the changes in the loss curve of GAN-Transformer. With the increase of Epoch, GAN-Transformer can quickly iterate to a stable loss value. When Epoch is around 20, the GAN-Transformer can reach a stable loss state, with a loss value of 3.29 for the model. Figure 11-b) represents the variation of the Transformer's LOSS curve. As Epoch increases, the Transformer can ultimately iterate to a stable loss value. When Epoch is around 70, the Transformer can reach a stable loss state, and the actual loss value of the model is 3.15. In summary, during the iteration process of GAN-Transformer, the training loss values of the model have a good coincidence with the actual loss values, and the model can also quickly iterate to a stable state. Compared to the iterative changes of the Transformer, GAN-Transformer has better representation of loss values.

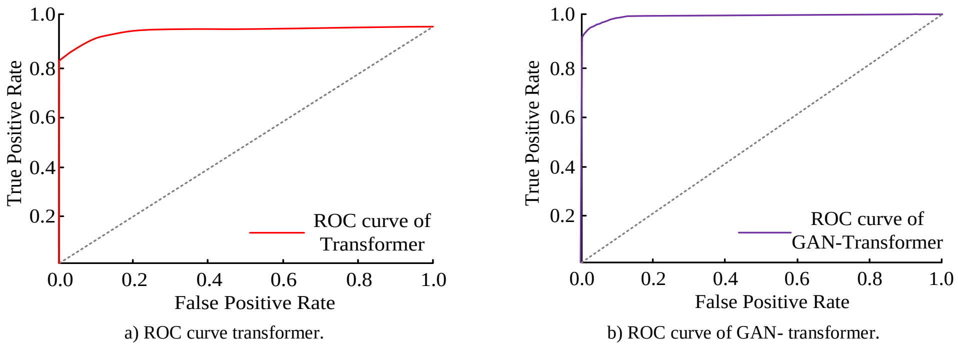


Figure 12. ROC curves of two Transformer models.

Figure 12-a) and Figure 12-b) show the Receiver Operating Characteristic (ROC) of the traditional Transformer model and GAN-Transformer. By comparing the ROC of the two, GAN-Transformer can finally reach the true positive value of 1.0 faster, which indicates that its AUC area is larger than that of traditional Transformer. Therefore, the optimized GAN-Transformer can have better error grammar recognition accuracy.

Table 2. Recognition performance of various models.

Model	**Test Data Set**	**Accuracy**	**Recall**	**F1 value**
CNN	CoNLL-2020	0.87	0.88	0.86
	JFLEG	0.88	0.87	0.87
RNN	CoNLL-2020	0.91	0.91	0.90
	JFLEG	0.88	0.90	0.89
Transformer	CoNLL-2020	0.95	0.93	0.96
	JFLEG	0.93	0.91	0.92
GAN-Transformer	CoNLL-2020	0.98	0.96	0.97
	JFLEG	0.96	0.98	0.97

Table 2 shows the comparison of recognition performance among four error grammar correction models. From Table 2, when the test dataset is CoNLL-2020 Test Set, the error syntax recognition accuracy, recognition recall, and F1 values of CNN on the dataset are 0.87, 0.88, and 0.86, respectively. The RNNs are 0.91, 0.91, and 0.90, respectively. Transformers are 0.95, 0.93, and 0.96. The GAN-Transformer values are 0.98, 0.96, and 0.97. When the test dataset is JFLEG Test Set, the three CNN values are 0.88, 0.87, and 0.87, respectively. RNN is 0.88, 0.90, 0.89. Transformers are 0.93, 0.91, and 0.92. GAN-Transformer values are 0.96, 0.98, and 0.97. In summary, the improved GAN-Transformer model shows significant performance improvements on CoNLL-2020 and JFLEG test datasets. On these two datasets, the recognition accuracy of the model is improved to 0.98 and 0.96, the recall rate is improved to 0.96 and 0.98, and the F1 value is improved to 0.97 and 0.97 respectively. These results directly demonstrate the advantages of GAN in improving the effectiveness of the standard Transformer model for English grammar error correction tasks, especially when dealing with more complex or low-frequency error structures.

To further prove the effectiveness of GAN-Transformer in English grammar error detection, five unusual grammar error types are selected for testing. The detection time and accuracy of GAN-Transformer for different grammar error types are obtained, as shown in Table 3.

Table 3. Detection time and detection accuracy in GAN-Transformer for different syntax error types.

Model	Error type	Accuracy rate /%	Detection time /s
GAN-Transformer	1	95.64%	1.22s
	2	96.29%	0.85s
	3	97.31%	1.18s
	4	98.05%	1.06s
	5	98.89%	0.69s

Table 3 shows the accuracy and detection time of GAN-Transformer for detecting five types of syntax errors, among which the lowest detection time is 0.69s and the highest detection accuracy is 98.89%. This indicates that GAN-Transformer has a good performance in detecting uncommon syntax errors.

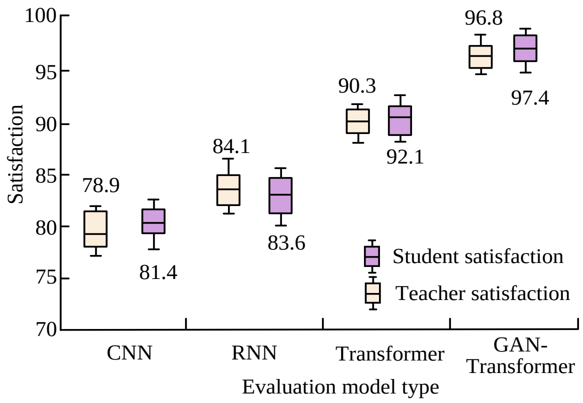


Figure 13. Students' and teachers' satisfaction with different models of English grammar error correction.

From Figure 13, the satisfaction scores of teachers and students for the four model. The satisfaction scores of CNN model are 78.9 and 81.4, respectively, for the RNN model are 84.1 and 83.6, for the Transformer are 90.3 and 92.1, and for the GAN-Transformer are 96.8 and 97.4.

5. Conclusions

In recent years, numerous scholars have employed the sequence-to-sequence learning framework neural network for research pertaining to NLP. In view of the shortcomings of traditional neural networks in English grammar detection, this paper innovatively combined GAN and Transformer encoder decoder structure to build an ESEC model based on GAN-Transformer. After testing the performance, in contrast with the traditional CNN, RNN, and Transformer structures, the optimized GAN-Transformer had better performance. The accuracy, recall, and F1 values of the CNN model in both test datasets were around 0.85, RNN was around 0.90, Transformer was around 0.95, and GAN-Transformer was above 0.95. The highest error syntax recognition accuracy value of GAN-Transformer was 0.98, the recall rate value was 0.98, and the F1 value was 0.97. The values of the three indicators were far superior to the other three comparative algorithms. In addition, compared to Transformer, GAN-Transformer had better loss curve variation. When Epoch was around 20, the GAN-Transformer could reach a stable loss state, with a loss value of 3.29 for the model. In the end, both students and teachers had a satisfaction score of over 95 points with GAN-Transformer in practical applications. In conclusion, the ESEC model designed by the paper had good performance and practical application, and could provide reference value for the research and development of NLP. However, due to the lack of extensive training on the model, there are still certain errors that are difficult to eliminate during the error correction process. In the future, more different encoder decoder models should be combined for optimization.

6. Future Work

Although the GAN-Transformer model has good syntax detection performance, limited training data sets may lead to limitations in model generality, especially when dealing with syntax structures in non-training data sets. To address this problem, future research should first consider introducing a wider range of data sources, such as English texts in different regions and different cultural contexts. In this way, the model can learn and adapt to a variety of language usages more comprehensively, thus improving its universality and robustness in practical applications. Furthermore, future research should concentrate on the explanatory and corrective mechanisms of the model. In future research, it would be beneficial to consider the introduction of interpretative modules, such as optimized attention mechanisms, which would allow for the examination of which parts of the text the model focuses on when correcting errors. Finally, to enhance the explanatory power of the model, future studies may integrate additional visual tools, such as the generation of parsing trees for sentences and the display of how the model identifies and rectifies each specific grammatical error.

Data Availability Statement

The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

References

[1] Alewijnse S., Buchin K., Buchin M., Sijben S., and Westenberg M., "Model-based Segmentation and Classification of Trajectories," *Algorithmica*, vol. 80, no. 8, pp. 2422-2452, 2018. https://dl.acm.org/doi/10.1007/s00453-017-0329-x

[2] Ay F., Ince G., Kamasak M., and Eksi K.,

"Classification of Pulsars with Dirichlet Process Gaussian Mixture Model," *Monthly Notices of the Royal Astronomical Society*, vol. 493, no. 1, pp. 713-722, 2020. DOI:10.1093/mnras/staa154

[3] Bozikas A. and Pitselis G., "Incorporating Crossed Classification Credibility into the Lee-Carter Model for Multi-Population Mortality Data," *Insurance: Mathematics and Economics*, vol. 93, no. 7, pp. 353-368, 2020. DOI:10.1016/j.insmatheco.2020.06.005

[4] Chang W. and Jiang I., "iClaire: A Fast and General Layout Pattern Classification Algorithm with Clip Shifting and Centroid Recreation," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 39, no. 8, pp. 1662-1673, 2020. DOI:10.1109/TCAD.2019.2917849

[5] Fan Y., Wen G., Xiao F., Qiu S., and Li D., "Detecting Anomalies in Videos Using Perception Generative Adversarial Network," *Circuits, Systems, and Signal Processing*, vol. 41, no. 2, pp. 994-1018, 2022. DOI:10.1007/s00034-021-01820-8

[6] Faustmann C., Bajzek M., Hick H., Edtmayer J., and Walch S., "System Models and Model Classification in Tribological System Development," *Systems Engineering*, vol. 23, no. 6, pp. 783-794, 2020. DOI:10.1002/sys.21562

[7] Heidari M., Lakshmivarahan S., Mirniaharikandehei S., Danala G., Maryada S., Liu H., and Zheng B., "Applying a Random Projection Algorithm to Optimize Machine Learning Model for Breast Lesion Classification," *IEEE Transactions on Biomedical Engineering*, vol. 68, no. 9, pp. 2764-2775, 2021. DOI:10.1109/TBME.2021.3054248

[8] Lee J., Kim H., and Kang T., "Classification Algorithm Using Halftone Features of Counterfeit Bills and CNN," *Journal of Forensic Sciences*, vol. 67, no. 1, pp. 345-352, 2022. DOI:10.1111/1556-4029.14867

[9] Li C. and Mao Z., "Generative Adversarial Network-based Real-Time Temperature Prediction Model for Heating Stage of Electric Arc Furnace," *Transactions of the Institute of Measurement and Control*, vol. 44, no. 8, pp. 1669-1684, 2022. DOI:10.1177/01423312211052213

[10] Li N., Liu Y., Wu Y., Liu S., Zhao S., and Liu M., "RobuTrans: A Robust Transformer-based Text-to-Speech Model," *in Proceedings of the 34th AAAI Conference on Artificial Intelligence*, New York, pp. 8228-8235, 2020. https://doi.org/10.1609/aaai.v34i05.6337

[11] Li Y., Kai Z., Hao X., Sheng-Ya H., Shen-Qi L., and Yu-Liang S., "Abnormal Detection Based on Graph Attention Mechanisms and Transformer," *Acta Electronica Sinica*, vol. 50, no. 4, pp. 900-908, 2022. DOI:10.12263/DZXB.20210722

[12] Lv C., Wu Z., Wang X., Dan Z., and Zhou M., "Ethnicity Classification by the 3D Discrete Landmarks Model Measure in Kendall Shape Space," *Pattern Recognition Letters*, vol. 129, no. 1, pp. 26-32, 2020. DOI:10.1016/j.patrec.2019.10.035

[13] Maree M., Eleyat M., and Mesqali E., "Optimizing Machine Learning-Based Sentiment Analysis Accuracy in Bilingual Sentences Via Preprocessing Techniques," *The International Arab Journal of Information Technology*, vol. 21, no. 2, pp. 257-270, 2024. https://doi.org/10.34028/iajit/21/2/8

[14] Mohasseb A., Bader-El-Den M., and Cocea M., "Customised Grammar Framework for Query Classification," *Expert Systems with Applications*, vol. 135, no. 11, pp. 164-180, 2019. DOI:10.1016/j.eswa.2019.06.010

[15] Sajjad M., Ramzan F., Khan M., Rehman A., Kolivand M., Fati S., and Bahaj S., "Deep sConvolutional Generative Adversarial Network for Alzheimer's Disease Classification Using Positron Emission Tomography and Synthetic Data Augmentation," *Microscopy Research and Technique*, vol. 84, no. 12, pp. 3023-3034, 2021. DOI:10.1002/jemt.23861

[16] Wang X., Cheng M., Eaton J., Hsieh C., and Wu S., "Fake Node Attacks on Graph Convolutional Networks," *Journal of Computational and Cognitive Engineering*, vol. 1, no. 4, pp. 165-173, 2022. DOI:10.47852/bonviewJCCE2202321

[17] Xiao Y., Jin Y., Cheng R., and Hao K., "Hybrid Attention-based Transformer Block Model for Distant Supervision Relation Extraction," *Neurocomputing*, vol. 470, no. 22, pp. 29-39, 2022. DOI:10.1016/J.NEUCOM.2021.10.037

[18] Xiao Y., Tan C., Fan Z., Xu Q., and Zhu W., "Joint Entity and Relation Extraction with a Hybrid Transformer and Reinforcement Learning Based Model," *in Proceedings of the 34th AAAI Conference on Artificial Intelligence*, New York, pp. 9314-9321, 2020. https://doi.org/10.1609/aaai.v34i05.6471

[19] Yang H., Kim K., Kim S., and Park J., "Deep Learning Application in Spinal Implant Identification," *Spine*, vol. 46, no. 5, pp. E318-E324, 2021. DOI:10.1097/BRS.0000000000003844

[20] Zhang S., Chen Y., Xiao J., Zhang W., and Feng R., "Hybrid Wind Speed Forecasting Model Based on Multivariate Data Secondary Decomposition Approach and Deep Learning Algorithm with Attention Mechanism," *Renewable Energy*, vol. 174, no. 8, pp. 688-704, 2021. DOI:10.1016/j.renene.2021.04.091

[21] Zhang Z., Wang Y., and Yang J., "Text-Conditioned Transformer for Automatic

Pronunciation Error Detection," *Speech Communication*, vol. 130, no. 10, pp. 55-63, 2021. DOI:10.1016/j.specom.2021.04.004

[22] Zheng Y., Li G., and Zhang W., "A New Efficient Algorithm based on Multi-Classifiers Model for Classification," *International Journal of Uncertainty, Fuzziness and Knowledge-based Systems*, vol. 28, no. 1, pp. 25-46, 2020. https://doi.org/10.1142/S0218488520500026

Xue Han born in January 1987 in Zhengzhou, Henan Province, is of Han nationality. She graduated with a Bachelor's Degree in Business English from Henan University of Technology in 2010, and subsequently earned a Master's Degree in Education from Henan University in 2012. Since 2012, she has held the position of Director of the Teaching and Research Office at Henan Institute of Economics and Trade. During this time, Xue Han has authored 6 academic papers, published 2 academic books and textbooks, participated in 8 research projects, obtained 4 patents, and received 3 academic awards.