Research Article

Lihua Cai*

Research on grammatical error correction algorithm in English translation via deep learning

https://doi.org/10.1515/jisys-2023-0282
received November 27, 2023; accepted February 19, 2024

Abstract: This study provides a concise overview of a grammatical error correction algorithm that is based on an encoder-decoder machine translation structure. Additionally, it incorporates the attention mechanism to enhance the algorithm's performance. Subsequently, simulation experiments were conducted to compare the improved algorithm with an algorithm based on a classification model and an algorithm based on the traditional translation model using open corpus data and English translations from freshmen. The results demonstrated that the optimized algorithm yielded superior intuitive error correction outcomes. When applied to both the open corpus and the English translations of college freshmen, the optimized error correction algorithm outperformed the others. The traditional translation model-based algorithm came in second, while the classification model-based algorithm showed the least favorable performance. Furthermore, all three error correction algorithms experienced a decrease in performance when dealing with English compositions from freshmen. However, the optimized algorithm exhibited a relatively smaller decline.

Keywords: error correction algorithm, English translation, deep learning

AMS Mathematics Subject Classification number: 68T07

1 Introduction

With the deepening of globalization, the prominence of English as the world's primary language is becoming increasingly evident. In cross-cultural communication, English translation plays a pivotal role [1]. While machine translation algorithms aid in English translation, the current algorithms, particularly for real-time translation, are still imperfect. Therefore, individuals are encouraged to master English as thoroughly as possible, using machine translation algorithms as aids in overcoming communication barriers arising from translation discrepancies. The process of learning English may involve grammatical errors in the translated text due to unfamiliarity with the language environment [2]. While these errors may not pose significant issues in daily spoken communication, they can introduce ambiguity in situations requiring precise information conveyance. Thus, it is imperative to correct grammatical errors during the process of acquiring the English language. Traditionally, error correction has been a teacher-supervised process. However, teachers have limited energy, making it difficult for them to provide feedback on all grammar errors [3], thereby reducing the efficiency of learning English. The emergence of intelligent algorithms offers a promising solution for grammar error correction. Recent studies have made notable contributions in this area. For instance, Zhang [4] developed an English grammar error correction model based on a sequence-to-sequence (seq2seq) approach, utilized edge computing methods to enhance performance, and verified the effectiveness of the

* **Corresponding author: Lihua Cai,** School of Foreign Studies, University of Science and Technology Liaoning, Anshan, No. 189, Qianshan Middle Road, Lishan District, Anshan City, Liaoning, 114051, China, e-mail: cailihua0412@ustl.edu.cn

improvement. Zhou and Liu [5] proposed an English grammatical error correction algorithm based on a classification model, analyzed the model architecture and optimizer of the algorithm, and verified its efficacy through simulation experiments. Lin et al. [6] treated grammatical error correction as a multi-classification task and integrated a multiple-language embedding model and a deep learning model to rectify various lexical errors in Indonesian text. Experimental results demonstrated that the word embedding-based long and short-term memory (LSTM) model delivered the most effective learning outcomes. Li et al. [7] endeavored to incorporate contextual information from pre-trained language models as a solution for the scarcity of annotation in multilingual contexts. The findings demonstrated that bidirectional encoder representations from transformers held significant promise when employed for grammar correction tasks. Additionally, Dai [8] introduced an optimized random forest model and utilized it for automated detection and rectification of pronunciation errors within English classrooms. They validated its efficacy through experiments. Wanni [9] combined a genetic algorithm with a *k*-nearest neighbor algorithm to construct an intelligent English grammar correction model. The effectiveness of this model was verified through experiments. The aforementioned studies have all discussed grammar correction and proposed corresponding solutions. However, this study considers the grammar correction issue in English translations as a special type of translation problem, aiming to translate translated texts with grammatical issues into ones without grammatical problems, thus achieving grammar correction and providing valuable references for correcting grammatical errors in translations. This study provides a brief introduction to a grammatical error correction algorithm based on an encoder-decoder machine translation structure and introduces an attention mechanism to enhance the algorithm's performance. Subsequently, simulation experiments were performed to evaluate the grammatical error correction algorithm.

2 Grammatical error correction of translated text based on a deep learning algorithm

When employing deep learning techniques to rectify grammatical errors in English translations, there are two main approaches. The first approach involves error correction using classification models, while the second focuses on correction through translation models [10]. In the former approach, common grammatical errors are first identified and labeled. Following this, a classification model is trained using a training dataset. This model is then utilized to predict the locations in the source text where grammatical errors may occur. If the prediction matches the source text, it is considered error-free; if not, a grammatical error is identified in that position within the source text [11]. In the latter approach, the problem of correcting grammatical errors is viewed as a translation problem. The source text, which may contain grammatical errors, is translated into a grammatically correct "translation." Subsequently, by comparing the source text with the "translation," grammatical problems are identified, and the "translation" becomes the corrected text [12].

Classification model-based grammatical error correction algorithms offer the advantage of relatively easier access to training corpora containing grammatical errors. However, their performance is limited by the specific types of errors present in the training corpus. In contrast, the grammatical correction algorithm treats the task of error correction as a sequence-to-sequence transformation task based on a translation model, which demonstrates its generalizability without being limited to specific types of grammar errors [13]. Consequently, this study uses the translation model to rectify the grammar of English translations.

The basic principle of the translation model-based grammatical error correction algorithm is shown in Figure 1. The grammatical error correction algorithm has a basic structure similar to that of the machine

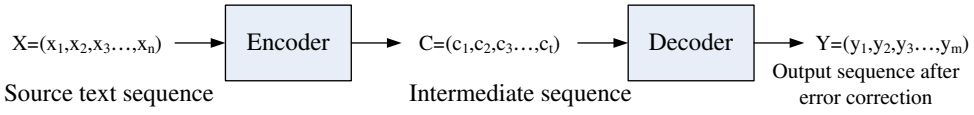

Figure 1: Fundamentals of translation model-based grammatical error correction.

translation algorithm because both refer to the principles of machine translation and consist of an encoder and decoder. When the overall algorithm carries out grammatical error correction for English translation [14], first, source text sequence X which may contain grammatical errors is input to the encoder for computation to get intermediate sequence C, and then C is input to the decoder for computation to get output sequence Y after error correction. The specific steps are shown in Figure 2.

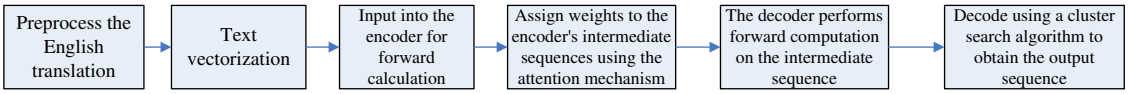

Figure 2: Steps of grammatical error correction.

(1) The English translation is preprocessed by word division and denoising, after which the English translation is text-vectored using Word2vec [15]. During preprocessing, word segmentation can be used to split abbreviation combinations and divide long sentences into shorter phrases, avoiding the processing of excessively long sequences by the encoder and decoder. Denoising refers to removing punctuation and special symbols from sentences to avoid any impact on subsequent processing.

(2) The vectorized English translation is input to the encoder for calculation, and the LSTM algorithm [16] is used within the encoder. The corresponding equations are:

$$\begin{cases} f_t = \sigma(\omega_f \cdot [h_{t-1}, x_t] + b_f) \\ i_t = \sigma(\omega_i [h_{t-1}, x_t] + b_i) \\ \tilde{C}_t = \tanh(\omega_C [h_{t-1}, x_t] + b_C) \\ C_t = f_t \times C_{t-1} + i \times \tilde{C}_t \\ o_t = \sigma(\omega_o [h_{t-1}, x_t] + b_o) \\ h_t = o_t \times \tanh(C_t), \end{cases} \tag{1}$$

where $\tilde{C}_t$ and C_t are the temporary state and update state of the "cell" at the current moment, h_t is the hidden state of the sequence data at the current moment, x_t is the input at the current moment, f_t, i_t, o_t are the outputs of the three gating cells at the current moment, namely, the forgetting, input, and output, ω_f, ω_i, and ω_o are the weights in the corresponding gating cell, b_f, b_i, b_o are biases in the corresponding gating cell.

(3) The intermediate sequence output from the encoder is fed into the decoder for decoding computation. The LSTM algorithm is also used in the decoder, and the equations are shown in the previous section. The decoder converts the intermediate sequence into another sequence and uses the converted sequence as the English-translated sequence after error correction. The traditional encoding-decoding structure assigns equal importance to each moment of data within the intermediate sequence when converting the sequence, which prevents focusing on linking moments of high relevance during decoding and leads to missing information. For this reason, the attention mechanism [17] is introduced for the purpose of assigning appropriate weights to each moment of the intermediate sequence, enabling the decoder to focus more on key moments. The calculation process is:

$$\begin{cases} c_t = \sum_{i=1}^{T} \alpha_{t,i} h_i \\ \alpha_{t,i} = \dfrac{\exp(e_{t,i})}{\sum_{j=1}^{T} \exp(e_{t,j})} \\ e_{t,i} = g(s_{t-1}, h_i), \end{cases} \tag{2}$$

where $e_{t,i}$ is the attention score of hidden state h_i in the encoder at moment t, $g()$ is the function to solve the attention score, s_{t-1} is the hidden state of the decoder at the previous moment, $\alpha_{t,i}$ is the attention weight of h_i at moment t, and c_t is the intermediate sequence at moment t.

(4) After the decoder performs LSTM forward computation on the intermediate sequence with attention weights, the sequence of character distribution probabilities is finally obtained. At this point, it needs to be decoded to obtain a definite sequence of characters. This study adopts the cluster search algorithm to decode the sequence of character distribution probabilities: the first k characters with the highest probability of distribution within each moment of the sequence are selected as a candidate output sequence, and then the sequence with the highest probability is selected [18].

3 Simulation experiment

3.1 Experimental data

The required corpus for this experiment was derived from publicly available grammar correction corpora as well as English translations produced by freshmen at the University of Science and Technology Liaoning. The grammatical error correction corpora used in this study consist of the CoNLL-2014 grammatical error correction dataset, Lang-8, NUCLE corpus, and the FCE grammatical error correction corpus. Prior to formal utilization, these corpora underwent denoising to remove sentences that were too short, contained excessive spaces, or had an abundance of special characters. This denoising process resulted in a total of 102,356 sentences. Totally 35,231 sentences were collected from the English translation written by freshmen. The corpora were divided into a training set, which consisted of 70% of the data, and a testing set, which comprised the remaining 30%.

3.2 Experimental setup

The settings of parameters for both the encoder and the decoder in the proposed algorithm are presented in Table 1. Moreover, the vector dimension for text vectorization using Word2Vec was set at 200. Before conducting formal testing with the aforementioned parameters, the grammar correction model under different parameter settings was tested. The parameters used for comparison were as follows: the activation functions of the hidden layers were set to relu, sigmoid, and tanh respectively; the number of neurons in the encoder (decoder) was set to 64 (32), 128 (64), 256 (128), 512 (256), and 1,024 (512). The performance of the error correction model under different parameter configurations was tested.

Table 1: Parameters related to encoder and decoder in the algorithm of this work

	Encoder	Decoder
Number of hidden layers	3	2
Number of neurons in hidden layer	256	128
Hidden layer activation function	Sigmoid	Sigmoid
Learning rate	0.1	0.1
Maximum number of training sessions	1.000	

To assess the performance of the designed algorithm, it was compared with two other error correction algorithms. One was based on a classification model. This classification model also utilized the LSTM algorithm to predict potential locations of grammatical errors. The predicted results were compared with the original grammar. If they were consistent, it meant the grammar was correct. If there was a mismatch, the prediction result was considered the corrected text. The relevant parameters for the LSTM model used to construct the

classification model are as follows: the text vectorization dimension was also set to 200; there were two hidden layers, each with 128 neurons that employed sigmoid activation functions, and the learning rate was set to 0.1.

The second algorithm also relied on a translation model. This translation model featured a similar encoder-decoder structure as the designed algorithm. The dissimilarity between this algorithm and the one mentioned in this article lies in its exclusion of an attention mechanism for processing intermediate sequences. Therefore, the relevant parameters used in configuring the algorithm presented in this study were used as a reference for setting up the parameters of this translation model-based algorithm.

3.3 Evaluation indicators

The problem of grammatical error correction for English translations can be regarded as a class of text categorization problem, i.e., positive cases that contain grammatical errors and negative cases that do not contain grammatical errors. Based on this, the confusion matrix (Table 2) was used to compute the precision, recall rate, and F value of the error correction algorithm. The relevant equations are

Table 2: Confusion matrix

	Predicted to be a positive case	**Predicted to be a negative case**
Actual is a positive case	TP	FN
Actual is a negative case	FP	TN

$$\begin{cases} P = \dfrac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \\ R = \dfrac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \\ F_\beta = \dfrac{(\beta^2 + 1) \cdot P \cdot R}{\beta^2 \cdot P + R}, \end{cases} \tag{3}$$

where P and R are the precision and recall rate of the error correction algorithm, respectively, F_β is the comprehensive evaluation of P and R, and parameter β represents the weight of R in the comprehensive evaluation. Since the error correction algorithm emphasized more on accuracy, β was set to 0.5.

In addition to the confusion matrix described above, the maximum matching algorithm was also used to evaluate the performance of error correction algorithms. Compared to the confusion matrix, this method is able to calculate the edit distance between the predicted results and the actual results at the phrase level, and it operates using the following equations:

$$\begin{cases} P_{M^2} = \dfrac{\sum_{i=1}^{n} |e_i \cap g_i|}{\sum_{i=1}^{n} |e_i|} \\ R_{M^2} = \dfrac{\sum_{i=1}^{n} |e_i \cap g_i|}{\sum_{i=1}^{n} |g_i|} \\ F_{\beta,M^2} = \dfrac{(\beta^2 + 1) \cdot P_{M^2} \cdot R_{M^2}}{\beta^2 \cdot P_{M^2} + R_{M^2}} \\ e_i \cap g_i = \{e \in e_i | \exists g \in g_i, \mathrm{match}(g, e)\}, \end{cases} \tag{4}$$

where $e_i \cap g_i$ is the maximum match between the predicted result given by the error correction algorithm and the actual result, e_i is the predicted result given by the error correction algorithm, and g_i is the actual error correction result.

3.4 Experimental results

The error correction performance ($F_{0.5}$) of the attention mechanism-combined model is shown in Table 3. It can be seen that using the sigmoid activation function had higher correction performance with the same number of encoder (decoder) neural nodes; with the same activation function, there was higher correction performance when the number of encoder (decoder) neural nodes was 256 (128).

Table 3: $F_{0.5}$ of the attention mechanism-combined model under different parameter settings

Number of neuron nodes in the encoder (decoder)	Relu activation function (%)	Sigmoid activation function (%)	tanh activation function (%)
64 (32)	76.8	84.1	71.7
128 (64)	80.3	89.7	79.6
256 (128)	83.7	92.9	82.9
512 (256)	79.8	88.6	79.7
1024 (512)	71.4	82.9	71.2

The partial error correction results of the three grammatical error correction algorithms are provided in Table 4. Based on the outcomes presented in Table 4, it is evident that the attention mechanism-combined algorithm exhibited a higher level of accuracy in identifying and rectifying grammatical errors within the source text. In contrast, the other two error correction algorithms both demonstrated error misidentification.

The three algorithms were evaluated using a test set derived from the public corpus, and their error correction performance is summarized in Table 5. Data in Table 5 revealed that the attention mechanism-combined algorithm delivered the highest performance whether through the confusion matrix method or the maximum matching method. The traditional translation-based algorithm ranked second, and the classification model-based algorithm performed the least effectively under both evaluation criteria.

The generalization performance of the error correction algorithms was evaluated by testing them with English translations from freshmen, following testing on the public corpus test set. The test results are detailed in Table 6. The attention mechanism-combined algorithm continued to exhibit the best performance, as evidenced by the comparison of performance among the three algorithms in Table 6. The traditional translation-based algorithm ranked second, while the classification model-based algorithm performed the least effectively. However, when comparing these results to those obtained from the public corpus test set, it is observed that the performance of all three algorithms decreased. Among them, the reduction in performance of the attention mechanism-combined algorithm was relatively minimal.

4 Discussion

With the deepening of globalization, there has been an increase in cross-cultural communication. The importance of English as the world's largest language is becoming increasingly prominent. For non-native English speakers, differences in language and cultural environments often lead to grammatical errors during translation. Serious grammar mistakes can directly affect the expression of meaning in sentences, so non-native learners need to pay attention to correcting grammar while learning English. In traditional learning processes, teachers guide students in grammar correction. However, on one hand, the effectiveness of correction is influenced by the teacher's proficiency level; on the other hand, teachers have limited energy and cannot provide one-on-one guidance to all students. The emergence of intelligent algorithms has provided a new way for grammar correction in English translation. This article regards the grammar correction issues in translated texts as a translation problem and corrects incorrect translations to achieve the identification and correction of grammatical errors in translations. The "encoder-decoder" structure of a translation model was applied to

Table 4: Partial error correction results of three error correction algorithms

Source text	这杯奶茶味道很好。	**Source text**	你明天要去跑步吗?
Translation	This cup of milk tea tasted good	Translation	Do you going for a run tomorrow?
Reference correction	This cup of milk tea **tastes** good	Reference correction	**Are** you going for a run tomorrow?
Classification-based error correction	This cup of milk tea **tasting** good	Classification-based error correction	**Did** you going for a run tomorrow?
Translation-based error correction	**A** cup of milk tea tastes good	Translation-based error correction	**Are** you **gone** for a run tomorrow?
Attention mechanism-combined error correction	This cup of milk tea **tastes** good	Attention mechanism-combined error correction	**Are** you going for a run tomorrow?
Source text	我在想你是否能借我一支笔。	**Source text**	你昨天看电影了吗?
Translation	I was wondering if you could lend me an pen.	Translation	Do you see the film yesterday?
Reference correction	I was wondering if you could lend me **a** pen.	Reference correction	**Did** you see the film yesterday?
Classification-based error correction	I **am** wondering if you could lend me an pen.	Classification-based error correction	**Are** you see the film yesterday?
Translation-based error correction	I was **wonder** if you could lend me a pen.	Translation-based error correction	Do you **seen** the film yesterday?
Attention mechanism-combined error correction	I was wondering if you could lend me **a** pen.	Attention mechanism-combined error correction	**Did** you see the film yesterday?

Table 5: Error correction performance of three grammatical error correction algorithms for corpus test set

Evaluation criteria	Confusion matrix method			Maximum matching method		
	P (%)	R (%)	$F_{0.5}$ (%)	P_{M^2} (%)	R_{M^2} (%)	$F_{0.5,M^2}$ (%)
Classification-based	45.3	55.4	47.0	42.6	53.9	44.5
Translation-based	68.9	79.5	70.8	66.8	78.7	68.9
Attention mechanism-combined	92.1	96.3	92.9	90.3	93.6	90.9

Table 6: Error correction performance of three grammatical error correction algorithms on freshmen' English translations

Evaluation criteria	Confusion matrix method			Best-fit method		
	P (%)	R (%)	$F_{0.5}$ (%)	P_{M^2} (%)	R_{M^2} (%)	$F_{0.5,M^2}$ (%)
Classification-based	30.5	31.3	30.7	26.9	29.8	27.4
Translation-based	64.8	75.6	66.7	61.7	72.7	63.6
Attention mechanism-combined	91.3	90.7	91.2	90.8	90.5	90.7

grammar correction in translations. The LSTM algorithm was used in both the encoder and decoder, and the attention mechanism was incorporated. Subsequently, simulation experiments were conducted to compare the improved LSTM algorithm with two other error correction algorithms. The results obtained have been shown above. The LSTM translation model employed in this article, which incorporated an attention mechanism, outperformed the other two algorithms. The reasons behind the aforementioned results were analyzed. The classification model-based algorithm relies on a classification model to predict and correct errors. However, this approach is limited in its ability to predict only specific types of errors and is constrained to specific locations within the text. This restricted scope results in the worst error correction performance. It also deteriorates when applied to English translations by freshmen. The traditional translation-based algorithm leverages machine translation principles to convert potentially erroneous phrases into correct ones without needing to pinpoint error locations. It is not limited by error types, making it superior to the classification model-based algorithm in terms of error correction performance. The attention mechanism-combined algorithm also employs machine translation principles but introduces the attention mechanism to enhance traditional machine translation. This mechanism helps highlight key elements in the intermediate sequence, reducing decoder interference during decoding and minimizing information loss. As a result, this algorithm exhibits even better error correction performance.

5 Conclusion

In summary, this study provides a brief introduction to grammatical error correction algorithms based on an encoder-decoder machine translation structure. The grammatical error correction algorithm was combined with the attention mechanism to enhance the algorithm's performance and conducted simulation experiments to assess its effectiveness. These experiments involved comparing the algorithm with error correction algorithms based on a classification model and a traditional translation model using both a public corpus and English translations from freshmen. The key findings are as follows: (1) In the grammar correction model used in this article, the best error correction performance is achieved when employing a sigmoid activation function and neural nodes of 256 (128) for the encoder (decoder). (2) The attention mechanism-combined algorithm was capable of correcting the source text more accurately. (3) When tested with a public corpus, the attention mechanism-combined algorithm outperformed the others, followed by the traditional translation-based algorithm, and the classification model-based algorithm performed the least effectively. (4) When applied to

English translations written by freshmen, the performance ranking of the three algorithms remained consistent. However, all algorithms experienced a decrease in error correction performance, with the attention mechanism-combined algorithm showing a relatively smaller reduction. The contribution of this article lies in treating the grammar correction issue in English translations as a specific translation problem and correcting the translated text with grammar mistakes into one without any grammatical problems, thereby providing an effective reference for correcting the grammar of translated texts.

Funding information: The author states no funding involved.

Author contributions: Lihua Cai designed research, performed research, analyzed data, and wrote the paper.

Conflict of interest: The author declares no conflict of interests.

Data availability statement: Data will be available on reasonable request.

References

[1] Tolosa C, Ordóñez CL, Alfonso T. Online peer feedback between colombian and New Zealand FL beginners: A comparison and lessons learned. Profile: Issues Teach Prof Dev. 2015;17:73–86.
[2] Rozovskaya A, Roth D. Grammar error correction in morphologically rich languages: The case of Russian. Trans Assoc Comput Linguist. 2019;7:1–17.
[3] Hussein SM. The correlation between error correction and grammar accuracy in second language writing. Int J Psychosoc Rehabilitation. 2020;24:2980–90.
[4] Zhang Y. Application of intelligent grammar error correction system following deep learning algorithm in English teaching. Int J Grid Util Comput. 2022;13:128–37.
[5] Zhou S, Liu W. English grammar error correction algorithm based on classification model. Complexity. 2021;2021:6687337-1–11.
[6] Lin N, Chen B, Lin X, Wattanachote K, Jiang S. A framework for Indonesian grammar error correction. ACM Trans Asian Low-Resour Lang Inf Process. 2021;20:1–12.
[7] Li Y, Anastasopoulos A, Black AW. Towards minimal supervision BERT-based grammar error correction (student abstract). Proc AAAI Conf Artif Intell. 2020;34:13859–60.
[8] Dai Y. An automatic pronunciation error detection and correction mechanism in English teaching based on an improved random forest model. J Electr Computer Enginering. 2022;2022:6011993:1–9.
[9] Wanni M. Research on English grammar recognition system based on combination of genetic algorithm and KNN algorithm. J Intell Fuzzy Syst: Appl Eng Technol. 2020;38:1–12.
[10] Lee JW. A comparison study on EFL learner and teacher perception of grammar instruction and error correction. Engl Teach. 2018;73:139–59.
[11] Barzang E. Fostering EFL learners' grammar achievement using recasts and meta-linguistic awareness error correction feedbacks. J Study Engl Linguist. 2019;8:35–46.
[12] Zhu J, Shi X, Zhang S. Machine learning-based grammar error detection method in english composition. Sci Program. 2021;2021:1–10.
[13] O'Brien J. Consciousness-raising, error correction and proofreading. J Scholarsh Teach & Learn. 2015;15:85–103.
[14] Eckstein G. Grammar correction in the writing centre: Expectations and experiences of monolingual and multilingual writers. Can Mod Lang Rev. 2016;72:1–23.
[15] Solyman A, Wang Z, Tao Q, Elhag AAM, Zhang R, Mahmoud Z. Automatic Arabic grammatical error correction based on expectation-maximization routing and target-bidirectional agreement. Knowl Syst. 2022;241:1–13.
[16] Park C, Yang Y, Lee C, Lim H. Comparison of the evaluation metrics for neural grammatical error correction with overcorrection. IEEE Access. 2020;8:106264–72.
[17] Hos R, Kekec M. Unpacking the discrepancy between learner and teacher beliefs: What should be the role of grammar in language classes? Eur Educ Res J. 2015;4:70–6.
[18] Chen X. Synthetic network and search filter algorithm in english oral duplicate correction map. Complexity. 2021;2021:9960101-1–12.