Statistical Language Model-based Analysis of English Corpora and Literature

Wenwen Chai*
School of Foreign Languages, Zhengzhou Normal University,
Zhengzhou, 450044, China

Abstract—**Despite widespread use of statistical language models in language processing, their ability to process natural languages is not advanced and they struggle to effectively capture linguistic information. Furthermore, there is a lack of automatic processing models in the field of natural language processing. In order to address these issues, and Improve the processing ability of statistical language models for English language a statistical language model optimization algorithm has been proposed. This algorithm is based on an improved resorting algorithm and is specifically applied to process English literary texts. Experimental results indicate that the proposed algorithm outperforms the N-gram algorithm in a majority of texts, with a maximum accuracy improvement of 14.5%. Additionally, in terms of the grammar analysis model, there is a high level of consistency between the model's scoring and the expert manpower scoring, as reflected by a correlation coefficient of 0.7893. This high level of consistency between the grammar analysis model and expert analysis results holds significant importance for the advancement of natural language processing.**

Keywords—Statistical language model; corpus; English literature; reordering; grammatical analysis

I. Introduction

Currently, utilizing automated algorithms to process natural language is one of the important research topics in the fields of corpus and translation. Statistical language models are models that use statistics to calculate the probability distribution of word occurrences in a particular language or context, which users use as a basis for operations and predictions [1]. With the maturity of technologies such as machine translation and speech recognition, statistical language models have become more widely used [2]. However, as a data-driven model, a single statistical language model has limited ability to process natural language and cannot reflect the linguistic features of natural language [3]. Based on the limitations of the statistical language model, various natural language processing algorithms that have applied the model also tend to be far less capable than human analysis [4]. In order to effectively improve the statistical language model's ability to process natural language and to apply it to natural language processing work, a reordering algorithm based on an improved minimum error training method is proposed. The reordering is an optimization technique, which of can optimize the output of statistical language models by reordering the phrases [5]. A grammar analysis algorithm for English literature is proposed based on the reordering algorithm. Overall, this study proposes an English prediction and literary analysis algorithm based on statistical language models. This model aims to effectively enhance the processing ability of statistical language models for English natural language.

This article is divided into seven sections. The second section introduces the research progress in related fields. The third section introduces the construction ideas and process of the model. The fourth section is the display of experimental results. The fifth section is the discussion. The sixth section is the conclusion. Lastly, seventh section discusses the limitations and future work.

II. Review of the Literature

Statistical language models, one of the most important models in the field of natural language processing, have been studied and applied currently. Desai and his team examined the eye and neural activity of forty subjects during reading activities based on statistical language models combined with medical tests and found that the processing cost of low-frequency words was reduced due to contextual cues. The meanings of high-frequency words were more easily accessible and integrated with context [6]. The research results provide results based on human science for the processing of natural language. Teks P led his research team to conduct a machine translation study for Lampung Nyo dialect and compared the approaches based on statistical language models [7]. The project aimed to help student immigrants in Lampung province to translate the Lampung dialect of Nyo through the model and the proposed method was adopted as a working model with an accuracy rate of 59.85%. Sreelekha and Bhattacharyya [8] provided a solution for machine translation of Indian languages where digital resources are scarce by using Indowordnet lexical database to extend statistical language models and evaluate 440 models for 110 pairs of languages for comparison. They found that using lexical database mapping helped to resolve linguistic ambiguities and improve translation quality. Collins et al. [9] provided a framework for processing communication language data based on statistical language models using generalized linear mixed models and Bayesian methods, which, based on the results of the sample analysis, was able to analyze and compare the discourse patterns of children who had experienced traumatic brain injury and typically developing children differences between them. This study has important implications for the field of language processing and the study of childhood brain injury. Ycel et al. [10] used statistical language models to construct a computer-based system for learning foreign language vocabulary. They used specified software to display various card sets constructed using the proposed algorithm and examined the polysemantic correlations between behavioral variables and difficulty levels of different word categories. This study provides an effective method for learning foreign language vocabulary. The author in [2] investigated the specific case of word frequency effects decreasing with age based on word frequency theory in statistical language models and suggested that word frequency effects may occur at different stages of language production. Ge [11] proposed a hybrid research framework combining word frequency analysis from Google Books Ngram Viewer with other analyses in conjunction with statistical language models. aimed at developing a linguistic and cultural concept analysis. Their findings showed a strong correlation between languages in different regions and their cultural concepts, and the frequency of concept words indicated a stronger collectivist culture in China compared to the U.S. Poncelas et al. [12] proposed a feature decay extension algorithm based on a parallel corpus and a statistical language model in order to delve into feature decay algorithm techniques to achieve a better method of training data instance selection. This method can reduce the execution time of FDA and improve the translation quality when multiple computational units are available. This study provides an important reference for improving the performance of machine translation using FDA technology.

A review of recent research related to statistical modeling of language reveals that most of the research in this field focuses on machine translation. In addition, some studies have combined statistical language models with the fields of medicine and sociology. In the field related to statistical language models, there are fewer studies investigating how to improve their ability to recognize natural language, and there is a lack of related applications in the last three years. Based on this gap area, this research focuses on the improvement of statistical language models and their application in the field of natural language recognition.

III. English Corpus Optimization and Literary Analysis based on Statistical Language Models

A. *A Statistical Language Model-Based Algorithm for Reordering English Corpus Output*

Statistical language models calculate the frequency of occurrence of these concepts in a corpus based on the historical data of a given sequence of words and the likelihood of each word in that sequence. Although this technique is currently widely used in areas involving language processing such as speech recognition, and its translation, statistical language models, as a data-driven model, have biases in the estimation of real natural language [13]. This is due to the limitation of data size and data content. Lexical models, N-gram models, and co-occurrence models are all reordering models that have emerged to make statistical language models closer to real natural language [14]. However, the degree of fit of these models to natural language still needs to be optimized. In this study, a reordering method based on minimum error rate training is proposed. Minimum error rate training is a theory applied to the field of machine translation, but it can be improved and applied to this English corpus optimization and literary analysis. In the English to other languages literary analysis scenario, the results of the statistical linguistic model-based translation for a specific utterance are shown in (1).

$$\hat{R} = \arg\max \Pr(R \mid f) \tag{1}$$

In (1) $\hat{R}$ is the output result, f is the original utterance to be processed, and R is the output target language utterance. To obtain the output with the lowest error rate, the log-linear model is used to compute the posterior probability of the sentence pair (R, f) and recalculate the score, i.e., the ranking basis. The calculation procedure is shown in (2).

$$S(R, f) = \Lambda \bullet \Phi(R, f) \tag{2}$$

In (2), $S(R, f)$ is the score, $\Phi(R, f)$ is the feature vector linking the log-linear model and the sentence pairs, and Λ represents the weights of all features. Then the posterior probability can be defined as (3).

$$P(R \mid f) = \frac{\exp(S(R, f))}{\sum_{R'} \exp(S(R', f))} \tag{3}$$

Based on the results of the recalculated scores and the posterior probabilities, the system reorders the candidate results and outputs the new optimal items as shown in (4).

$$\hat{R} = \arg\max \Pr(R \mid f) = \arg\max S(R, f) \tag{4}$$

In the process of minimum error rate training, feature parameter weights need to be tuned and determined. The session first requires giving each parameter an initial value of weight and debugging for individual parameters. The other non-object parameters are treated as constants during debugging. Next proceed to apply the parameter in to other sentences of the corpus. The process is shown in Fig. 1. Fig. 1(a) shows the tuning process of one parameter a on the optimal solution selection of sentence R1. Different parameters take different value intervals corresponding to different optimal solutions. Fig. 1(b) depicts the test results of parameter a corresponding to sentence R1 in other sentences.

After completing this test, all segmentation points are identified and the optimal values of all sentences are found between each segmentation point. The next step is to perform error statistics for the optimal values in each interval, as shown in Fig. 2. Fig. 2 shows the total number of errors statistics for parameter a. As the value of a varies, the total number of errors statistics also fluctuates significantly, with smaller total number of errors representing better results from the statistical language model output. After following this process for all parameters, the whole algorithm is iterated until the error value statistics tend to be stable, which is more desirable.

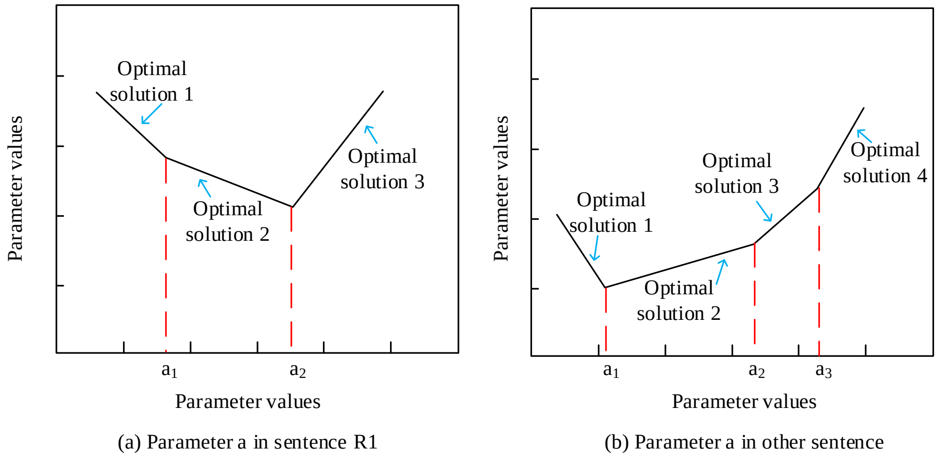


Fig. 1. Adjustment process of feature parameter weights.

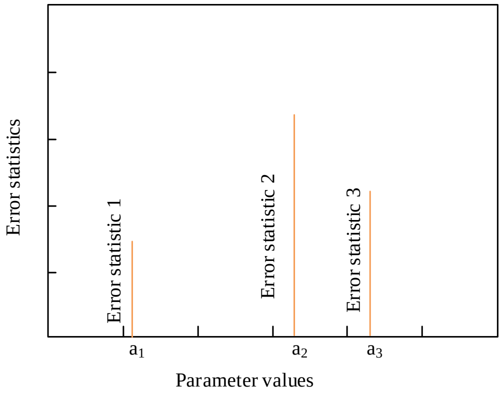


Fig. 2. Count of total error of different parameters.

In order to further enhance the performance of reordering and optimize the results, two sub-models with embedded minimum error rate training are proposed. The sub-models include lexical indication model and lexical N-element co-occurrence model. The lexical indication model performs lexical classification work for the statistical language model. Accurate lexical classification is the basis for the statistical language model to work properly and perform correct literary analysis. There are many possible lexical sequences for a word string, and some of the traditional models directly output the most common lexical properties of words. This method is the most cost-efficient and fast, but the accuracy rate is not satisfactory. To improve this situation, a lexical indication model is considered using a hidden Markov model. The hidden Markov model is shown in Fig. 3, which consists of hidden sequences, observed sequences and different probability distributions. According to the structure of this model, the selection of parameters directly affects the model performance. It has three main parameters, which are noted here as $\lambda = (\pi, a, b)$. The lexical indication task can be analogized to a decoding problem, i.e., finding the optimal sequence of states based on a given word sequence to generate a sequence of observations and a set of parameters. Hidden Markov models can efficiently solve such decoding problems.

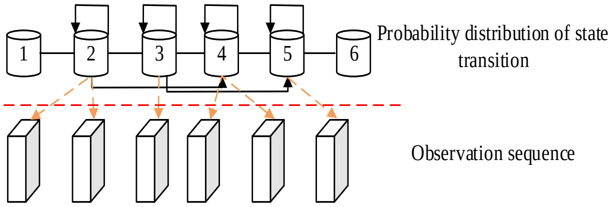


Fig. 3. Hidden markov model.

The lexical N-element co-occurrence model is to integrate lexicality into the traditional word N-element co-occurrence model. The traditional word N meta model calculates the probability distribution by lexicon, while the lexical N meta co-occurrence model calculates it by lexicality, as shown in (5).

$$P(T) = \Pi^{n}_{i=1} p(t_i | t_1, \cdots\cdots, t_{i-1}) \tag{5}$$

In (5), $p(t_i | t_1, \cdots\cdots, t_{i-1})$ represents the lexical N$ probability. t_i represents the different lexical properties. After obtaining the lexical N-probability, we need to deal with the co-occurrence relationship between different words. The co-occurrence is when two words appear together, and the more co-occurrence of two words in the text, the stronger the connection between them. In the lexical N meta co-occurrence model, instead of word-to-word co-occurrence, word-to-word co-occurrence is used, as shown in (6).

$$P(T|W) = \Pi^{n}_{i=1} p(t_i | w_i) \tag{6}$$

In (6), W is a word sequence and T is its corresponding lexical sequence. Correspondingly, the co-occurrence frequencies of words and lexemes are shown in (7).

$$P(W|T) = \Pi^{n}_{i=1} p(w_i | t_i) \tag{7}$$

The two sub-models are embedded in the minimum error training with linear interpolation, and the optimal results are re-output using linear re-ordering. Specifically, when the statistical language model based on minimum error training outputs the ranking results, the two sub-models process the output word order with probability calculation, and then the probability calculation results are linearly interpolated with the ranking results of the statistical language model, as shown in (8).

$$P(W)=c_1p_1(W)+c_2p_2(2W)+c_3p_3(W) \tag{8}$$

In (8), $P(W)$ is the recalculated probability. c_i is the weight of the sub model, and p_i is its probability. This completes the construction of the proposed reordering algorithm, which utilizes two sub-models for optimization and is able to output results that are closer to natural language than the general reordering model.

B. English Grammar Evaluation Model for Literary Analysis

The analysis of English literature has been one of the important application areas of statistical language models [15]. Due to the complexity and variability of natural language, algorithm-based literary analysis has been more difficult [16]. In this study, a grammar evaluation model based on statistical language models is proposed for the grammar evaluation aspect of English literary analysis. The model applies the proposed minimum error training reordering algorithm and incorporates the Transformer structure. The Transformer structure is an encoder-decoder model as shown in Fig. 4. The structure consists of six identical decoders with sub-layers. Each sublayer is connected with a normalization module and residuals between them [17]. There are two types of sub-layers, the fully connected network layer and the attention mechanism layer [18]. The number of layers of encoder and decoder is adjustable under this structure [19]. Considering the cost and computational consumption, the number of layers of both encoder and decoder is set to 6 here.

In written English literature, most of its grammar is fluent and correct, and the problematic ones are usually small. Therefore, the Transformer model is used to move the sentences without grammatical problems directly to the target sentences, thus avoiding the interference of the grammar evaluation model with the sentences without grammatical problems. The mechanism of the probability distribution of words in the target sentence is shown in (9).

$$P_t(W)=a_tp_t^{copy}(w)+p_t^{gen}(w)(1-a_t) \tag{9}$$

In (9), $P_t(W)$ is the lexical probability distribution in the target sentence. p_t^{gen} is the probability distribution of grammar evaluation generation, and p_t^{copy} is the probability distribution of original utterance replication. a_t is the parameter used to control the probability of generation and replication at each time t. The Transformer structure is used in English grammar evaluation in the way shown in Fig. 5. The Transformer model itself is used to generate the probability distribution of the target vocabulary. The replication score is then calculated by the joint determination of the original utterance input this and the implicit state of the target word. The concept of attention mechanism of the Transformer model needs to be introduced here. The attention mechanism solves the problem of interaction, selection and integration between multiple information sources. It enables the model to focus more on the parts of high importance in the operation. Under the attention mechanism, sentences with a higher probability of grammatical problems are given higher weights.

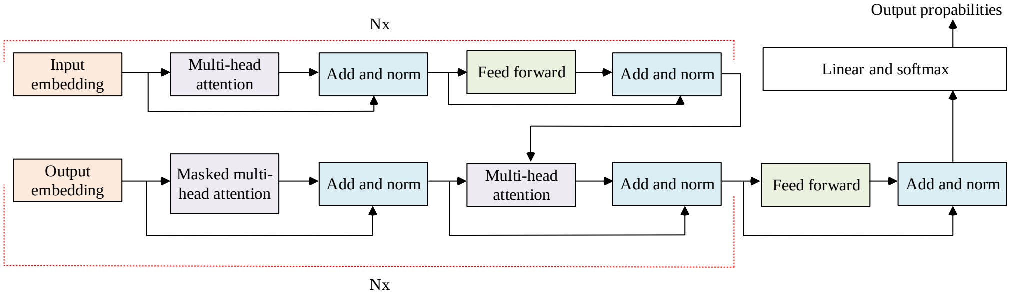


Fig. 4. Transformer structure.

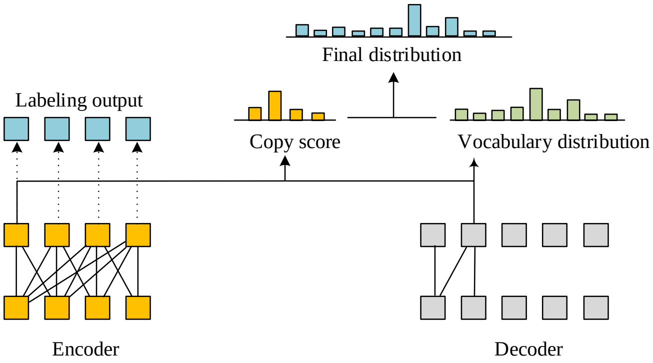


Fig. 5. Transformer structure in grammar evaluation.

Since the Transformer structure alone suffers from the problem of sparse gradients, optimization methods need to be utilized to improve this problem. Here, Adaptive moment estimation (ADAM) optimization is chosen in combination with Transformer structure. This is an adaptive learning rate optimization algorithm that is commonly used to train deep neural networks. The Adam algorithm is derived by combining the advantages of Adagrad and RMSProp algorithms to dynamically adjust the learning rate and track the exponential mean of each parameter and the exponential mean of the squared values. This adaptive learning rate can be automatically adjusted during the training process to ensure that the learning rate is neither too large nor too small, improving the training efficiency and convergence speed. Compared with the traditional gradient descent method, Adam's algorithm has faster convergence speed and higher efficiency, and is widely used in the optimization of various deep learning models. Suppose the objective function is $f(\omega)$, then the gradient of the objective function under Adam's algorithm for the current moment parameters g_t is shown in (10).

$$g_t = \nabla f(\omega_t) \tag{10}$$

After obtaining the gradient, it is also necessary to calculate the data of first-order momentum and second-order momentum in the process, where the solution process of first-order momentum is shown in (11).

$$m_{1t} = \phi(g1, g2, \cdots\cdots, g_t) \tag{11}$$

Equation (11) in m_{1t} is the first-order momentum. The process of solving for second-order momentum is similar to first-order momentum, and the mathematical expression of the process is shown in (12).

$$m_{2t} = \psi(g1, g2, \cdots\cdots, g_t) \tag{12}$$

In (12), the second-order momentum is denoted by m_{2t}. At a particular moment t, the gradient solution process of the algorithm is shown in (13).

$$\mu = l \frac{m_{1t}}{\sqrt{\mu}} \tag{13}$$

In (13), μ represents the gradient. l represents the learning rate of the algorithm. Adam's algorithm also needs to update the parameters, and the mathematical procedure of parameter update is shown in (14).

$$\omega_t = -(\mu_{t-1} - \omega_{t-1}) \tag{14}$$

In (14), ω_t represents the parameters at the time of t. Finally, the functions $\phi(g1, g2, \cdots\cdots, g_t)$ and $\psi(g1, g2, \cdots\cdots, g_t)$ for solving the first- and second-order momentum are defined as shown in (15).

$$\begin{cases} \phi(g1, g2, \cdots\cdots, g_t) = lm_{1,t-1} + (1-l)g_t \\ \psi(g1, g2, \cdots\cdots, g_t) = lm_{2,t-1} + (1-l){g_t}^2 \end{cases} \tag{15}$$

This completes the construction of a grammatical analysis model for English literature based on statistical language models. The complete flowchart of the proposed algorithm can be summarized in the form shown in Fig. 6. The reordering algorithm based on minimum error training is used to adjust the output of the English corpus based on the statistical language model, while the lexical indication model and lexical N-element co-occurrence model are proposed to further optimize the output of the corpus. The proposed reordering algorithm is applied to the English corpus for English literary analysis, and it can analyze the utterances more effectively and make the output results closer to natural language. Applying this feature to English literary grammar analysis, the study combines the improved Transformer structure to propose an English grammar analysis model which can analyze and point out the grammars that may be problematic in English literature. In this part, the Transformer structure improved by Adam is used to process English literary texts and analyze them based on a corpus.

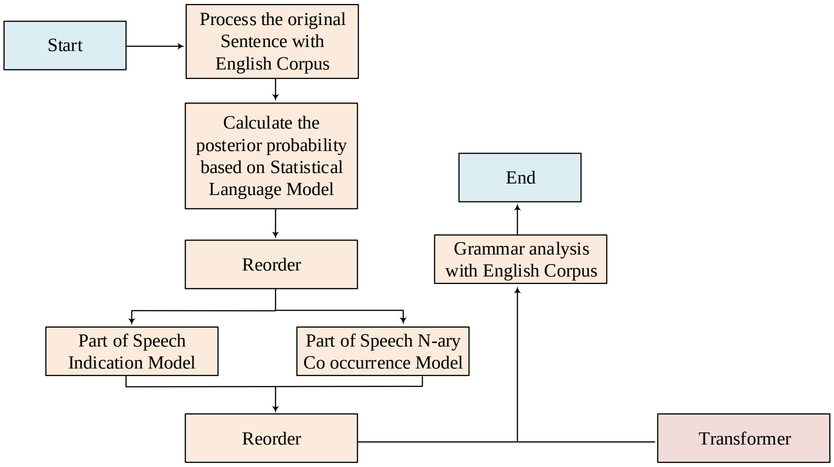

Start
Process the original Sentence with English Corpus
Calculate the posterior probability based on Statistical Language Model
Reorder
Part of Speech Indication Model
Part of Speech N-ary Co occurrence Model
Reorder
Transformer
Grammar analysis with English Corpus
End

Fig. 6. Flow chart of proposed algorithm.

IV. Model Testing and Result Presentation

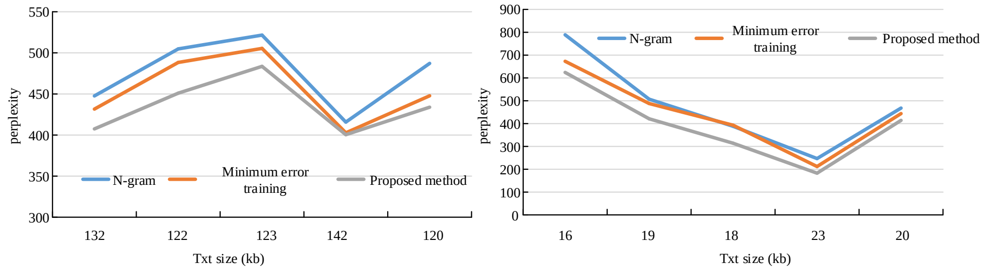
This test focuses on the reordering algorithm for the English corpus and the grammatical analysis model of English literature combined with this algorithm. To ensure that the performance of the algorithm is fully exploited, adequate configurations as well as a large amount of data are required. The various environment configurations and the corpus used for the process of this test are shown in Table I. For system stability reasons, Windows 10 was chosen as the operating environment and Python was used as the programming environment. Four English corpora were selected, namely Gutenberg, Wikitext-103, News crawl 2018, and Tatoeba. The lowest of these databases contained 1. The lowest of these databases contains 1,000,000 statements and the highest contains 4,000,000 statements. The four databases have a total of 10,000,000 statements. The large volume of data eliminates the impact of various special cases in the experiment.

First, we measure the Perplexity of the English corpus based on the proposed reordering technique. Perplexity is an important index to evaluate the performance of linguistic statistical models, which represents the average number of branches of the target text. The reciprocal of Perplexity expresses the average probability of each word. When the language model has low Perplexity, it means that it has high performance. A high degree of Perplexity means that the model selection is more difficult and the performance is lower. Fig. 7 shows the test results of algorithm Perplexity. In order to get comparable results, N-gram algorithm and unimproved minimum error training method are used for comparison. Fig. 7(a) shows the Perplexity of several algorithms in the text with a large amount of data, and Fig. 7(b) shows their performance in the text with a small amount of data. When the test text is a large text with a size of more than 100kb, the Perplexity of several algorithms fluctuates less. Their fluctuation range is between 400 and 550. When the text is a small file of 20kb or less, the Perplexity of several algorithms fluctuates greatly, ranging from 150 to 800. On the whole, the Perplexity of the proposed algorithm is lower than that of the other two algorithms under each TXT text, which shows that the proposed sub algorithm optimization can effectively reorder, thus controlling the complexity of the language model and ensuring the efficiency of the model.

After completing the evaluation of the perplexity, the accuracy of the algorithm output also needs to be evaluated. Since N-gram has been widely used in related fields, N-gram is directly used here as a comparison object. Fig. 8 shows the results of comparing the output accuracy of the proposed reordering algorithm with N-gram. The curves in the Fig. 8 indicate the difference in accuracy between the two on the same text, and positive values indicate that the accuracy of the proposed algorithm is higher than that of N-gram, while negative values indicate the opposite from the overall view of the curves. The majority of the accuracy curves are above 0, i.e., the proposed algorithm is more accurate than N-gram for most of the texts. The proposed algorithm is up to 14.5% more accurate than N-gram. In the few texts where its accuracy is lower than N-gram, its accuracy is no less than 5% of N-gram. A larger sample size eliminates accidental phenomena, so based on the results, although both perform negatively when dealing with different texts, the proposed algorithm has a higher reordering ability than the widely used N-gram algorithm in terms of accuracy.

TABLE I. Test Environment Configuration and Corpus Selection

Item	Detail	
CPU	i5-13400f	
Memory	32 GB	
Operative System	Windows 10	
Programming Environment	Python	
Corpus	Wikitext-103	3,000,000 sentences
	Tatoeba	1,000,000 sentences
	Gutenberg	4,000,000 sentences
	News crawl 2018	2,000,000 sentences



(a) Large txt size

(b) Small txt size

Fig. 7. The degree of confusion of the algorithm in different environments.

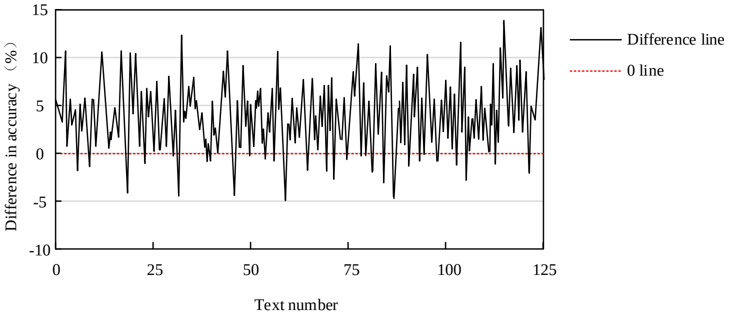


Fig. 8. Accuracy difference between the proposed algorithm and N-gram.

In addition to the accuracy, the accuracy, recall and F0.5 values of different algorithms were also compared on the dataset and the results are shown in Table II. This test was performed on the accuracy, recall and F0.5 values of each N-gram, minimum error training and the proposed reordering algorithm. The tests were done on each of the four datasets to ensure the comprehensiveness of the results. On the Gutenberg dataset, the proposed algorithm has a precision rate of 57.98 and accuracy and F0.5 values of 25.68 and 52.23, which are higher than the other two algorithms in these three dimensions. Combining the test results on the four datasets, the proposed algorithm has the highest accuracy rate of 62.13, the highest recall rate of 37.43, and the highest F0.5 value of 54.32. The proposed algorithm consistently outperforms the N-gram and the minimum error rate training methods in several dimensions of accuracy rate, recall rate, and F0.5 value, both in terms of individual dataset comparisons and in terms of the dataset as a whole.

After completing the analysis of the proposed reordering algorithm, the testing of the English literary grammar analysis algorithm based on this algorithm is continued. Since grammatical analysis mainly deals with natural language, human analysis from experts is currently the most correct way for natural language processing. Therefore, 750 texts were selected for the test and the results of human analysis from experts were compared with the results of the algorithm, and the results are shown in Fig. 9. The horizontal coordinates in this Fig. 9 represent the different texts and the vertical coordinates represent the scores of the two methods for the grammar. Looking at the overall distribution of scores, we can see that the distribution of scores scored by the algorithm is more concentrated than that scored by the expert human, but in general there is a certain correspondence. The expert scores are concentrated in the range of 98 to 75, while the algorithmic scores are concentrated in the range of 75 to 87. The mean score of expert scoring was 85.15 and the mean score of algorithmic scoring was 84.27. The correlation analysis of the results showed that the correlation coefficient of the two scoring methods was 0.7893, which means that there is a significant correlation between them. The change of the results indicates that the proposed English grammar analysis algorithm is somewhat synchronized with the results of the human analysis, and therefore its correctness is to some extent trustworthy.

TABLE II. COMPARISON RESULTS OF ALGORITHM PERFORMANCE

Corpus	Algorithms	Precision	Recall	F0.5
Wikitext-103	Proposed	66.54	37.43	38.42
	Minimum error traning	60.78	32.84	33.43
	N-gram	57.31	30.11	29.75
Tatoeba	Proposed	60.84	23.52	54.32
	Minimum error traning	53.13	20.18	43.81
	N-gram	51.27	18.60	41.58
Gutenberg	Proposed	57.98	25.68	52.23
	Minimum error traning	50.55	21.14	47.64
	N-gram	47.83	18.93	42.41
News crawl 2018	Proposed	62.13	27.61	46.58
	Minimum error traning	57.64	24.33	40.77
	N-gram	55.53	20.58	36.12

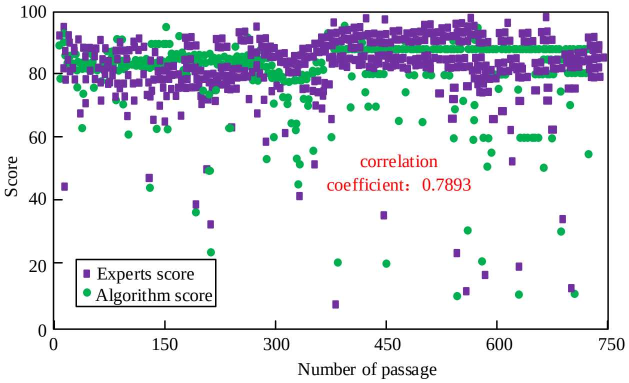
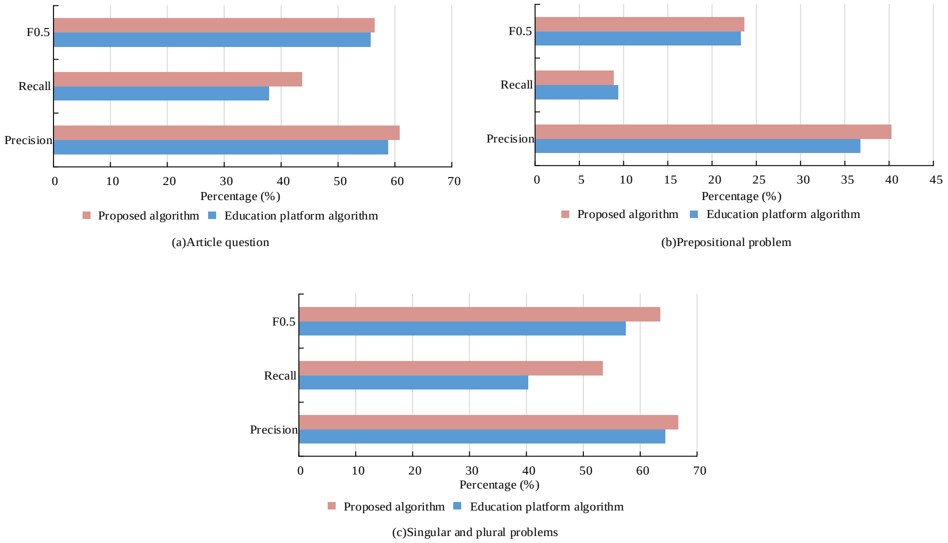


Fig. 9. Comparison results of algorithm and manual analysis.

There are currently grammar analysis algorithms being applied, and to confirm the superiority of the proposed algorithms compared to existing algorithms, a certain online teaching platform's analysis algorithm was used as the comparison object. Three common types of grammar problems were used as the comparison objects: article questions, prepositional problems, and singular and plural problems. The test results are shown in Fig. 10, where Fig. 10(a) shows the comparison results of article questions, Fig. 10(b) shows the comparison results of prepositional problems, and Fig. 10(c) shows the comparison results of singular and plural problems. Compared with the education platform algorithm, the proposed algorithm is superior in precision, recall, and F0.5 in all three dimensions. The proposed algorithm achieved a precision rate of 64.37%, a recall rate of 40.32%, and an F0.5 value of 57.51% in singular and plural problems. For article questions, the precision rate of the proposed algorithm reached 60.79%, while the education platform algorithm only reached 58.82%. Through a comprehensive analysis of the comparison results, it can be seen that the proposed grammar analysis algorithm has a stable advantage over existing algorithms.



(a)Article question

(b)Prepositional problem

(c)Singular and plural problems

Fig. 10. Grammar problem test results.

V. Discussion

The proposed model is an English natural language analysis model based on an English corpus, designed to analyze English corpora and literature. The reordering algorithm based on Minimum Error Training is employed to adjust the output of the English corpus using statistical language models. Additionally, the introduction of the part-of-speech indicator model and part-of-speech n-gram co-occurrence model further enhances the optimization of the corpus output. When applied to English literary analysis using the proposed reordering algorithm, it facilitates more effective sentence analysis, resulting in output that closely aligns with natural language. By incorporating this feature into English literary grammar analysis, a research study proposes an improved Transformer-based English grammar analysis model to identify potential grammar issues in English literary works. In this study, an enhanced Transformer structure, utilizing improvements from Adam optimization, is utilized to process English literary texts and perform analysis based on the corpus.

In the results display section, multiple datasets were used to compare the proposed model with other similar models. The reason for using multiple datasets is that this comparison method can to some extent eliminate randomness and increase the reliability of experimental results. According to the experimental results, the proposed model has the highest accuracy of 62.13, the highest recall rate of 37.43, and the highest F0.5 value of 54.32 on the four datasets used. From these indicators, the proposed model has stable advantages compared to similar algorithms. Due to the fact that manual analysis by humans is currently difficult for machines to replace in the field of natural language analysis, the results of expert human analysis are also entered here and compared with the results of algorithm analysis. After conducting correlation analysis on the statistical results, it was found that the correlation coefficient between the two analysis methods was 0.7893, indicating that the results of algorithm analysis and manual analysis are to some extent similar. This means that the proposed model is to some extent close to people's processing ability of English literature and natural language.

VI. Conclusion

Aiming at the optimization of current statistical language models and English corpora, as well as the gaps in automatic algorithms in the field of English literature analysis, this research proposes an improved re-sorting algorithm based on the minimum error rate training. Based on the re-sorting algorithm, a grammar analysis model for English literature is also proposed. The test results show that in the vast majority of texts, the accuracy of the proposed algorithm is higher than that of the N-gram algorithm. The proposed algorithm has a maximum accuracy of 14.5% higher than N-gram. In a small portion of text with accuracy lower than N-gram, its accuracy is not less than 5% of N-gram. On the Gutenberg dataset, the accuracy of the proposed algorithm is 57.98, with accuracy and F0.5 values of 25.68 and 52.23, which are higher than the other two comparative algorithms in these three dimensions. In addition, in terms of grammar analysis models, the correlation coefficient between model scoring and expert manpower scoring results is 0.7893, indicating a significant correlation between the two. On Singular and plural problems, the accuracy of the model's scoring reached 64.37, the recall rate was 40.32, and the F0.5 value was 57.51, all higher than existing grammar analysis models. The results show that the proposed model has considerable application potential in the field of English literature analysis.

VII. Limitations and Future Work

This study has made certain contributions to relevant fields, but the research results still have limitations. The proposed algorithm is greatly influenced by the size of the text. When the text is too small, there will be significant fluctuations in the performance of the model. How to maintain stable performance of algorithms at any text size is the direction of future work. In addition, this study did not focus on the consumption of algorithms, so it is necessary to evaluate this aspect in future work to determine its practical value.

References

[1] Fang H, Shi H, and Zhang J, “Heuristic bilingual graph corpus network to improve English instruction methodology based on statistical translation approach,” Transactions on Asian and Low-Resource Language Information Processing, vol. 20, no. 3, pp. 304-318, 2021.

[2] Zhang L, and Xuan B, “Neural mechanisms and time course of the age-related word frequency effect in language production,” Advances in Psychological Science, vol. 30, no. 2, pp. 333-342, 2022.

[3] Niesen M, Vander Ghinst M, Bourguignon M, et al. “Tracking the effects of top-down attention on word discrimination using frequency-tagged neuromagnetic responses,” Journal of Cognitive Neuroscience, vol. 32, no. 5, pp. 877-888, 2020.

[4] Wei Z, and Zhang X, “A filtering algorithm of main word frequency for online commodity subject classification in e-commerce,” International Journal of Circuits, vol. 15, no. 1, pp. 218-224, 2021.

[5] Chaouch-Orozco A, Alonso J G, and Rothman J, “Individual differences in bilingual word recognition: the role of experiential factors and word frequency in cross-language lexical priming,” Applied Psycholinguistics, vol. 42, no. 2, pp. 447-474, 2020.

[6] Desai RH, Choi W, and Henderson JM. “Word frequency effects in naturalistic reading,” Language, cognition and neuroscience, vol. 35, no. 5, pp. 583-594, 2020.

[7] Teks P, Lampung B, Nyo D, et al. “Translation of the Lampung language text dialect of Nyo into the Indonesian language with DMT and SMT approach,” INTENSIF Jurnal Ilmiah Penelitian dan Penerapan Teknologi Sistem Informasi, vol. 5, no. 1, pp. 58-71, 2021.

[8] Sreelekha S, and Bhattacharyya P, “Indowordnet's help in Indian language machine translation,” AI & SOCIETY, vol. 35, no. 1, pp. 689-698, 2020.

[9] Collins G, Lundine J P, and Kaizar E, “Bayesian generalized linear mixed-model analysis of language samples: detecting patterns in expository and narrative discourse of adolescents with traumatic brain injury,” Journal of Speech Language and Hearing Research, vol. 64, no. 4, pp. 1256-1270, 2021.

[10] Ycel Z, Supitayakul P, Monden A, et al. “An Algorithm for Automatic Collation of Vocabulary Decks Based on Word Frequency,”. IEICE Transactions on Information and Systems, vol. 103, no. 8, pp. 1865-1874, 2020.

[11] Ge Y, “The linguocultural concept based on word frequency: correlation, differentiation, and cross-cultural comparison,” Interdisciplinary Science Reviews: ISR, vol. 47, no. 1, pp. 3-17, 2022.

[12] Poncelas A, Wenniger G, and Way A, “Improved feature decay algorithms for statistical machine translation,” Natural Language Engineering, vol. 28, no. 1, pp. 71-91, 2020.

[13] Atici, Ramazan, Pala, et al. “Prediction of the ionospheric foF2 parameter using R Language forecasthybrid model library convenient time series functions,” vol. 122, no. 4, pp. 3293-3312, 2022.

[14] Guirong B, Shizhu H, Kang L, and Jun Z, “Using Pre-trained Language Model to Enhance Active Learning for Sentence Matching,” ACM transactions on Asian and Low-Resource Language Information Processing, vol. 21, no. 2, pp. 19, 2022.

[15] Lee O, “Statistical learning and language: English RCs and number agreement,” Studies in Linguistics, vol. 58, pp. 251-274, 2021.

[16] Boussakssou M, Ezzikouri H, and Erritali M, “Chatbot in Arabic language using seq to seq model,” vol. 81, no. 2, pp. 2859-2871, 2022.

[17] Ming Y, and Yi P, “Meta-learning for compressed language model: A multiple choice question answering study,” Neurocomputing, vol. 487, pp. 181-189, 2022.

[18] Ivan F, Alexey Z, Pavel B, Ekaterina D, Nikita K, Andrey K, Ekaterina A, Evgenia K, and Evgeny B, “A differentiable language model adversarial attack on text classifiers,” vol. 10, pp. 17966-17976, 2022.

[19] Gaeta L, and Brydges C, “An examination of effect sizes and statistical power in speech, language, and hearing research,” Journal of speech, language, and hearing research: JSLHR, vol. 63, no. 5, pp. 1572-1580, 2020.