Hindawi
Scientific Programming
Volume 2022, Article ID 3764230, 7 pages
https://doi.org/10.1155/2022/3764230

Research Article

Research on the Semantic Analysis Method of Translation Corpus Based on Natural Language Processing

Xin Yue

Northeast Agricultural University, Harbin 150030, China

Correspondence should be addressed to Xin Yue; xinyue@neau.edu.cn

Received 8 April 2022; Revised 5 June 2022; Accepted 10 June 2022; Published 31 July 2022

Academic Editor: Sheng Bin

Accurate recognition and analysis of semantics is the most important research field in the process of English translation with the help of natural language processing technology. This paper proposes an English semantic analysis method based on the neural network. First, the idea of model transfer is used to construct a topic segmentation model and the topic granularity segmentation of the translated text is carried out. Then, in order to obtain all the information in the English text, the recursive neural network is selected to recognize the word model. In order to recognize English texts with different sentence patterns, the long-term and short-term memory network is selected to extract the useful information of the text. Through the experimental data measurement and analysis results, compared with the traditional sentence analysis methods, the accuracy of the proposed method is as high as 95.8% and the model occupies less hardware resources.

1. Introduction

With the update and development of information technology, the design of the machine translation system based on a computer-integrated information processing method has significantly improved the intelligence and accuracy of English translation. Machine translation refers to the process of using computers to automatically transform a natural language into another natural language with exactly the same meaning [1]. Machine translation is an important problem in the field of artificial intelligence and a major task in natural language processing. It is closely related to the core theories of computational linguistics, such as syntactic analysis, semantic understanding, and natural language generation. However, due to the complexity and diversity of human natural language itself, the understanding of natural language by existing computer systems is still at a relatively low stage. There are still a lot of problems to be solved in order to properly translate the rich natural language phenomena. At the same time, as a new machine learning method, deep learning can automatically learn the abstract feature representation and establish the complex mapping relationship between input and output signals, which provides a new idea for the research of statistical machine translation. This paper will explore the method of neural network learning and study several key problems of the English semantic analysis method based on neural networks.

Machine translation systems can be divided into rule-based machine translation (RBMT), example-based machine translation (EBMT), and statistical machine translation (SMT) [2, 3]. The mainstream method of early machine translation was RBMT. The rule-based machine translation system relies on the manually compiled bilingual dictionary and various forms of translation rules summarized by experts. The computer is used to decode the dictionary and translation rules and translate the sentences in the source language into the target language. The RBMT does not need training and has low requirements for computer performance. However, the collection of bilingual dictionaries and the summary of translation rules are very difficult and complex, which requires a lot of expert knowledge. With the development of computer technology, there are a large number of text corpora that are helpful for translation. The rule-based system is difficult to effectively use new resources to automatically improve the performance of the translation system. Therefore, the rule-based machine translation method is gradually replaced by new methods. In order to automatically learn translation knowledge from text corpus, Somers [4] first proposed the example-based machine translation method. This method starts from the existing translation experience and knowledge and translates the new source language sentences through the principles of analogy and so on. In this method, the source language sentences are segmented into phrase fragments seen in the translation knowledge, and then, the obtained phrase fragments are matched with the empirical knowledge through analogy and other methods to obtain the translation of phrase fragments, and then, the translated phrase fragments are spliced into the target language sentences. After EBMT, Hua and Wang [5] further proposed the statistical machine translation method, which regards the correspondence between source language and target language as a probability problem. In this method, any target language sentence is regarded as a possible translation candidate of the source language sentence, but the probability of different candidates is different. The core problem of statistical machine translation is to use statistical methods to automatically learn the translation model from the corpus, and then based on this translation model, to find a target language sentence with the highest score for the input source language sentence as the translation result. The machine translation system based on statistics is obviously superior to the other two methods in robustness and scalability. It can naturally deal with the ambiguity of language, quickly build a high-performance translation system from the existing corpus, and automatically improve the translation performance when the corpus is increased. However, in practical applications, English character recognition is easily disturbed by noise, which seriously affects the efficiency of English semantic recognition.

Translation models can be divided into the following three categories according to different basic translation units and modelling methods: word-based translation, phrase-based translation, and syntax-based translation [6–8]. The word-based translation model takes the translated word pair as the basic translation unit. Due to the large ambiguity and obvious order adjustment in vocabulary level translation, and the lack of context information in vocabulary translation model, these factors make it difficult for vocabulary based translation model to achieve good results. The phrase-based translation model takes the translation phrase pair as the basic translation unit and directly records the common translation corresponding patterns and local ordering information in the translation rules. The translation model needs to extract phrase translation rules from bilingual data. A basic method is the extraction method based on the alignment template [9]. The basic idea of this method is to extract all translation phrase rules consistent with word alignment according to the alignment results of bilingual data. The probability estimation of phrase-based translation rules generally adopts count based maximum likelihood estimation [10, 11]. The traditional neural network is trained mainly through the back propagation of errors. However, due to the complex characteristics of English character images, the convergence is poor in the process of network training and it is easy to fall into the local optimal solution. Therefore, the generalization performance of the network is poor when the test set is used, which affects the recognition accuracy of the network.

The main contributions of the research are as follows:

(1) A new English semantic analysis method based on a neural network is proposed. The new method can greatly improve the efficiency of English semantic recognition.

(2) Long-term and short-term memory networks are used to extract the useful information of the text. In this paper, the gradient descent method is used for network training to enhance the convergence performance of the network and improve the training speed.

2. A Topic Segmentation Model Based on Model Transfer

Text segmentation refers to the process of extracting coherent text blocks from text. In terms of segmentation granularity, text segmentation tasks can be divided into topic segmentation tasks and basic discourse unit segmentation tasks. Topic segmentation refers to the text division of a paragraph of text according to the topic semantic information and each topic is continuous. Generally speaking, the text is composed of different topics, and the text is segmented by topic granularity through technical means. The task of basic discourse unit segmentation is to divide the sentences in the text into basic discourse units.

In terms of task requirements, natural language processing tasks such as emotional feature extraction, semantic mining, viewpoint mining, topic segmentation, language recognition, and dialogue system are inseparable from the text processing process based on the text segmentation task. Suppose the current text $T = \{s_1, s_2, \ldots, s_n\}$, s represents the i-th sentence of the current text T. Taking topic recognition as the task, the challenge of how to recognize the topic in text T and make $T = \{\text{topic}_1, \text{topic}_2, \ldots, \text{topic}_m | m \leq n\}$ has become the most basic task of text segmentation task.

This paper proposes a topic segmentation method with a transformer as the auxiliary of coding structure and sentence coherence. This method makes full use of the advantages of transformer encoder's multihead attention mechanism, residual mechanism, and parallel computing. The data processing flow of the proposed method is as follows: First, the text enters the word level transformer encoder with a vector representation and position coding of word granularity to obtain the sentence representation, and the sentence representation of the text is input into the sentence level transformer encoder to obtain the sentence coding. Then, the text sentences are classified by the classifier to judge whether the sentence is the topic boundary. The word set encoder would generate the text representation according to the word coding of the text, get the final text representation through the sentence encoder, and get the coherence score of the text through a regression model.

The bidirectional coding representation model based on the transformer makes the text data better represented. Specifically, in sentence $S = \{t_1, t_2, \ldots, t_n\}$, $t_i\,(1 \leq i \leq n)$ represents the i-th word of the current sentence S; each word in the sentence will be represented by the vector of the word w_i through the pretraining model. The sentence coding method is represented by the average value of the sum of 9 to 12 layers of coding.

$$w_i = \mathrm{BET}^{\text{represent}}(t_i), \tag{1}$$

where w_i is the sentence coding value.

$$\text{represent} = \frac{1}{3}\sum_k \text{layer}_k, \tag{2}$$

where represent is the specific code of the current i-th word t_i and layer$_k$ represents the coding of k-th layer.

The topic decision model is used to segment the composition data in topic granularity. The topic decision model takes the last sentence of each topic in the text as the topic cut-off point, the sentence in the topic is the topic coherent sentence, and whether the sentence in the text is the cut-off point is the basis for topic division. Based on the sequence of input data, the topic decision model uses LSTM-Attention model structure to judge the input sentence pairs. In order to make the model learn the influence of key words on coherence, the word granularity is used as the input of the model and the attention mechanism is used to give weight to the input.

Input data sentence team $\text{sentence}_i, \text{sentence}_{i+1} = [w_1, w_2, \ldots, w_m]$. The model first encodes the input at each time as shown in (3):

$$o_t = B_{\text{LSTM}}(w_t), \tag{3}$$

where o_t represents the code of input w_t at the current t time.

The attention mechanism is used to calculate the weight of the output of each time point of the model as shown in equations (4) and (5):

$$y_t = \tanh(W_w.O_t + b_w). \tag{4}$$

$$a_t = \frac{\exp\left(y_t^T . y_w\right)}{\sum_t \exp\left(y_t^T . y_w\right)}, \tag{5}$$

where y_t, W_w, and b_w are attention mechanism layer parameters and a_t represents the weight of the sequence input at the t-th time point in the whole input. Therefore, through the attention mechanism layer, the input vector v_t with a weight expression can be obtained; the calculation formula is shown in (6).

$$v_t = a_t.o_t. \tag{6}$$

After the vector representation with lexical weight is calculated through using the attention mechanism, it is spliced and input into a dimension reduction full connection layer and then classified by using the SoftMax function [12]. The spliced vector is v' and the calculation formulas are shown in equations (7) to (9):

$$v = \text{con}(v_1, v_2, \ldots, v_m), \tag{7}$$

$$v' = \text{fc}(v), \tag{8}$$

$$\widetilde{Y} = \text{soft}\max(v'.W_s + b_s), \tag{9}$$

where con represents splicing of vectors, v represents the input representation, $\widetilde{Y}$ represents the final classification results, W_s and b_s are the network parameters of current classification layer, and v' represents input representation after dimensionality reduction.

The semantic vectorization of the translated text topic and text topic is the key to accurately obtain the similarity between them and verify the method in this paper. After semantic vectorization, topics before and after translation are projected into the same semantic space, and the distance between them indicates their semantic similarity. By calculating the distance between the two vectors, the similarity between them can be obtained.

α − ave score calculation method is used to obtain the final relevance score, where α represents the reward factor and ave represents the mean value thought. The translated text $\text{TX} = \{\text{topic}_1, \text{topic}_2, \ldots, \text{topic}_n\}$ consists of n topics; $\text{topic}_i\,(1 \le i \le n)$ represents the i-th topic of the translated text. The semantic similarity between each topic vector topic$_i$ and the topic vector Prompt of the translated text is calculated. The calculation formula of fitness *Score* is defined as follows:

$$\begin{aligned}
\text{Score} &= \frac{1}{n+2}\left(\left(\sum_{i=1}^{n}\text{sim}^i\right) + \varepsilon + \text{sim}^E\right),\\
\varepsilon &= \frac{\left(1 - e^{-n}\right)}{\left(1 + e^{-n}\right)},\\
\text{Sim}^i\left(\text{topic}^i, \text{Prompt}\right) &= \frac{\text{topic}^i, \text{Prompt}}{\left|\text{topic}^i\right||\text{Prompt}|},\\
\text{sim}^E(\text{TX}, \text{Prompt}) &= \frac{\text{Tx}, \text{Prompt}}{|\text{TX}||\text{Prompt}|},
\end{aligned} \tag{10}$$

where Sim^i represents the relevance between the current topic and the topic of the translated text and sim^E represents the relevance of the whole text to the topic of the translated text. ε represents the reward factor calculated by relevance which is used to reward the number of topics in the translated text.The more topics, the higher the reward score. ε as the number of topics increases, it will gradually approach a limit value of 1 to avoid the impact of the number of extreme topics in the translated text. The calculated similarity scores of all topics in the translated text are averaged with the results of ε and sim^E, so that the number of topics in the translated text can be rewarded.

3. Semantic Analysis Model Based on LSTM-RNN

The topological relationship between words is very important for text semantic analysis. The language model based on the recurrent neural network (RNN) is more suitable for processing text sequence data.

3.1. Translation Preordering Model Based on the Neural Network. The word order of one language is often very different from that of another language. Dealing with the difference of word order between different languages is called the ordering problem of machine translation, which is one of the main problems in statistical machine translation research.

Preordering refers to the transformation of the input source language sentences into an order similar to the target language through preprocessing before translation and decoding. On the one hand, the preordering method can effectively use the lexical and syntactic information of the source language to help solve the ordering problem; on the other hand, it retains the simplicity of phrase-based translation system decoding. In this paper, a preordering model based on the neural network is proposed. The low dimensional vector representation of arbitrary ordering features is learned through neural networks, so that the model has better generalization ability.

Given a source language sentence and a translation model, the preordering model transforms the source language sentence into a sentence similar to the word order of the target language, and then, the translation model transforms the ordered source language into a target language sentence.

A and B represent the set of sentences in source language and target language, respectively, and the nonprobabilistic preordering model H and translation model T can be expressed as follows:

$$\begin{aligned} H&: A \times A \longrightarrow \mathscr{R}, \\ T&: A \times B \longrightarrow \mathscr{R}. \end{aligned} \tag{11}$$

The score given by the translation system with preordering for a bilingual sentence pair $(\boldsymbol{a}, \boldsymbol{b})$ is as follows:

$$s(\mathbf{a}, \mathbf{b}) = \max_{\mathbf{a}'} \left(H(\mathbf{a}, \mathbf{a}') + T(\mathbf{a}', \mathbf{b}) \right), \tag{12}$$

where $\mathbf{a}' \in A$ represents any replacement of $\mathbf{a}$.

Since the possible source language replacement has an exponential relationship with the sentence length, we first find the N replacements with the highest score in practice. In general, because the preordering model and the translation model are trained separately, we generally ignore the score of the preordering model.

$$s(\mathbf{a}, \mathbf{b}) = \max_{\mathbf{a}' \in \mathbf{N}(\mathbf{a})} T(\mathbf{a}', \mathbf{b}), \tag{13}$$

where $\mathbf{N}(\mathbf{a})$ represents replacement of N highest scores given by the preordering model.

In preordering, the basic unit of source language ordering can be nonsyntactic units such as words and phrases, or syntactic units such as subtrees in syntactic trees. In the simplest case, vocabulary can be used as the basic unit of preordering:

$$\begin{aligned} \mathbf{a} &= w_1 w_2 \ldots w_n, \\ \mathbf{a}' &= \pi(w_1, w_2, \ldots, w_n), \end{aligned} \tag{14}$$

where w_i represents a word in the source language and π represents any replacement of a sequence. Vocabulary is taken as the basic unit of preordering is the freest. The source language can be adjusted into any word order, which can express the most word order of the target language.

We can also first cut the source language into phrases and then preorder at the phrase level:

$$\begin{aligned} \mathbf{a} &= p_1 p_2 \cdots p_n, \\ \mathbf{a}' &= \pi(p_1, p_2, \ldots, p_n), \end{aligned} \tag{15}$$

where p_i represents phrases in source language sentences.

Compared with vocabulary preordering, the granularity of phrase ordering is coarser. Because the local ordering information has been recorded in the phrase table in the phrase-based translation model, it is more appropriate to take the phrase as the basic unit for preordering.

In addition, we can also define the preordering basic unit as the node of the source language syntax tree. In this way, we assume that the source language syntax tree T_e is given, and all nodes in the tree are marked as $t_1, \ldots, t_m$. For one of the tree nodes t and all its children $c(t) = c_1, \ldots, c_k$, these child nodes are reordered as follows:

$$c(t)' = \pi(c_1, c_2, \ldots, c_k). \tag{16}$$

Each tree node of tree T_e is ordered separately and we get the syntax tree T_e' after ordering. Through the preorder traversal of T_e', the source language reordering guided by the syntax tree is obtained. Generally speaking, because there are no phrases as tree nodes, the structure of the dependency tree is relatively flat, and each tree node has more child nodes, which can cover more language ordering phenomena. The height of the phrase structure tree is high, there are fewer sub-nodes of each node, which have strong restrictions on ordering, but they also provide more information.

3.2. Design of Semantic Analysis Model. The topological relationship between words is very important for text emotion analysis. The language model based on the recurrent neural network (RNN) is more suitable for processing text sequence data. A RNN consists of three modules: input layer, hidden layer, and output layer [13]. In the RNN model, the time input layer and time hidden layer are summarized into a new input layer, which is also used as the hidden layer at time t. The loop structure of the RNN enables the hidden layer to retain all the information in the previous words, so as to improve the ability to recognize the sequential relationship between words. There are too many expanded state layers in the back propagation of the RNN model through time optimization algorithm [14, 15], which will lead to the gradient attenuation of historical information during training. In this paper, LSTM is used to avoid the long-term dependence of the model on words. Its structure is shown in Figure 1.

As shown in Figure 1, LSTM is a chain form of repeated neural network modules, but the repeated modules have a different structure. Unlike a single neural network layer, there are four, which interact in a very special way. It relies on some gate structures to make information selectively affect the state of each time in the recurrent neural network.

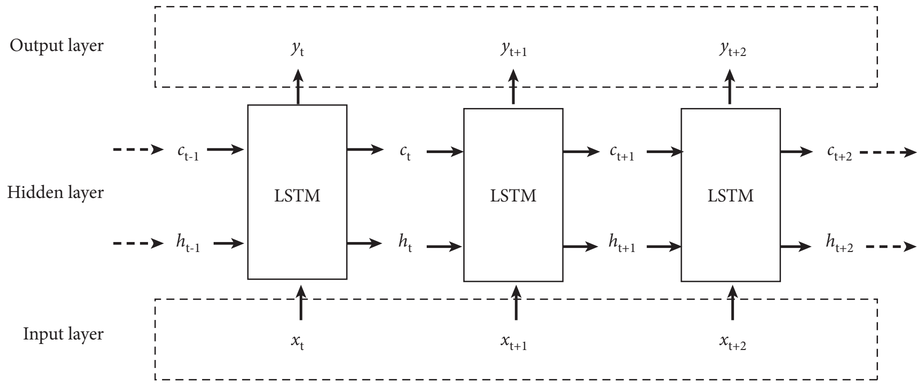

FIGURE 1: Structure of the semantic analysis model based on LSTM.

The calculation process of LSTM mainly includes four steps: (1) calculate the value of forget gate and input gate; (2) update the status of the LSTM unit; (3) calculate the value of the output gate; and (4) update the output of the entire unit.

The RNN with LSTM can be regarded as an improved model of the traditional RNN language model, which takes text statements as input sequences to calculate the error of each model. However, when the text sequence information is long, the RNN model with LSTM can effectively overcome the problem of sequence information attenuation. For English sentences, we first apply the word segmentation standard to convert the sentences into word segmentation. Then, the LSTM is forward calculated, the word segmentation in the sentence is searched from left to right, and the word sequence probability of the word before the probability time t is output. Finally, the error value of the sentence is measured by the joint distribution probability of all words. A higher joint distribution probability can effectively reduce the error value of the text sentence.

In the training stage, the training data are divided into multiple categories according to their emotional labels. For each kind of data, the LSTM model is trained, respectively, and multiple LSTM models are generated. Each LSTM model is used for the corresponding emotional comments. In order to predict the emotional bias of new input comments, the LSTM model obtained in the training stage will be evaluated on the new input comments and give the detection error value. The model with the smallest error value is designated as the emotion category of new input comments. Compared with the traditional RNN language model, the RNN with LSTM can completely cover longer sentences. It performs well in many verification experiments, especially for English sentence structures with connectives.

LSTM has advantages in processing sequence data, but it is limited by the problem of unable to obtain context semantic information caused by one-way computing. The structure of LSTM effectively solves the short-term dependency bottleneck of the RNN. However, from the model structure, it can be seen that compared with RNN, LSTM contains more parameters to be learned, resulting in a greatly reduced learning speed of LSTM.

In order to obtain the context semantic information of the text, this paper adopts the bidirectional long-term and short-term memory network model Bi-LSTM, which can learn the context information well.

The data flow of the Bi-LSTM model is as follows: The translated text is represented by text vector $(\text{sen}_1, \text{sen}_2, \ldots, \text{sen}_n)$ through a BERT pretraining model with sentence granularity, and the vector is used as the input of the Bi-LSTM model. Then, we extract the output vectors $\overrightarrow{h}_t$ and $\overleftarrow{h}_t$ of Bi-LSTM two terminals, with which we obtain the previous information and the subsequent information of the text, respectively. Then, the two vectors are spliced to obtain $H(\overrightarrow{h}_t, \overleftarrow{h}_t)$. Finally, through a full connection layer, a semantic score of translated text with a score range of 0–1 is obtained by using the Sigmoid function.

Specifically, the encoding of translation text $\text{Text} = \{\text{sen}_1, \text{sen}_2, \ldots, \text{sen}_n\}$ and Bi-LSTM are expressed as follows:

$$\begin{aligned}\overrightarrow{h}_t &= \text{LSTM}\left(\text{sen}_1, \ldots, \text{sen}_{t-1}, \text{sen}_t, \text{sen}_{t+1}, \ldots, \text{sen}_n\right),\\ \overleftarrow{h}_t &= \text{LSTM}\left(\text{sen}_1, \ldots, \text{sen}_{t-1}, \text{sen}_t, \text{sen}_{t+1}, \ldots, \text{sen}_n\right),\end{aligned} \tag{17}$$

where $\overrightarrow{h}_t$ and $\overleftarrow{h}_t$ are forward and reverse outputs of the Bi-LSTM model, respectively. $H(\overrightarrow{h}_t, \overleftarrow{h}_t)$ represents the final output of the Bi-LSTM model. The full connection layer is used to reduce the dimension of $H(\overrightarrow{h}_t, \overleftarrow{h}_t)$, the vector after dimension reduction is set as $H'(\overrightarrow{h}_t, \overleftarrow{h}_t)$, and the activation function Sigmoid is used to get a score in the range of 0–1.

$$Y_e' = \sigma\left(H'\right), \tag{18}$$

where Y_e' represents semantic score of the current translated text and σ represents the Sigmoid activation function.

The mean square error (MSE) loss function is the loss function of the model. Suppose the training batch $B = \{\text{Text}_1, \text{Text}_2, \ldots, \text{Text}_n\}$ with n training samples at present, and the loss function formula is shown in (19).

$$\text{Loss}\left(Y_B, Y_B'\right) = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - Y_i'\right)^2, \tag{19}$$

where Y_B is the total score of translated text in a real data set and Y_B' is a predicted semantic score.

Table 1: Comparison of experimental results.

Data set	The proposed model	LSTM	RNN
comp.graphics	0.91	0.87	0.78
comp.os.ms-windows.misc	0.92	0.85	0.78
comp.sys.ibm.pc.hardware	0.88	0.83	0.73
comp.sys.mac.hardware	0.85	0.79	0.68
comp.windows.x	0.92	0.87	0.81
misc.forsale	0.96	0.92	0.85
rec.autos	0.87	0.81	0.73
rec.motorcycles	0.88	0.82	0.78
rec.sport.baseball	0.83	0.74	0.68
rec.sport.hockey	0.92	0.85	0.81
talk.politics.misc	0.93	0.83	0.78
talk.politics.guns	0.84	0.79	0.76
talk.politics.mideast	0.88	0.83	0.78
sci.crypt	0.87	0.82	0.77
sci.electronics	0.89	0.84	0.80
sci.med	0.91	0.85	0.81
sci.space	0.94	0.83	0.77
talk.religion.misc	0.93	0.88	0.84
alt.atheism	0.89	0.83	0.77
soc.religion.Christian	0.87	0.82	0.79

4. Experiment and Analysis

The experiment uses Pearson evaluation criteria to evaluate the proposed model in this paper. $\text{Text\_set} = \{\text{text}_1, \text{text}_2, \ldots, \text{text}_n\}$ is the data set of n translated texts, the corresponding total score set is $S = \{s_1, s_2, \ldots, s_n\}$, and the current score prediction set is $\text{As} = \{\text{as}_1, \text{as}_2, \ldots, \text{as}_n\}$; then, the Pearson evaluation criteria can be defined as follows:

$$\text{Pearson}_{As,S} = \frac{E(\text{As}.S) - E(\text{As})E(S)}{\sqrt{N\left(\text{As}^2\right) - E^2(\text{As})}\sqrt{N\left(S^2\right) - E^2(S)}}, \qquad (20)$$

where E represents mathematical expectation.

In order to verify the effectiveness of the proposed model in this paper, 20 Newsgroups data set [16] is used as experimental data; the proposed model in this paper is compared with the baseline models such as LSTM and RNN; the experimental results are shown in Table 1.

As can be seen from Table 1, the overall performance of the proposed model is higher than that of the text translation method based on RNN and LSTM deep learning.

5. Conclusions

This paper mainly discusses the statistical machine translation method based on a deep neural network and studies the main problems in statistical machine translation, such as word tone order, translation modelling, and so on. Compared with traditional sentence analysis methods, the proposed method greatly improves the analysis accuracy. Aiming at the problem of ordering, a statistical machine translation preordering model based on the neural network is proposed. In this method, the dimension reduction method of neural networks is used to learn the low dimensional vector representation of any ordering feature from the unmarked data, and then, a multilayer neural network is used to combine this lexical representation with other features and integrate it into a linear ordering model. Aiming at the problem of translation modelling, a language model based on improved RNN and LSTM is proposed; it covers all historical sequence information and has better performance than conventional RNN. It can be used to realize the multiclassification of emotional attributes of the English text and can recognize the emotional attributes of text more accurately than the traditional RNN. The model proposed in this paper cannot accurately identify other sentence elements except the subject, predicate, and object, which will be an important research direction in the future.

Data Availability

The basic data used in this paper are downloaded from the online public data set: 20 Newsgroups data set http://qwone.com/~jason/20Newsgroups/.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by a grant from Research and Planning Project of Philosophy and Social Sciences in Heilongjiang Province (No.18YYB165).

References

[1] A. Balahur and M. Turchi, "Comparative experiments using supervised learning and machine translation for multilingual sentiment analysis," *Computer Speech & Language*, vol. 28, no. 1, pp. 56–75, 2014.

[2] S. Hiroya and M. Honda, "A multi-domain translation model framework for statistical machine translation [J]," *European Journal of Operational Research*, vol. 215, no. 2, pp. 832–840, 2013.

[3] J. Lee, K. Cho, and T. Hofmann, "Fully character-level neural machine translation without explicit segmentation," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 365–378, 2017.

[4] H. Somers, "Review article: example-based machine translation [J]," *Machine Translation*, vol. 14, no. 2, pp. 113–157, 1999.

[5] W. Hua and H. Wang, "Pivot language approach for phrase-based statistical machine translation [J]," *Machine Translation*, vol. 21, no. 3, pp. 165–181, 2007.

[6] C. Ma, "Syntax-based transformer for neural machine translation," *Journal of Natural Language Processing*, vol. 28, no. 2, pp. 682–687, 2021.

[7] P. Giulio, R. Marco, and G. Chiara, "Expressive ontology learning as neural machine translation [J]," *Journal of Web Semantics*, vol. 52, pp. 66–82, 2018.

[8] N. Thien, N. Lam, and T. Phuoc, "Improving transformer-based neural machine translation with prior alignments [J]," *Complexity*, vol. 2021, pp. 1–15, 2021.

[9] V. M. Sánchez-Cartagena, J. A. Pérez-Ortiz, and F. Sánchez-Martínez, "A generalised alignment template formalism and its application to the inference of shallow-transfer machine

translation rules from scarce bilingual corpora," *Computer Speech & Language*, vol. 32, no. 1, pp. 46–90, 2015.
[10] B Sheng and G Sun, "Optimal energy resources allocation method of wireless sensor networks for intelligent railway systems [J]," *Sensors*, vol. 20, no. 2, p. 482, 2020.
[11] Q Jiang, Y Dong, and J Peng, "Maximum likelihood estimation based nonnegative matrix factorization for hyperspectral unmixing," *Remote Sensing*, vol. 13, no. 13, p. 2637, 2021.
[12] N. Ahmed and M. Campbell, "Variational bayesian learning of probabilistic discriminative models with latent softmax variables," *IEEE Transactions on Signal Processing*, vol. 59, no. 7, pp. 3143–3154, 2011.
[13] M. Pohl, M. Uesaka, and K. Demachi, "Prediction of the motion of chest internal points using a recurrent neural network trained with real-time recurrent learning for latency compensation in lung cancer radiotherapy," *Computerized Medical Imaging and Graphics*, vol. 91, Article ID 101941, 2021.
[14] Y. Wang, L. Wang, and G. Chen, "An improved ant colony optimization algorithm to the periodic vehicle routing problem with time window and service choice," *Swarm and Evolutionary Computation*, vol. 55, Article ID 100675, 2020.
[15] L. Farhi and A. Yasir, "Optimized intelligent auto-regressive neural network model (ARNN) for prediction of non-linear exogenous signals," *Wireless Personal Communications*, vol. 124, no. 2, pp. 1151–1167, 2022.
[16] Y. Ai, C. Sun, and J. Tie, "Research on recognition model of crop diseases and insect pests based on deep learning in harsh environments," *IEEE Access*, vol. 8, pp. 171686–171693, 2020.