Hindawi
Mobile Information Systems
Volume 2022, Article ID 3114123, 8 pages
https://doi.org/10.1155/2022/3114123

Research Article

Toward English-Chinese Translation Based on Neural Networks

Fan Xiao

Henan Polytechnic Institute, Nanyang, Henan 473000, China

Correspondence should be addressed to Fan Xiao; 2007046@hnpi.edu.cn

Received 8 January 2022; Revised 23 February 2022; Accepted 25 March 2022; Published 29 April 2022

Academic Editor: Hasan Ali Khattak

In traditional interactive English translation systems, English semantic context is not obvious in the process of the English translation, and the selection of optimal feature semantics does not reach the optimal translation solution, which leads to low translation accuracy. Toward this solation, a neural network (NN) based translation approach is proposed to predict word order differences in language translation and improve translation accuracy in long sentences. In this study, a multilayer NN model has been established to victories the unlabeled text words, realize the combination of word representation, vector features, and extract effective information of various sentences and semantics. In linear ranking frameworks, the NN is used to rank and score words, obtain semantic information of sample data, and predict the difference of word order. Experimental results show that the NN preorder model can significantly improve translation accuracy and system performance. The application of NN-based translation model in the practical translation process can reduce the effort of translation work, improve the efficiency of translation, and has a good practical significance.

1. Introduction

Machine translation (MT) has a long history. In 1949, Warren Weaver put forward the first influential machine translation proposal, which marked the beginning of machine translation [1]. In Weaver's proposal, he mentioned the idea of computer translation and proposed to combine the knowledge of statistics, logic, and linguistics to solve the problem of ambiguity in language. In the decades since, MT has come a long way. In 1949, the success of MT is started, but in the early decades of MT research, the so-called MT was almost entirely word-to-word substitution relying on bilingual dictionaries. MT research soon fell into a cold winter. In 1966, the American Advisory Committee on Automatic Language Processing (ALPAC) pointed out in its research report language and machine that "there is no hope for machine translation in the near future." Since then, MT research has stagnated. It was not until 1980, with the increasing international communication, that the urgency of MT led to the prelaunch of MT research. As the research direction of natural language processing and artificial intelligence (AI), MT mainly uses computers to realize the mutual transformation between different languages [2–4].

At present, many Internet companies provide multiple language translation services online such as Google, Microsoft Bing translation, and hundred degrees translations. But the quality of MT still exists a bigger difference between professional translation, especially in the translation of some long sentences, the word order difference between the source language and target language is difficult to accurately describe. To solve the problem of long-distance sequencing, many researches have been carried out. For example, the reordering model based on maximum entropy can achieve accurate translation of sentences through the relationship between different words in sentences. Some learners embed syntactic information of source language into the translation model, which effectively improves the description accuracy of long-distance reordering, but it is easy to prolong the decoding time of translation. Some scholars have proposed a preordering method, which directly converts the source language segment into the target language word order, effectively solving the reordering problem in long sentence translation.

In daily English translation, an interactive English translation system is usually used for English-Chinese translation, but the traditional interactive English translation system

does not select the optimal feature context in the process of characteristic semantic and context extraction, resulting in low accuracy of the translation. Based on the past related research works, this aim of paper is as follows:

(i) To design an interactive ENGLISH-Chinese translation system using a neural network

(ii) In this paper, we propose a preset language translation based on a neural network model, by establishing a linear framework to establish a neural network model for the implementation in this statement and the availability of semantic information extraction, predict the word order of differences in language translation, and improve the accuracy of the translation

(iii) Feature extraction algorithm. The feature extraction algorithm is introduced to select feature semantics, the semantic ontology mapping model is established to select the optimal solution for the interactive English-Chinese translation process, and finally, the English-Chinese translation process is realized through coding

(iv) In this paper, in order to ensure that the design is based on the effectiveness of the English-Chinese translation system by feature extraction algorithm, design the contrast simulation test, through the experimental data show that this design is based on feature extraction of English-Chinese translation system by numerical method to interactive translation between English and Chinese at the same time can effectively solve the selection process of the optimal characteristics of the context

2. Literature Review

In 1993, the pioneering work of IBM scientist Brown et al. [5] opened the chapter on statistical machine translation. In IBM's work, the MT system automatically learns probabilistic word alignment rules from a large number of bilingual parallel corpora, rather than manually maintaining bilingual word lists. Based on mM's work, Firat et al. [6] proposed a phrase (phase.based) statistical machine translation. Phrase-based word alignment remains a core component of statistical machine translation to this day. Another important component of statistical MT is the language model, which is limited by the dimensional curse [7]. Wang et al. [8] are identical to the neurolanguage model proposed in 2003, which uses dense fixed-length vectors to represent words in the vocabulary and calculates sentence probability in turn. NN is used for line training, which overcomes the dimension explosion problem of traditional language models. Later, [9] combined the neural language model with statistical machine translation in their study, which greatly improved translation performance.

At present, See et al. [10] are the mainstream model of neural machine translation, an encoder-decoder model structure. The model is composed of an encoder and decoder. The encoder is a recurrent neural network (RNN). The input is a sequence of words, and the embedded representation vector of one word is input to the RNN unit each time, and the hidden state vector in the RNN is updated according to the input. After the input is complete, a final state vector of constant length is obtained. Theoretically, this final state vector replaces all the semantic information previously entered in sequence. The decoder is a generative model, which also restores information in the state vector step by step through the RNN unit until the output of a special mark ends. In model training, the encoder and decoder are supervised to learn according to the probability of output tag sequence maximized under the premise of the given input sequence.

Based on the decoder model structure, Zhang and Zong [11] proposed the SEQ2SEQ model structure, which is suitable for scenes where both input and output are sequences, and the relationship between length and length is uncertain, such as MT or question answering system. They improved the cyclic NN from one layer to four layers, and through some small techniques, such as reversing the input sequence. The final results show that this neural MT has been able to meet the requirements of the 2014 World Machine Translation Conference (World Machine Translation. WMT), a statistical MT model based on phrase and language models.

Later, Larochelle and Hinton [12] and Denil et al. [13] combined the Seq2Seq model with word alignment commonly used in statistical MT and proposed the attention mechanism. When the model is decoded, it not only uses the hidden state of the global information finally output by the model in the encoding stage but also focuses on the partial words in the input sequence (the words currently being translated) and increases the robustness of the model through word alignment. By introducing the attention mechanism, the model can maintain good performance even when the input sequence is as long as 50 words. The results of the Seq2Seq model based on the attention mechanism in WMT2015 exceeded the highest level at that time. Recently, Facebook Research Institute combined an attention mechanism with a convolutional neural network (CNN) to construct neural machine translation model [14]. In Google's work, an attention mechanism was completely used to a construct the neural machine translation model [15]. Currently, the research on neuromachine translation is growing in every field.

3. Neural Machine Translation

At present, the mainstream model of neural machine translation is the encoder-decoder model structure proposed in 2014. It is also an end-to-end learning model, so it solves the problem of learning many intermediate components in statistical machine translation models. Before introducing neural machine translation, a brief introduction to deep learning (DL) approaches is as follows.

3.1. Deep Learning and Recurrent Neural Networks. Machine learning (ML) technology is now increasingly used in all aspects of life such as search engines, information filtering,

recommendation on social networks, image recognition, image capturing, speech recognition, and natural language processing (NLP). But traditional machine learning is limited in its ability to process raw and unstructured data. To learn structured feature information from unstructured raw data, a method called representation learning is required. DL is a representation learning approach with multiple layers of representation. Through layers of simple but nonlinear processing, starting with the original input, each layer can obtain a higher level and more abstract representation of features. With enough multilevel transformations, even very complex transformational relationships can be learned. DL usually refers to deep feed forward neural networks. In feed-forward neural networks (FFNN), information starts from the input and moves forward one layer at a time. Errors are transmitted back propagation from the last layer, and parameters in the network are updated by stochastic gradient descent (SGD) algorithm and other optimization methods.

The structure of RNN was first proposed in 1990 and was designed to deal with sequence data. RNN has also been successfully applied to language models. In general, for sequence vector data $XL, X2 \cdots, Xn$, RNN is processed from beginning to end, and each time for input x_i, RNN updates its hidden layer state HI. Its hidden state is often referred to as the "memory" of the RNN and is used to store information from the previous $I-1$ data. The key of RNN is to update the hidden state of RNN. RNN can be represented by the following equation:

$$h_t = f(h_{t-1}, x_t). \tag{1}$$

As shown above, the implicit state is determined by the current input x_i and the previous implicit state h_{t-1}. The hidden state of the initial test can be set to 0 or a learnable parameter. F of the above formula is usually a nonlinear function, such as sigmoid or tanh, which is represented by the equation.

$$h_t = \sigma(W_{hh}h_{t-1} + W_{xh}x_t). \tag{2}$$

In the above formula, W_{hh} and W_{xh} are parameters of RNN. RNN has an implicit state and an output. In RNN of simple form, as shown in equation (1), the output is the same as the hidden state, namely, $y_t = h_t$. However, in more complex structures, the output and the hidden state are often different, such as LSTM. If the output of RNN is used for classification, then it can be represented by the following equation.

$$P_t = \text{soft max}\left(W_{hy}y_t\right). \tag{3}$$

Although the structure of RNN is proposed to overcome the problem of long dependence, and it can solve the problem of arbitrary length dependence theoretically, it is not ideal in practical application. The reason lies in the two key problems of RNN: gradient explosion and gradient disappearance. The gradient explosion and gradient disappearance of RNN were first proposed in 1994, and it still restricts the application effect of RNN until now. Pascanu explained the gradient explosion and gradient disappearance of RNN in detail from various aspects and summarized some solutions in 2013. When the gradient explosion occurs, the parameter update becomes extremely unstable, and the model cannot converge due to the violent oscillation. When gradient disappearance occurs, the contribution of the long-distance term far from the output to the gradient becomes negligible, so the model cannot obtain the long-distance dependence information. Gradient explosion can generally be solved through gradient clipping, but gradient disappearance is difficult to solve, and now, the most widely used solution is the use of a gated neural unit. In Figures 1(a) and 1(b), the basic RNN and expanded RNN architecture for language translation are represented.

3.2. LSTM and GRU. Gradient explosion can be conveniently solved by gradient clipping, but the problem of gradient disappearance is very difficult to solve and is still a difficult problem in recurrent neural networks until today. Therefore, many methods have been proposed in the academic world, such as the regularization method, jump connection, hierarchical neural network, and second-order training method. Among these methods, the most widely used and the most effective one is the gating unit to construct recurrent neural networks. Among them, LSTM (long short-term memory) and GRU (gated recurrent unit) are the most widely used. The disappearance of the gradient is caused by the updating of the implicit state in the recurrent neural network (equation (2)). The implicit state circulates in the network by multiplying with the weight matrix, resulting in exponential growth or attenuation of gradient in the reverse derivation. Therefore, if we modify the form of equation (2) and add a term that propagates linearly along with the network, the gradient disappearance will not occur. The specific form can be represented by the following equation.

$$h_t = h_{t-1} + \sigma(W_{hh}h_{t-1} + W_{xh}x_t). \tag{4}$$

But the structure is shown above still has some problems. Because when the hidden state of the structure shown is propagated, the next stage copies exactly the hidden state of the previous moment. But in practice, text input, for example, may require a partial reset of the previous state when it comes to punctuation, stop words, or a new subject. Based on the above formula, LSTM introduces forgetting gates, input gates, and output gates to make the expression ability of neural networks more flexible. In Figures 2(a) and 2(b), the GRU and LSTM architecture for language translation is represented.

3.3. Seq2Seq Model. Seq2Seq (sequence-to-sequence) model was first proposed in 2014. It is suitable for scenarios where both input and output are sequences such as machine translation, speech recognition, automatic question answering, and text summarization. This section will introduce the application of the Seq2Seq model in MT. Seq2Seq model is

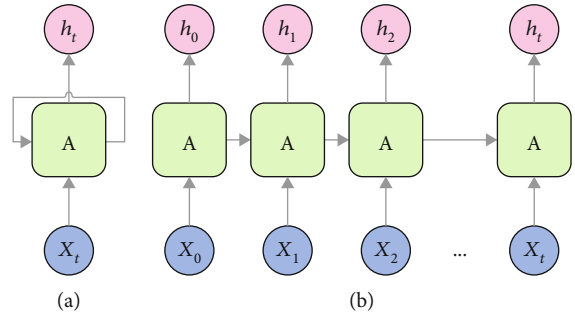


Figure 1: (a) Basic architecture of RNN. (b) Expanded architecture of RNN.

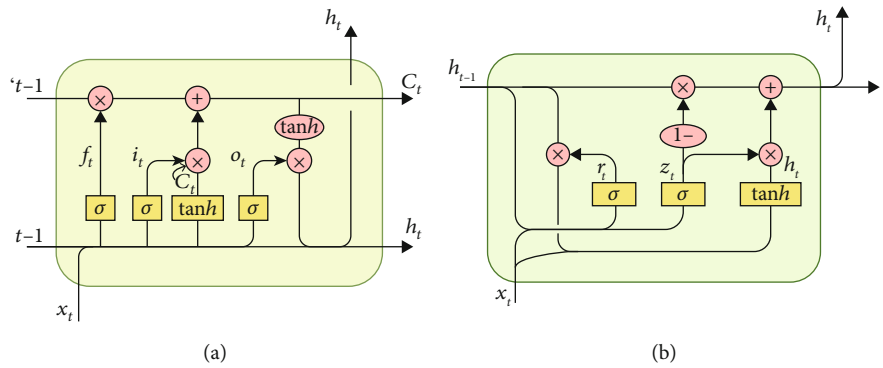


Figure 2: (a) LSTM architecture. (b) GRU architecture.

an encoder-decoder framework. As shown in Figure 3, during translation, the encoder encodes the source language sequence of variable length into the representation vector of fixed length, and then the decoder generates the target language sequence of variable length according to the input representation vector.

The core of the Seq2Seq model is the representation vector connecting the encoder and decoder. The representation vector is a numeric vector of a given length that represents the meaning of the input language. Different from the word-by-word translation of statistical MT Seq2Seq model does not start to translate until the whole meaning of the source language is acquired after the encoding of the input sequence is completed. This is similar to human translation work, that is, first understand the meaning of the source language and then organize the language for translation. Neuromachine translation is thus able to capture long-distance dependencies in languages and also to overcome differences in word order between languages to produce smoother translations.

More specifically, the structure of the encoder and decoder is analyzed. Since both input and output are sequence data, it is natural to use cyclic NN to process sequence data. Most coders and decoders in seq2seq model are cyclic NN. The differences are the types of RNN, including ordinary RNN, LSTM, GRU, BiLSTM, and Bi-GRU. Where the depth of RNN is either single or multilayer, and the direction of the RNN is either unidirectional or bidirectional.

The framework structure shown in Figure 4 translates the input source language "I am a student" into the target language's "Je suis e 'tudiant." The underscore "___" indicates the end of the sentence. The encoder receives word input of the source language in turn, and each input word is converted into fixed-length embedding vector through the embedding layer. Embedding layers can use random initial values or word vectors with training, such as Word2vec or GloVe. The embedded layer is followed by the hidden layer, which consists of RNNs. The initial state of the RNN unit is 0, and it receives the signal of the embedded layer and the output of the previous RNN unit in turn until the end of the sentence is encountered. Begin decoding after coding is complete. When decoding, the initial state of the RNN unit is the state of RNN at the end of the encoding stage, and then the probability of words translated to each target language is calculated through the SoftMax layer according to the output of each RNN unit.

A slight difference between the training stage and the inference stage is that during the training, the loss is calculated by the output probability of softmax and the corresponding correct word to update the parameters of the model. The input of the decoder at each moment is the next word from the corresponding target language in the training corpus pair. Even if the decoder is not at the maximum probability of the word in the softmax layer at the previous moment; in inference, if greedy the strategy is used, the input of the decoder at each moment is the word with the maximum probability output by the softmax layer of the decoder at the last moment, and the decoder will keep output until the end mark of the output sentence "-" at a certain moment, or the longest translation length set by the model is reached. The trainable parameters of the whole model

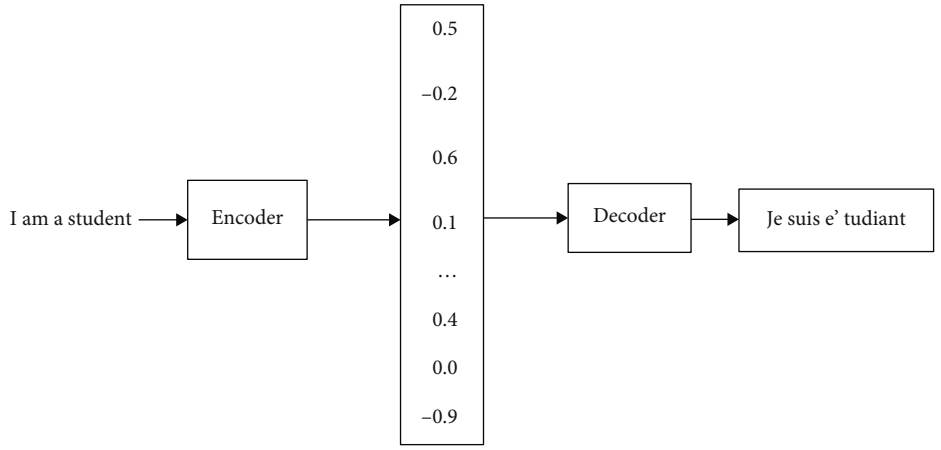


Figure 3: Encoder-decoder frame structure diagram.

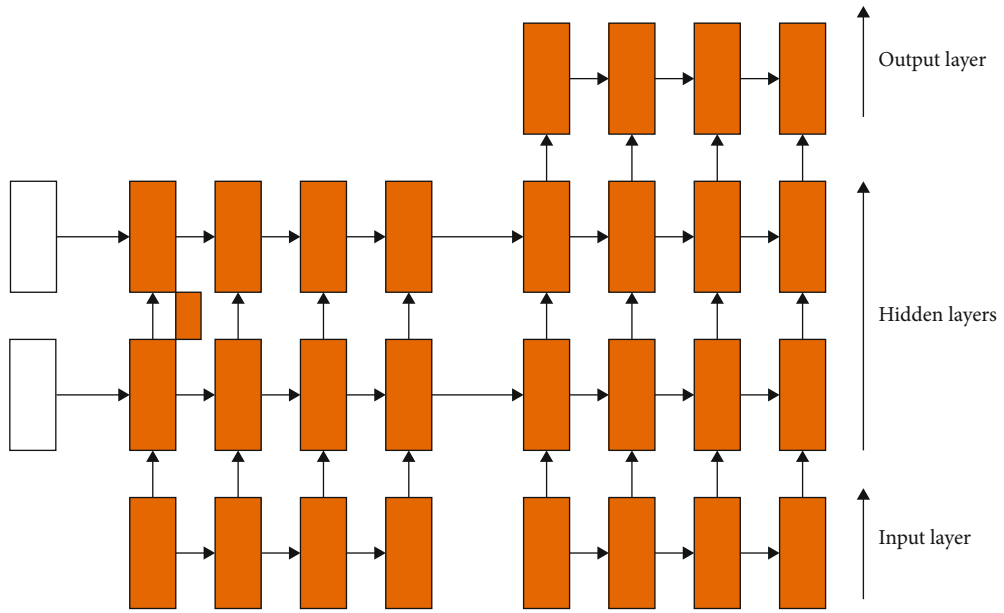


Figure 4: Seq2Seq architecture composed of two layers of one-way RNN.

Table 1: Experimental parameters.

Parameters names	Values
Embedding dim	256
RNN dim	256
Batch size	128
Attention	True
Learn rate	1.0
Train set	All data except the test set
Test set	3000
Epoch	30 K (TED2013) 60 K (UM-Corpus) 120 K (UNv1.0)

include the parameters of the embedded layer, RNNs layer, and softmax layer. The vocabulary size of the language is V, the embedding layer vector dimension is S, and the state dimension of the RNN is H. The parameters of the embedding layer are several digits V ∗ s, the number of parameters of RNNs varies with the type of RNN, the order of magnitude is $S * H + H * h$, and the parameter of softmax layer is $H * V$. H is usually no larger than 1000, and V is often in the hundreds of thousands or more, so the size of the entire model is largely determined by the size of the vocabulary. Stochastic gradient descent (SGD) or more complex optimization methods with adjustable step sizes are usually used in model training, such as Adam.

3.4. Attention Mechanism. Although neuromachine translation based on the seq2seq model described in the previous section has reached its highest level in large-scale translation in 2014, the translation of long sentences is still a challenge

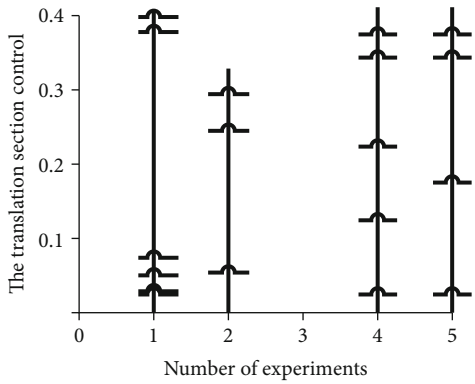


Figure 5: Results of comparison test.

for neuromachine translation. Based on this, the attention mechanism is proposed to be applied to the seq2seq model to construct a stronger neural machine translation model. Attention mechanism was first applied in the field of the image. Google Mind uses an attention mechanism in the RNN model to classify images. It was also first used in machine translation in the same year to align translation and words simultaneously, the first application of the attention mechanism in a natural language. Subsequently, the application of attention mechanism in a recurrent neural network is refined and divided into two categories: global attention and local attention. Attention mechanism, first applied in computer vision research, borrowed from the attention machine in human vision. The attention mechanism in vision is a special visual signal processing mechanism in the human brain. When humans look at a picture, they will first quickly scan the whole picture to get the key areas that need attention and then focus on these key areas to reduce the interference of other information. This mechanism of human brain greatly improves the efficiency and accuracy of human processing visual signals.

The attentional mechanism in neuromachine translation is similar to that in visual processing. In the seq2seq model described in the previous section, the source language is first encoded into a representation vector, and then the representation vector is decoded into the target language. In the decoding process, the model cannot see the specific words of the source language, so attention is scattered on the whole sequence of the source language during translation. This approach is feasible for small sequence lengths. However, for long sequences, there is too much information in the source language, and a single representation vector becomes the bottleneck of message delivery in the translation system. Therefore, through the attention mechanism, the direct information transmission channel between the encoder and decoder is added, and the system will use each output state of the RNN unit in the encoding stage again during decoding. Through the mechanism of attention, neuromachine translation can process longer sequences.

Seq2seq model, which introduces an attention mechanism, will disperse attention in some words of source language differently in the decoding stage. In decoding, the translation and word alignment are carried out simultaneously through the attention mechanism, which is different from word alignment before translation in statistical machine translation. In the Seq2Seq model with attention mechanism introduced, the encoding stage will not change, but the output state H of the RNN layer will change when decoding. It will not be directly connected to the SoftMax layer, but first combined with the context vector, and then output to the SoftMax layer. The background vector is calculated by the attention weight at each moment in the coding period. It can be divided into the following four steps:

(1) Through the current decoding state and the state of each moment in the encoding stage, the weight of attention force at each moment is calculated by equation (5)

(2) The state of each moment in the coding stage was weighted and averaged by attention weight to get the background vector by using equation (6)

(3) Combining the current decoding state with the background vector, the attention vector is calculated by equation (7)

(4) The attention vector outputs to the softmax layer are represented by equation (8)

Expressed by the formula:

$$\alpha_{ts} = \frac{\exp\left(\text{score}(h_t, h_s)\right)}{\sum_s \exp(\text{score}(h_t, h_s))}, \tag{5}$$

$$c_t = \sum_s \alpha_{ts} h_s, \tag{6}$$

$$a_t = \tanh\left(W[c_t ; h_t]\right), \tag{7}$$

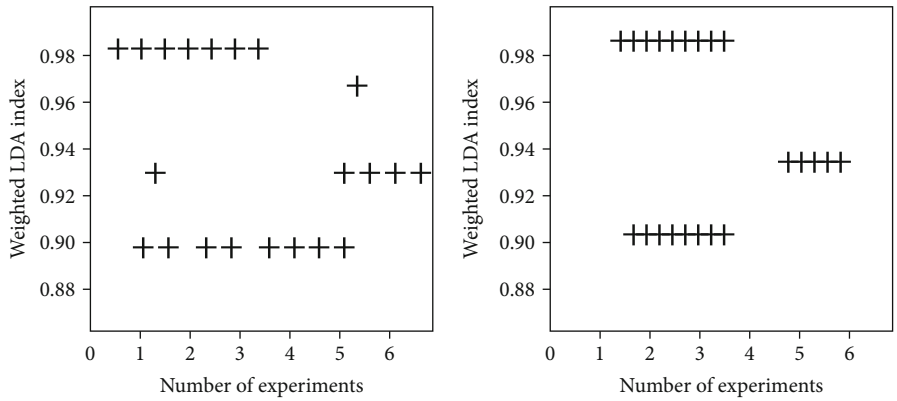
$$p(y_t | y_{<t}, x) = \text{soft max}\left(W_s a_t\right). \tag{8}$$

In the above formula, a_{ts} represents the attention weight corresponding to the moment S in the encoding stage in the moment t in the decoding stage; c_t represents the background vector, because the sum of the weight a_{ts} over s is l , here, the weighted sum over the coded state hs is the weighted average of the forward row; at is the attention vector, and $[c_t ; h_t]$ represents the connection operation between c_t and h_t. Once the attention vector port is calculated, it can be output to the softmax layer to calculate the loss or infer the output, similar to the ordinary seq2seq model.

4. Experiments

For the existing models mentioned in Section 2, this chapter will conduct comparative analysis through experiments.

4.1. Experimental Settings. The experiment in this paper was carried out on MacOS High Sierra operating system. Python3.6 and Tensorflow L. 4.1 were used to construct the model. The model was trained and broken on CPU Intel Core I5 2.7 GHz. In the SEQ2SEQ model used in this paper, the encoder and decoder are composed of two layers of bidirectional LSTM, that is, a total of four layers of LSTM, LSTM unit dimension is 256 dimensions, word embedding



(a) The traditional system (b) Improved system

Figure 6: (a, b) Results comparison of different English-Chinese translation systems.

Table 2: Translation results from Chinese to English.

Model	NIST05	NIST06	NIST08
CNN	0.416	0.357	0.312
RNN	0.418	0.363	0.315
Seq2Seq	0.432	0.389	0.322

Table 3: Average value of Chinese to English cross-connections.

Model	Cross-connection number
CNN	30.4
RNN	28.7
Seq2Seq	16.8

dimension is 256 dimensions, and there are 128 sentence pairs in each batch of data during training. Detailed parameters are shown in the following Table 1.

4.2. Result Analysis. Figure 5 is based on feature extraction algorithm is designed in this paper the interactive English Han translation system and the traditional interactive English-Chinese translation system, section number of control points in the process of translation, on the left side of the translation process of the design of section point distribution diagram, a relatively balanced distribution can be seen from the diagram, the right for the traditional translation of English-Chinese translation system section point distribution, restrained distribution can reflect the correlation between semantics and context of the translation system. The loose distribution indicates that the translation is correct but lacks contextual coherence. The interactive English-Chinese translation system based on the feature extraction algorithm designed in this paper has a compact distribution of translation node control points and no loose distribution.

The comparison of weighted LDA indices is shown in Figures 6(a) and 6(b).

As can be seen from Figure 4, the weighted LDA indices of the interactive English-Chinese translation system designed in this paper based on feature extraction algorithm can be correlated and distributed in an orderly manner, while the translation results of traditional English-Chinese translation systems are lacking of the relationship between the weighted LDA indices and the weighted LDA indices. Unweight LDA index is a measure of semantic depth connection in the process of translation. When weighted LDA indexes are connected in order, it indicates that the translation process is vivid and deep; when weighted LDA indexes are scattered, it indicates that the focus of translation semantics is not grasped.

4.3. Results Analysis. Experiments were conducted on Chinese and English data sets, and BLEU-4 was used as the evaluation index. Table 2 shows the experimental results of machine translation. Can see out, use the preset sequence model of neural network can improve the translation effect, this is because in Chinese and English translation, the sequence is more lexicalization patterns for the access point, and god said network model through the study of the vectorization of vocabulary, find the similarity, effectively enhance the correlation degree between the words, thus, has the better effect of translation.

At the same time, word alignment cross-linking between the source language and target language was used for evaluation. The closer the word order of source language and target language is, the smaller the number of aligned cross-links is and the better the preordering effect is. In Table 3 of the randomly selected 500 English language term accumulation standard test data sets, it can be seen that the preset sequence model of a neural network to obtain the cross-connection number only 16.8 a is far less than the sparse features preset order type mold and not order translation system, effectively improve the preset sequence effect, and get better quality of the translation.

5. Conclusion

In this paper, the Seq2Seq model in neural machine translation (NMT) is studied in depth. According to the characteristics of Chinese, corresponding improvements are proposed for text preprocessing and word embedding parameter initialization, and the structure of the Seq2Seq model is

improved. Different methods of text preprocessing are proposed. In the task of natural language processing, it is necessary to transform unstructured text data into a data format that can be recognized by computers through preprocessing. The traditional Chinese preprocessing method in the translation system is to convert Chinese sentences into word sequences by word segmentation, but this method depends on the accuracy of word segmentation and leads to a large number of Chinese vocabularies. This paper proposes a preprocessing method for converting Chinese sentences into character + named entity sequences through named entity recognition. Experimental results show that the preprocessing method can reduce the parameter scale and training time of the translation model by about 20% and improve the translation performance by about 0.4 BLEU in the English-Chinese translation task.

Data Availability

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

Conflicts of Interest

The author declares that he has no conflict of interest.

References

[1] J. Zhang and C. Zong, "Exploiting Source-Side Monolingual Data in Neural Machine Translation," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 1535–1545, Austin, Texas, USA, 2016.

[2] F. Stahlberg, E. Hasler, A. Waite, and B. Byrne, "Syntactically Guided Neural Machine Translation," http://arxiv.abs/org/1605.04569.

[3] A. Eriguchi, K. Hashimoto, and Y. Tsuruoka, *Tree-to-Sequence Attentional Neural Machine Translation*, Proceedings of the Association for Computational Linguistics, Berlin, Germany, 2016.

[4] R. Sennrich and B. Haddow, "Linguistic Input Features Improve Neural Machine Translation," in *Proceedings of the ACL 2016 First Conference on Machine Translation,*, pp. 83–91, Berlin, Germany, 2016.

[5] D. Dong, H. Wu, W. He, D. Yu, and H. Wang, "Multi-Task Learning for Multiple Language Translation," in *Proceedings of the Association for Computational Linguistics*, pp. 118–127, Berlin, Germany, 2016.

[6] O. Firat, K. Cho, Y. Bengio, O. Firat, K. Cho, and B. Y. Multi-Way, "Multilingual Neural Machine Translation with a Shared Attention Mechanism," in *Proceedings of the 2016 Conference of the Association for Computational Linguistics*, pp. 866–875, Berlin, Germany, 2016.

[7] S. Shen, Y. Cheng, Z. He et al., "Minimum risk training for neural machine translation," *Proceedings of the Association for Computational Linguistics, Berlin, Germany*, vol. 16, no. 1, pp. 1683–1692, 2016.

[8] M. Wang, Z. Lu, H. Li et al., "Memory-Enhanced Decoder for Neural Machine Translation," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing,*, pp. 278–286, Austin, Texas, USA, 2016.

[9] B. Zhang, D. Xiong, J. Su et al., "Variational Neural Machine Translation," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 521–530, Austin, Texas, USA, 2016.

[10] A. See, M. T. Luong, and C. D. Manning, "Compression of neural machine translation models via pruning," pp. 1–11, 2016, http://arxiv.abs/org/1606.09274.

[11] J. Zhang and C. Zong, "Bridging neural machine translation and bilingual dictionaries," pp. 1–10, 2016, http://arxiv.abs/org/1610.07272.

[12] H. Larochelle and G. Hinton, "Learning to Combine Foveal Glimpses with a ThirdOrder Boltzmann Machine," in *Proceedings of the International Conference on Neural Information Processing Systems*, pp. 1243–1251, Sydney, Australia, 2010.

[13] M. Denil, L. Bazzani, H. Larochelle, and N. de Freitas, "Learning where to attend with deep architectures for image tracking," *Neural Computation*, vol. 24, no. 8, pp. 2151–2184, 2012.

[14] T. Cohn, C. D. V. Hoang, E. Vymolova, K. Yao, C. Dyer, and G. Haffari, "Incorporating Structural Alignment Biases into an Attentional Neural Translation Model," in *Proceedings of the Association for Computational Linguistics*, pp. 876–885, Berlin, Germany, 2016.

[15] W. He, Z. He, H. Wu et al., "Improved Neural Machine Translation with SMT Features," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pp. 151–157, Phoenix, Arizona USA, 2016.