Innovative Design and Intelligent Manufacturing
L.C. Jain et al. (Eds.)

doi:10.3233/FAIA241162

Deep Learning-Based Performance Optimization of English Machine Translation

Yuping SONG[a,1]
[a]*School of Foreign Languages, Shenyang, Jianzhu, University, Liaoning 110168, China*

Abstract. This article proposes a performance optimization method for English machine translation based on deep learning and introduces the Bahdanau attention mechanism. Using a parallel corpus of approximately 30,923 sentences for training and testing, the model achieved an average BLEU score of 45.83, showing excellent English translation results. Research results show that this method effectively improves translation accuracy, helps enhance global communication efficiency, and brings new development ideas to the field of language translation.

Keywords. Neural network; deep learning; machine translation; LSTM; attention mechanism; BLEU

1. Introduction

Machine Translation (MT), as a technology that can convert one natural language into another natural language, aims to reduce the high-cost manual translation work, thereby promoting communication and development between different language regions and promoting. The progress of human society [1]. After decades of development, machine translation has gradually evolved from the initial simple literal translation to a highly accurate and real-time large-scale system [2]. With the popularization of personal computers and the development of t[1]ranslation memory tools for translators, machine translation has been widely used in practice [3]. In recent years, end-to-end machine translation models based on neural networks have achieved superior results than traditional methods by simplifying complex system design and optimizing the training process [5].

With the acceleration of economic globalization, exchanges between countries have become increasingly frequent, and the need for language translation has become increasingly important. English is a global language, and English-Chinese translation is particularly prominent [8]. Traditional human translation is not only inefficient and error-prone [9], therefore, machine translation models based on deep learning have attracted much attention [10]. As an interdisciplinary subject, machine translation involves multiple fields such as linguistics, mathematics, and computing technology [11], and is indispensable. In recent years, with the advancement of deep learning algorithms and improvements in computing power, machine translation based on neural networks has developed rapidly, significantly simplifying system design and improving translation

[1] Corresponding Author.Yuping SONG syp115@126.com

effects [14]. For example, some studies combine deep neural networks with rule learning to improve translation accuracy compared with traditional deep neural network models.

In summary, although the existing machine translation models have many advantages in terms of efficiency and cost, there are still some challenges, such as hindered gradient return and insufficient translation accuracy. The purpose of this paper is to improve the English machine translation model by using deep learning technology based on neural network, and to explore new ways to improve the system performance by introducing new evaluation indexes and methods, so as to promote the further development of the machine translation technology and to facilitate the communication and understanding among people all over the world.

2. Methods

2.1 System design

Standard English text translation includes the following different stages: preprocessing of source and target languages, word embedding, encoding, decoding, and then generating the target text. The workflow is shown in Figure 1.

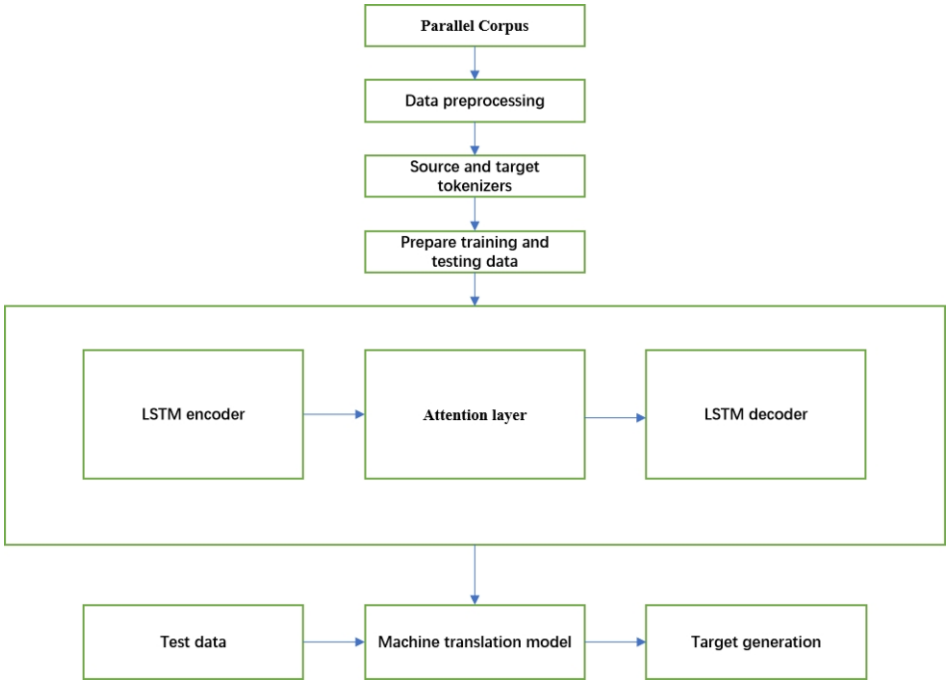

Figure 1 Workflow

1) Preprocessing. Corpus preprocessing is the most important task in developing any neural machine translation system.The preprocessing of parallel corpus is crucial for the development of neural or statistical models. The specific steps include case conversion, disambiguation and cleaning.

2) Fill in the sentences. After preprocessing, when passing text as input to a recurrent neural network or LSTM, some sentences will naturally be longer or shorter, and their lengths are inconsistent, so sentences need to be padded to ensure consistent lengths.

3) Word embedding. This paper uses GloVe (global vector for word representation) to effectively learn word vectors, combined with matrix factorization techniques such as LSA and local context-based learning, similar to word2vec.

4) Encoder. The encoder is responsible for generating think vectors or context vectors that represent the meaning of the source language. A number of symbolic representations are used in the encoding process: the x_t It's time to step t the losers.$h_t$andc_tIt's time to step$t$LSTM internal state;y_tIt's time to step$t$Generated output.

Take a simple sentence "How are you, sir?" as an example. This sequence can be viewed as a sentence consisting of 4 words. it's here: x_1 is "How", x_2 is "are", x_3 is "you",x_4It's "sir".

The sequence will be read in 4 time steps as shown in Figure 2.

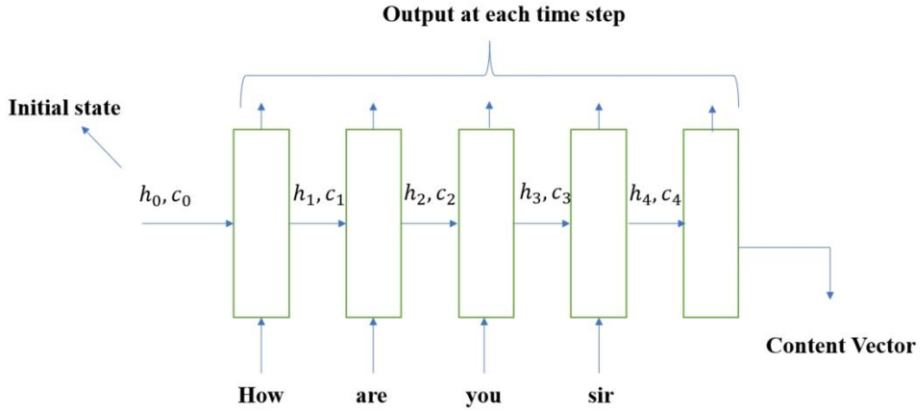

Figure 2 Sentences read by the LSTM encoder

exist$t = 1$, it remembers that the LSTM unit read "How"; in$t = 2$, it recalls that the LSTM read "How are"; in$t = 4$When the final state$h_4$andc_4Memorized the entire sequence "How are you, sir?"

5) Context vector. The context vector becomes the starting state of the decoder. The LSTM decoder does not use zero as the initial state, but uses the context vector as the initial state.

6) Decoder.

In the traditional machine translation decoding process, the decoding of each word is the same weight, where the weight refers to the neural network in the back propagation of the loss function for the gradient, the loss of each word is directly summed up to take the average, which can be interpreted as the loss of each word is the same weight. The decoder based on the attenuation weight loss function will give different weight loss to the words that appear successively, i.e., each time the gradient descent to update the parameters, favoring the words that appear first to be translated first to this case.

The decoder states used in this paper use context vectors that $v = \{v_h, v_c\}$as initialization, where$h_0 = v_h$and$c_0 = v_c$, $h_0$and$c_0 \in \mathrm{LSTM}_{\mathrm{dec}}$Context vectors. Context vectors are an important link between encoder and decoder, forming an end-to-end

computational chain for end-to-end learning. In machine translation, the target language is composed of multiple words. The translation effect of a sentence is jointly influenced by the prediction results of these words. In training, these words are trained in parallel.

The only thing shared between the encoder and decoder is the context vector v, since it is the only information available to the decoder about the source language sentence. The mth prediction of the translated sentence is calculated by the following equation:

$$\begin{aligned} \mathrm{C_m}, h_{\mathrm{m}} &= \mathrm{LSTMdecoder}\left(\mathrm{yT}^{\mathrm{m-1}} \mid \mathrm{v}, \mathrm{y}_{\mathrm{T}}^{1}, \mathrm{y}_{\mathrm{T}}^{2}, \cdots, \mathrm{y}_{\mathrm{T}}^{\mathrm{m-1}}\right), \\ \mathrm{y}_{\mathrm{T}}^{\mathrm{m}} &= \mathrm{softmax}\left(\mathrm{w}_{\mathrm{sodfmax}} * h_{\mathrm{m}} + \mathrm{b}_{\mathrm{soffmax}}\right) \end{aligned} \tag{1}$$

2.2 Attention mechanisms

The attention mechanism is often used in the network model of encoding-transcoding, where encoding is responsible for turning a sequence of variable-length signals into fixed-length vectors to express them, and decoding is responsible for generating a specified sequence of fixed-length vectors according to the semantics, which realizes the normal functions of encoding-transcoding inputs and outputs.

Integrating the pre-trained model BERT into NMT, this article uses the attention mechanism method. Consider two methods to weigh the two attention mechanisms in the model: one is to process multiple attention mechanisms in series, input the input to multiple attention mechanisms in sequence, and use the result of the previous attention mechanism as The input of the next attention mechanism considers the interaction between attention mechanisms, but does not take the results of other attention mechanisms into the final result and only uses them as input; the other is to process multiple attention mechanisms in parallel , input the input into multiple attention mechanisms respectively, and then use the weighted average of their respective results as the final result. Although the results of the attention mechanism are weighted and taken into account in the final result, the attention mechanism is not considered. interaction between.

Conceptually, attention is regarded as a separate layer whose responsibility is to provide the first layer of the decoding process$i$generated by a single time step$c_i \circ c_i$The calculation procedure is shown below.

$$\begin{aligned} \mathrm{c_i} &= \sum_{\mathrm{j=1}}^{\mathrm{L}} \propto_{\mathrm{ij}} h_{\mathrm{j}}, \\ \propto_{\mathrm{ij}} &= \frac{\exp\left(\mathrm{e_{ij}}\right)}{\sum_{\mathrm{j=1}}^{\mathrm{L}} \exp\left(\mathrm{e_{ij}}\right)} \end{aligned} \tag{2}$$

Eqe_{ij}It's a calculations$_i$Time encoder no$j$The importance or contribution factor of a hidden state and the previous state of the decoder is expressed as follows.

$$\mathrm{e_{ij}} = \mathrm{a}\left(\mathrm{s_{i-1}}, h_{\mathrm{j}}\right) \tag{3}$$

In general, the attention mechanism occurs between the target element and all the elements in the target, while the self-attention is different, basically occurs between the internal elements of the target, in essence, the computational process is still consistent with the attention, only the object of computation has changed.

2.3 Training of the algorithm

The proposed system training algorithm is as follows:

Preprocessing the sentence pairs of the source and target language, $x_s = x_1, x_2, \cdots, x_L$ and $y_i = y_1, y_2, \cdots, y_L$, As described in the preconditioning section.

Word embedding was performed using the GloVe word embedding matrix. Create a nested layer object, embedding_layer = Embedding (num_words, EMBEDDING _ SIZE, weights=embedding_matrix).

The source statement $\mathrm{x_s} = \mathrm{x_1}, \mathrm{x_2}, \cdots, \mathrm{x_{Ls}}$ feeds the encoder and obtains the context vector $\mathrm{v_\circ}$ under the condition of $\mathrm{x_s}$ through the attention layer.

The initial state of the decoder is $(\mathrm{h_0}, \mathrm{c_0})$ is set to the context vector $\boldsymbol{v}_\circ$.

$$\begin{aligned} \mathrm{y_T^M} &= \mathrm{softmax}\left(\mathrm{w_{softmax}} * h_\mathrm{m} + \mathrm{b_{softmax}}\right) \\ \mathrm{w_T^m} &= \underset{\mathrm{w^- \in v}}{\mathrm{argmax}} \mathrm{P}\left(\hat{\mathrm{y}}_\mathrm{T}^{\mathrm{m,w^v}} \mid \mathrm{v}, \hat{\mathrm{y}}_\mathrm{T}^1, \cdots, \hat{\mathrm{y}}_\mathrm{T}^{\mathrm{m-1}}\right) \end{aligned} \tag{4}$$

Where $\mathrm{w_T^m}$ is the argmax vocabulary at the mth position in the vocabulary.

3. Results and discussion

The model was simulated multiple times to obtain the values of multiple evaluation indicators, as shown in Table 1, the average BLEU score was 45.83.

Table 1 Values for several assessment indicators

BLEU	Precision	Recall	NIST	WER	F-measure
38.15	60.23	29.40	53.2	5	39.51
36.20	89.56	27.30	51.5	6	41.84
37.50	61.31	25.56	62.56	6	36.07
46.05	88.12	27.77	70.62	7	42.23
43.12	90.19	28.67	65.5	7	43.50
25.56	86.21	26.56	41.09	9	40.60
73.03	60.43	21.32	79.21	4	31.51
71.23	62.31	19.55	78.56	5	29.76

50.12	80.69	28.26	69.15	4	41.85
75.02	78.24	28.57	79.27	3	41.85
28.41	52.56	21.64	58.1	7	30.65
36.65	56.16	24.55	63.23	5	34.16
35.23	58.19	26.71	60.15	6	36.61
61.17	60.25	20.22	70.03	4	30.27
30.15	75.57	28.53	60.96	7	41.42

It can be clearly seen from the table that when the word error rate increases, the BLEU score decreases; and when the word error rate decreases, the BLEU score increases. This is because the more errors there are, the higher the word error rate, and the lower the BLEU score; when the word error rate is lower, it means the translation quality is better, so the BLEU score is higher.

4. Conclusion

This article proposes English machine translation performance optimization based on deep learning. Neural machine translation is a new paradigm in machine translation research. This paper proposes a deep learning encoding-decoding model based on LSTM. The Bahdanau attention mechanism was used in the study. In order to evaluate the efficiency of the proposed system, this paper uses multiple automatic evaluation indicators, such as BLEU, F-measure, NIST, WER, etc. After extensive simulations, the average BLEU score of the proposed system is 45.83. The machine translation optimization model based on language features and transfer learning was researched and designed. Based on the machine translation model of the basic Transformer, transfer learning was used to import the BERT pre-trained language model to optimize the machine translation model, and then improve the Chinese-English machine translation model. Translation effect, reducing the problems of word order reversal, excessive ambiguity, and insufficiently accurate translation content during the translation process. Experimental results show that the proposed optimization model integrating the BERT pre-trained language model can optimize the machine translation of the basic Transformer and greatly improve the machine translation effect, verifying the feasibility and effectiveness of the machine translation optimization model of BERT and transfer learning.

References

[1] Shi, X, Yu, Z., & Niu, L. Y.. (2023). Text-image matching for multi-model machine translation. Journal of supercomputing, 79(16), 17810-17823.

[2] Takahashi, K. , Sudoh, K. , & Nakamura, S. . (2022). Automatic machine translation evaluation using a source and reference sentence with a cross-lingual language model. Journal of Natural Language Processing, 29(1), 3-22.

[3] Minh, N. V. , Minh, K. N. T. , Nguyen, L. H. B. , & Dinh, D. . (2024). Exploring composite indexes for domain adaptation in neural machine translation. Vietnam Journal of Computer Science, 11(01), 75-94.

[4] Maimaiti, M. , Liu, Y. , Luan, H. , & Sun, M. . (2022). Enriching the transfer learning with pre-trained lexicon embedding for low-resource neural machine translation, 27(1), 150-163.

[5] Jian, L. , Xiang, H. , & Le, G. . (2022). Lstm-based attentional embedding for english machine translation. Scientific programming, 2022(Pt.7), 1-8.

[6] Wang, C. , Cai, S. J. , Shi, B. X. , & Chong, Z. H. . (2023). Visual topic semantic enhanced machine translation for multi-modal data efficiency. Journal of Computer Science and Technology, 38(6), 1223-1236.

[7] Wenbin Liu, Yanqing, H. E. , Tian, L. , & Zhenfeng, W. U. . (2023). Research on system combination of machine translation based on transformer, 29(3), 310-317.

[8] Mistree, K. B. , Thakor, D. , & Bhatt, B. . (2023). An approach based on deep learning for indian sign language translation. International Journal of Intelligent Computing and Cybernetics, 16(3), 397-419.

[9] Tan, X. , Zhang, L. Y. , & Zhou, G. D. . (2022). Document-level neural machine translation with hierarchical modeling of global context. Journal of Computer Science and Technology, 37(2), 295-308.

[10] Sujaini, H. , Cahyawijaya, S. , & Putra, A. B. . (2023). Analysis of language model role in improving machine translation accuracy for extremely low resource languages. Journal of Advances in Information Technology, 14(5), 1073-1081.

[11] Li, F. , Chi, C. , & Yan, B. S. M. . (2023). Sta: an efficient data augmentation method for low-resource neural machine translation. Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology, 45(1), 121-132.

[12] Goni, J. '. , Fuentes, C. , & Raveau, M. P. . (2023). An experiential account of a large-scale interdisciplinary data analysis of public engagement. AI & society: The journal of human-centered systems and machine intelligence, 38(2), 581-593.

[13] Liu, C. . (2022). Machine translation of scheduling joint optimization algorithm in japanese passive statistics. Scientific programming, 2022(Pt.18), 1-13.

[14] Liu, X. , Chen, J. , Qi, D. , & Zhang, T. . (2024). Exploration of low-resource language-oriented machine translation system of genetic algorithm-optimized hyper-task network under cloud platform technology. Journal of supercomputing, 80(3), 3310-3333.