Journal of Combinatorial Mathematics
and Combinatorial Computing
www.combinatorialpress.com/jcmcc

Combinatorial Press

Using Machine Learning Techniques to Improve the Accuracy of Computer Translation in the English Translation of Specialized Terms

Yuan Wang[1,✉], Jing Jin[1], Meiling Ye[2], Tingting Tao[2]

[1] *School of International Studies, Maanshan Teacher's College, Maanshan, Anhui, 243041, China*

[2] *Department of General Education, Maanshan Teacher's College, Maanshan, Anhui, 243041, China*

ABSTRACT

At present, machine translation performs better in the general domain translation effect of large-scale bilingual corpus, but the translation effect in specific domains still needs to be improved. In order to optimize the accuracy of machine translation in the domain of English translation of professional terms, this paper proposes a translation model that incorporates syntactic knowledge and terminology. Aiming at the problem of more limited translation domain knowledge in the RNMT and Transformer models based on the self-attention mechanism, an optimization method is proposed. According to the domain characteristics of English translation of professional terms, English syntactic keywords are incorporated into the model training process, the special information contained inside the text of professional terms is learned, and the lexical properties of each word in the dataset are recognized before they are input into the model. Then attempts are made to incorporate the specialized terminology into the model to enrich the parallel corpus required by the model. The experiments confirm the excellent performance of the optimized translation model in this paper on the De→En terminology translation task, which improves 22.67 BLEU values compared to the base model. And the fluctuation of its BLEU value with the change of sentence length is small, which further indicates that the method optimizes the accuracy of the machine translation model in the English translation of professional terms.

Keywords: Self-attention mechanism, Transformer model, Machine learning, Syntactic keywords

1. Introduction

Under the environment of continuous development and updating of information technology, computerized translation has developed rapidly and has been widely used in various fields of language services. Computer translation can be divided into machine translation and computer-assisted

✉ Corresponding author.
E-mail address: m17755502925@163.com (Y. Wang).
Received 05 February 2024; Revised 12 April 2024; Accepted 05 December 2024; Published Online 15 April 2025.
DOI: 10.61091/jcmcc127a-333

translation from the application point of view. Machine Translation (MT) Machine translation is the process of automatically converting one natural language text into another natural language text using a computer [1]. From a linguistic point of view, machine translation is the development and application of natural language processing technology, the principle of which is to utilize a computer to automatically convert one natural language into another target natural language according to specific rules [2]. From the earliest dictionary matching, to the dictionary combined with linguistic rules of translation, and then based on the corpus of statistical machine translation, with the rapid improvement of computer hardware and software level, machine translation technology is also becoming more and more mature, able to provide real-time, convenient translation services for ordinary users [3-4]. The current common network of online translation systems, such as Google Translate, Baidu Translate, Youdao Translate, Tencent Translate, etc. all belong to the category of machine translation. The working principle of computer-aided translation (CAT) is translation memory + machine translation + human translation/proofreading, and its core technology is translation memory and terminology database [5]. To put it simply, the system will divide the original text into individual sentences or paragraphs and decompose them into words or phrases according to the corresponding rules, and then output the corresponding translation results according to the terminology database attached to the system or the corpus built by the user [6-7]. In addition, the system will automatically retrieve the same or similar translation resources in the translation memory, and give the reference translations for the translators to learn from [8]. At present, the common CAT software includes SDL Trados, Smarteat, YiCAT, iCAT, Transmate, Wordfast and so on.

In thesis writing, academic communication, academic cooperation, etc., the English translation of professional terms is extremely demanding, and such translations not only need to accurately convey the meaning of the original text, but also need to maintain the academic style and professionalism, so as to ensure that the academic results can be effectively disseminated and applied. The specialization and complexity of the academic field make academic translation require translators to have profound academic literacy and professional knowledge; at the same time, the terminology and conceptual differences in various subject areas also increase the difficulty of translation, and also need to solve the problems of cultural differences and language expression to ensure that the translation results meet the understanding and acceptance of the target readers [9-10]. However, computer translation faces a number of challenges. Among them, example-based machine translation is an empirical English language and literature translation strategy, which does not require complex deep syntax as well as semantic analysis and improves the efficiency of English language translation [11]. However, the instance-based machine translation method has higher requirements on the quality of the instance base. The traditional machine translation method uses pipelined operation-by-operation to implement lexical identification as well as syntactic analysis of the original corpus to obtain the syntactic structure of the English language, which makes the iterative transfer of errors between translation tasks and the reduction of structured instances' accuracy, and leads to the reduction of the accuracy of English language and literature translation [12-13]. In addition, in terms of language processing ability, although CAT technology performs well in dealing with highly structured and repetitive texts, the quality of co-translation is often unsatisfactory for some highly contextualized, culturally specific and creative contents [14-15]. Therefore, the accuracy of machine translation needs to be optimized.

Aiming at the defect of insufficient utilization of English terminology translation domain knowledge in the training process of traditional machine translation models, this paper proposes a terminology English translation model that integrates English syntactic knowledge and terminology. The simplified English features are represented as syntactic and lexical features, and the keywords and syntactic labels of the text are extracted through the syntactic tree. Then the syntactic and lexical features are combined with the translation model respectively, and the terminology is used to expand the parallel corpus needed for training. Through comparison experiments with the baseline model and

performance analysis experiments, the training accuracy of the optimized machine translation model on the task of English translation of professional terms is verified.

2. Overview

In CAT, literature [16] developed a knowledge graph model for IoT and CAT systems supported by an improved algorithm with long and short-term memory and support vectors, which has good accuracy in labeling entities in translated sentences and excels in relationship extraction. Literature [17] found an 8% improvement in translation accuracy for CAT with automated programming approach by comparing manual translation. Literature [18] found that human post-editing based on CAT significantly improved the accuracy and cultural fidelity of English translation of Cantonese opera terms.

In terms of machine learning, literature [19] constructed a deep learning system to optimize the fluency of machine translation and quality in professional news translation. Literature [20] optimized the quality of statistical machine translation in the translation of both English and Slovenian, a Slavic language, using a differential evolutionary approach. Literature [21] improved machine translation using corpus weight balancing approach and practical comparison showed that the performance of the new machine translation was improved. The combination of the two in the application of intelligent algorithms and human translators does improve the translation performance, but its accuracy improvement is not great, and further efforts are needed.

3. Neural machine translation model

3.1. RNMT model based on self-attention mechanism

Recurrent Neural Machine Translation (RNMT) model based on self-attention mechanism [22] can be divided into three parts: encoder, decoder and attention mechanism. The source language sentence gets the input sequence $X=(x_I,x_2,\ldots,x_{T_v})$ after word embedding, the input sequence is encoded at the encoder side to generate intermediate vectors, and the decoder extracts the corresponding feature information from the intermediate vectors through the attention mechanism to generate the target sequence $Y=(y_1,y_2,...,y_{T_s})$.

The model adopts bidirectional RNN, in the encoding stage, the forward RNN reads the source language sequence from left to right, and the backward RNN reads the source language sequence from right to left, and splices the hidden state generated by the forward RNN at the current moment with the hidden state generated by the backward RNN to obtain the hidden state at the current moment, as shown in the following equation.

$$h_t^{(1)}=f(U^{(1)}h_{t-1}^{(1)}+W^{(1)}x_t^{(1)}+b^{(1)}) \tag{1}$$

$$h_t^{(2)}=f(U^{(2)}h_{t-1}^{(2)}+W^{(2)}x_t^{(2)}+b^{(2)}) \tag{2}$$

$$h_t=[h_t^{(1)},h_t^{(2)}] \tag{3}$$

In the decoding phase, the word y_t at the current moment is predicted by the hidden state s_t at the current moment of the decoder, the predicted word y_{t-1} at the previous moment, and the source-side context vector c_t at the current moment, see Equation (4).

$$p(y_t\,|\,y_1,y_2,.....,y_{t-1},x)=g(y_{t-1},s_t,c_t) \tag{4}$$

The hidden state of the decoder at the current moment is predicted by the hidden state s_{t-1} of the previous moment, the predicted word y_{t-1} of the previous moment, and the source-side context vector c_t of the current moment, see equation (5).

$$s_t = f_2(y_{t-1}, s_{t-1}, c_t) \tag{5}$$

In order to obtain the source context vector c_t, the attention mechanism is introduced into the model, and the weighted sum operation is performed on the source hidden vectors to obtain the source context vector c_i, and the detailed procedure is as follows.

$$c_t = \sum_{j}^{T} \alpha_{tj} h_j \tag{6}$$

$$\alpha_{tj} = \frac{\exp(e_{tj})}{\sum_{k=1}^{T} \exp(e_{tk})} \tag{7}$$

$$e_{tj} = a(s_{t-1}, h_j) \tag{8}$$

Where: $a(\cdot)$ denotes the nonlinear function, e_{ij} denotes the weight of the attention mechanism, h_j denotes the encoder hidden vector, and α_{ij} denotes the attention weight after performing softmax operation.

3.2. Transformer model based on self-attention mechanism

The classical translation model Transformer [23], the translation process adopts the self-attention mechanism and completely abandons neural network structures such as RNN and CNN, which improves the translation efficiency and also obtains accurate translation results. Transformer's encoder consists of n identical encoder layer, and each encoder layer consists of a multi-attention sublayer and a fully-connected feedforward neural network sublayers, in order to alleviate the problem of gradient vanishing, residual connections are introduced between every two sublayers, and then layer regularization is performed, which is to make the gradient transfer easier and to speed up the convergence of the model, see Eq. (9).

$$layernorm(h + sublayer(h)) \tag{9}$$

Where: $layernorm(\cdot)$ denotes the output of the layer regularization function, $sublayer(\cdot)$ denotes the sublayer output. h denotes the hidden state of the sublayer input.

The decoder consists of n identical decoder layers stacked on top of each other, and in contrast to the encoder, the decoder has an additional encoding-decoding cross-attention sublayer to combine the information from the encoder and the decoder, which also requires the use of residuals connected between each of the two sub-layers, followed by the layer regularization operation.The basic unit of the multi-head attention mechanism in the encoder sublayer and the decoder sub-layers of the Transformer is the scaling of the dot product attention mechanism, whose inputs are the query matrix Q and the key-value pair matrices K and V. The dot product attention mechanism training process is as follows.

The source sentence, after word embedding and positional embedding, yields the input sequence $X = (x_1, x_2, ..., x_n) \in R^d$. Three matrices W_Q, W_K, W_v are defined, and three new vectors q_t, k_t, v_t are generated by using W_Q, W_K, W_v to perform linear transformation operations on the input sequence respectively. Put all q_t's together into one large matrix, denoted as the query matrix q_t Put all k_t's together into one large matrix, denoted as the key matrix K, and all v_t's together into one large matrix, denoted as the value matrix V. The attention weights of the first word are obtained by multiplying the query vector q_t of the first word with the key matrix K. After that, it is also necessary to softmax the obtained values so that their sum is 1. The obtained weights are multiplied by the value vectors of the corresponding words respectively v_t for weighted summation to obtain the output of the first word. The above operation is continued for the subsequent input vectors to obtain all the outputs after passing through the dot product attention mechanism, see equation (10).

$$Attention(Q,K,V) = soft\max(\frac{QK^T}{\sqrt{d_k}})V \tag{10}$$

Finally the multi-head attention mechanism splices the results of multiple dot product attention mechanisms to obtain the final output. See equation (11), equation (12).

$$head_i = Attention(QW^Q, KW^K, VW^V) \tag{11}$$

$$MultiHead(Q,K,V) = Concat(head_i)W^o \tag{12}$$

Where: W^Q, W^K, W^V, W^O is the parameter matrix. $head_i$ is the output vector of the i rd attention head.

The feedforward neural network is a fully connected network layer between two layers and uses RELU as the activation function, see equation (13).

$$FFN(x) = \max(0, xW_1 + b_1)W_2 + b_2 \tag{13}$$

Where: W_1, W_2, b_1, b_2 is the model parameter.

Since Transformer does not contain recurrent and convolutional neural networks that process each word in a sentence in parallel and does not have the ability to recognize the order of each word, the addition of positional encoding can help the translation model to recognize the positional information of the words in a sentence. The dimension of positional encoding is the same as that of word vectors, and the result of adding the two can be used as the input sequence.Transformer calculates positional encoding by using sine and cosine functions with different frequencies, see Eq. (14), Eq. (15).

$$PE_{(POS,2i)} = \sin(pos / 10000^{2i/d_{model}}) \tag{14}$$

$$PE_{(POS,2i+1)} = \cos(pos / 10000^{2i/d_{model}}) \tag{15}$$

Where: pos is the absolute position of the element in the sequence, $2i$ is the current dimension of the position encoding vector, and d_{model} denotes the dimension of the input sequence.

The Transformer model architecture is shown in Figure 1.

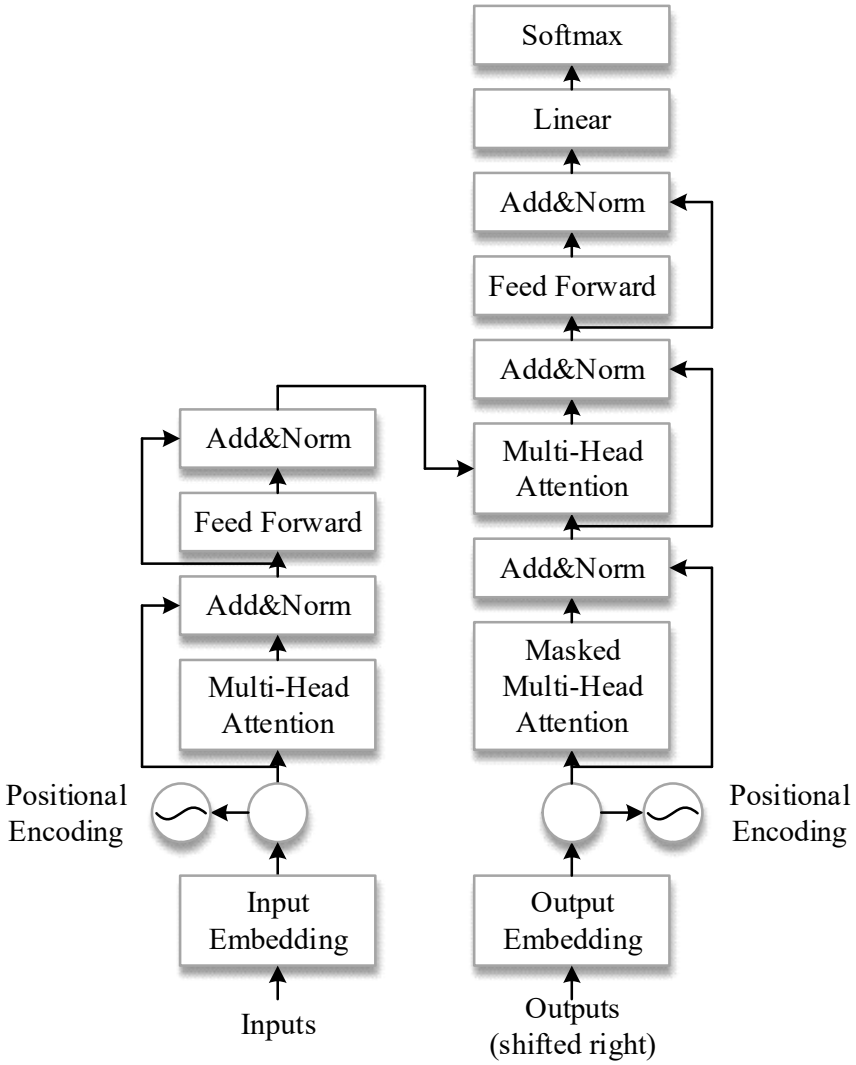

Fig. 1. Transformer model architecture

4. Translation models incorporating syntactic knowledge and terminology

4.1. Translation models incorporating syntactic knowledge and terminology

In a dependency syntax tree, each node may have one or more dependencies that describe the syntactic relationships between that node and other nodes. Typically, the importance of a node can be measured by its position in the dependency syntax tree and the number of dependencies it has. A node is usually considered to be a more important node if it has more dependencies in the dependency syntax tree. This is because this node has a more complex role in the sentence, it may play more grammatical roles such as subject, object, determiner, etc., and also more information flows to and from this node. Therefore, in dependency syntactic analysis, the importance of each node in a sentence can be determined by looking at the number of dependencies of each node. When a node has more dependencies, it can be considered more important in the sentence.

Therefore, according to the domain characteristics of English translation data, this paper incorporates syntactic keywords into the domain translation training process, which mainly consists of two parts: the left half is the basic model structure of Transformer, which is responsible for realizing the translation function from source language to target language; the right half is the dependency syntactic structure, which is obtained using the relevant tools provided by Stanford CoreNLP [24]. Then the out-degree of each token is calculated by the syntactic structure, and the keywords are

selected by setting a threshold according to the out-degree, and the keywords are input to the encoding side of the Transformer for guided encoding.

1) Input Module. Given a piece of text $X=\{x_1,x_2,...,x_i,...,x_n\}$, this is the text that has been tokenized by BPE, where x_i represents the ird token in the sentence. For example, "Institution of systematic searches of baggage and passengers at airports" gets "Institution of systematic searches of b@@agg@@age and passengers at air@@port@@s" after BPE participle. Among them, the input of the translation model proposed in this paper includes two parts: the text after BPE word segmentation and the keywords obtained according to the dependent syntactic tree.

The text that was not subjected to BPE disambiguation was obtained into a dependent syntactic tree structure using the CoreNLP tool.

According to the relevant definitions in directed graphs, the in-degree is defined as the number of edges with the vertex as the head; the out-degree is defined as the number of edges that refer to the vertex as the tail. In this paper, we consider a node to be more important in a dependent syntactic tree if the number of points pointed out by that node is greater, i.e., the out-degree is greater. In this paper, a node whose out-degree number is greater than a certain threshold is called a keyword.

Then the out-degree of each word is calculated based on the dependency syntax tree structure.

The input of the Transformer model is the text processed by BPE, firstly, we use the open source tool to process the data to get the dependency analysis results and the structure of the dependency syntax tree, and then we count the number of out-degree of each word to construct the list of out-degree numbers. However, at this time, this list of degrees can not be used as a token keyword extraction, the text is input to the Transformer encoder is split into a sequence of tokens, at this time, the need to analyze the results of the original text in the list of degrees and the results of the word division to get a new list of degrees.

According to the aligned list of token counts, combined with different threshold num values for screening, from which the keyword tokens are selected, and then the selected keywords are input to the Encoder side.

The input at the Encoder side consists of three specific parts, which include text encoding, position encoding PE, keyword encoding, and input information fusion:

$$sen\_emb_i = emb(x_i, vocab\_size) \tag{16}$$

where x_i denotes the ind token split in the text and $vocab\_size$ denotes the lexicon size.

$$keyword\_emb_i = emb(k_i, vocab\_size) \tag{17}$$

where k_i denotes the ind keyword of all syntactic keywords and $vocab\_size$ denotes the word list size.

$$In\_En = (sen\_emb + PE) : keyword\_emb \tag{18}$$

where PE denotes the positional encoding of the data.

2) Encoding Module. This module describes the Encoder encoding structure after extraction based on keywords.

The attention mechanism selects the relevant information directly from the source language information as an aid, and the self-attention score at the Encoder side of the Transformer is calculated as shown in Equation (19):

$$A = soft\max\left(\frac{QK^T}{\sqrt{d_k}}\right) \tag{19}$$

After Transformer initializes Q, K so that it is equal to $\ln\_En$, and after incorporating the syntactic keywords $\ln\_En$ after the deformation, the dimension of the Q, K matrix changes from the original

$n_s \times d$ to $(n_s + n_k) \times d$, where n_s denotes the length of the sentence, n_k denotes the length of the keyword, and d denotes the dimension of the feature at each position.

Where the lower left part of the dashed line is the original shape of Transformer's self-attention score, and the other parts are the new attention after incorporating syntactic keywords.

In addition, the syntactic positional information is the positional information of that node that points to the node in the dependent syntactic tree, i.e., the positional information of its parent node. In this paper, the obtained syntactic analysis results are traversed so as to obtain the position information of the parent node pointing to the current node in the sentence, and the obtained position information of the parent nodes of all the nodes are later added behind the source language and input to the model for training, thus increasing the possibility of the model to learn syntactic knowledge.

4.2. Translation model incorporating lexical features

In ordinary English context, a word may correspond to multiple translations or even to multiple lexemes. In order to ensure the unambiguity of words in scientific and technical literature, Simplified English stipulates that some words can only be of a single lexical gender, e.g., some words can only be verb lexical, some words can only be nouns, and some words can only be adjectives and so on. Therefore, correctly identifying the lexical properties of each word in the frequency-space data and inputting them into the NMT for training helps the model to learn the information embedded inside it.

In this paper, we first use the relevant tools provided by CoreNLP to obtain the lexical information of each word, fuse the obtained lexical information with the data, and finally feed the new data with the lexical information into the model for training, so that the model can learn the lexical features of the data.

4.3. Translation Model for Expanding Domain Data with Terminology

Since NMT requires a large number of bilingual parallel corpora to train the model, the number of bilingual parallel corpora within a specific domain is most sparse and insufficient to meet the training requirements. In order to more accurately reflect the characteristics of the English translation domain data, and in order to verify the effect of adding additional English translation terms with the smallest number of maxima on the model performance, this paper tries to add English translation terms into the translation model. In this paper, there are more than 30,000 pairs of English translation bilingual terms, which are added according to different ratios and mixed with the original English translation data, and then integrated into the English translation model, respectively, to train different translation models and test their translation performance.

In this paper, we use the syntactic analysis tool to syntactically analyze the English translation data and the English translation terms respectively, and perform depth-first traversal of the obtained syntactic analysis results to obtain the linear sequence of the syntactic tree. Then the out-degree number of each node is obtained according to the obtained syntactic tree, and the syntactic keywords are selected according to the out-degree number. Finally, the data, terms, linear syntactic sequence and syntactic keywords are spliced correspondingly and fed into the NMT model for training.

5. Experimental analysis of models incorporating syntactic knowledge and terminology

5.1.Experimental data set

In this paper, we use Multi30k, a dataset commonly used in the field of machine translation, which contains about 30,000 parallel pairs of utterances in English and German for specialized terminology. Although the dataset is small in data volume, it focuses on common vocabulary and epitomizes a million-volume machine translation dataset. Randomly extracting data from the million-size utterance pairs, the data results obtained cannot highlight the key vocabulary, so we cannot get more accurate

experimental results. Therefore, under the comprehensive consideration of computer power and model effect, Multi30k is used as the main experimental dataset in this paper. In this experiment, German is used as the source language and English as the target language, and the translation from the source language to the target language is realized through the Transformer model.

5.2.Indicators for testing and evaluating the model

Due to the complexity of sentence structure and the diversity of vocabulary, more than one correct translation can be obtained for a given sentence. Different linguistic sequences may have been chosen and different vocabularies may have been used between these translations. In order to measure the quality of translations, the BLEU score is used as a "numerical indicator" to effectively measure the degree of similarity between the translations obtained by machine translation and those obtained by human translation.

The BLEU method is a similarity measure based on word accuracy, which can characterize the degree of co-occurrence of n-unit segments in machine translations and reference translations. The specific calculation method is:

$$BLEU = B_p * e^{\sum_{n=1}^{N} w_n \log P_n} \tag{20}$$

The n-unit segments of the machine translation and the reference translation reflect the adequacy of the translation, and the number of matching segments reflects the fluency of the translation. It is based on the above calculation logic that the BLEU method is fast in calculation, easy to understand, not limited by language, and widely used in various machine translation experiments.

The higher BLEU score represents the better effect of the machine translation model, through the reading of the literature, it is known that the BLEU score of a better machine translation model is generally higher than 25, and in this paper, we will also choose the BLEU score to evaluate the effect of the machine translation model, and constantly improve its value.

5.3.Analysis of model optimization results

Twenty rounds of training and validation were performed for the two models using dynamic and static position coding, respectively. The cross-entropy loss on the training and validation sets for each round was recorded, and the variation of cross-entropy loss versus the number of trainings for different positional coding methods is shown in Fig. 2. The cross-entropy loss under static coding is slightly smaller under the two different positional coding methods.

Overall, the cross-entropy loss on the training set shows a gradual decrease during the training process of the model, while the cross-entropy loss on the validation set shows a trend of decreasing and then increasing. The loss on the validation set reaches the minimum after 8 rounds of training for both different models, which indicates that too many rounds of training for the model will fully extract the information of the training set, and there will be overfitting on the validation set, so setting the appropriate number of training times can avoid ineffective training at a later stage, which saves the training time. Therefore, in the experiments of the subsequent chapters, the training rounds of the model are set to 10 times to reduce unnecessary training rounds and save the training time.

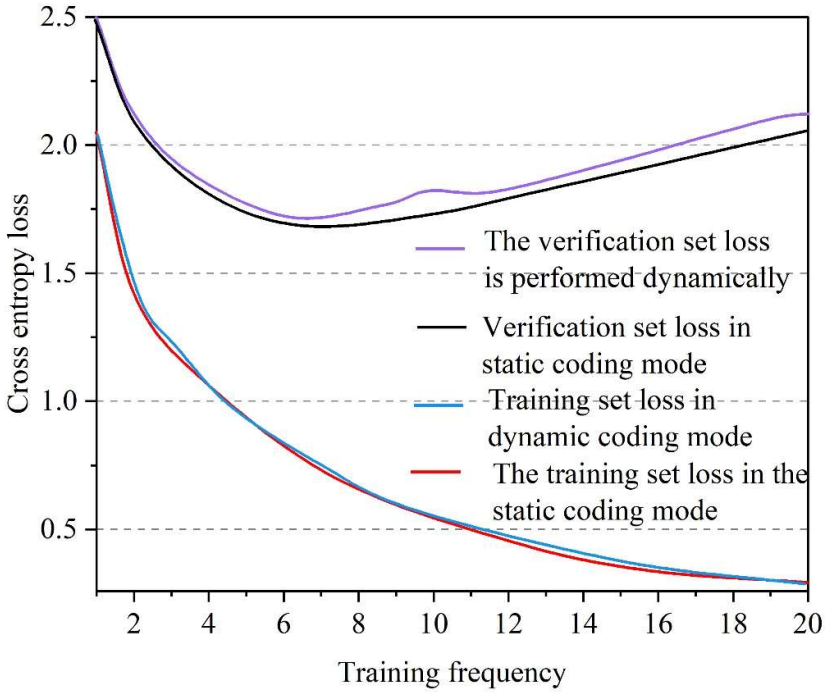

Fig. 2. Cross entropy loss in different location coding mode

The parameter that loads the model with the least cross-entropy loss on the validation set is tested using both models and the cross-entropy loss and BLEU scores obtained on the test set are shown in Table 1. Comparison of the results in Table 1 shows that the Transformer model with static positional coding approach has lower cross-entropy loss and higher BLEU scores on the test set, so the static positional coding approach is used in all the subsequent models, but this approach leads to a significant increase in the model training parameters.

Table 1. Cross entropy loss and BLEU score of different models in the test set

Model type	Cross entropy loss	BLEU score
Dynamic location coding	1.78	28.59
Static location coding	1.70	29.83

Finally, the optimal parameters in the Transformer model of static positional coding are loaded, and the greedy algorithm is implemented to predict the single corpus sentences, the input source language is “Ein paar sitzt mit baby und sportwagen im gras.”, the real translation is The real translation is “A couple sit on the grass with a baby and stroller”, which is translated as “A couple sitting in the grass with their baby.”

By drawing the cross-attention mechanism in the last decoder of the decoding layer, the cross-attention weight matrix of the eight heads of the input source language to the translation language is obtained, and the cross-attention weight matrix of the eight heads is shown in Figure 3-Figure 10, respectively.

The horizontal axis is the source language of the input, and the vertical axis is the target language term predicted by the algorithm, and the darker the color at the intersection of the two, the more affected the current output prediction word is by the input word at the corresponding position. In the first 7 heads, the intersection of "peer" in the source language and "couple" in the prediction word is darker, indicating that the source language has a strong influence on the vocabulary and is correctly translated. However, there are also areas where the performance is not very good, such as in the cross-attention weight matrix of the 1st, 2nd, 4th, 6th, and 8th heads, where the intersection of "sportwagen" in the source language and "their" in the predicted target language is darker, and the translation is

actually wrong. This phenomenon is mainly caused by the overfitting of the model or insufficient information extraction, which is also the room for model improvement. On the whole, the translation model of this paper performed well, and the accuracy of the English translation of professional terms was good.

Based on the above analysis, in order to avoid the problem of noise interference caused by model overfitting and insufficient extracted information, this paper considers to propose a solution: the LSTM model can be added to the input layer in order to solve the problem of noise interference caused by long-term memory through long and short-term memory.

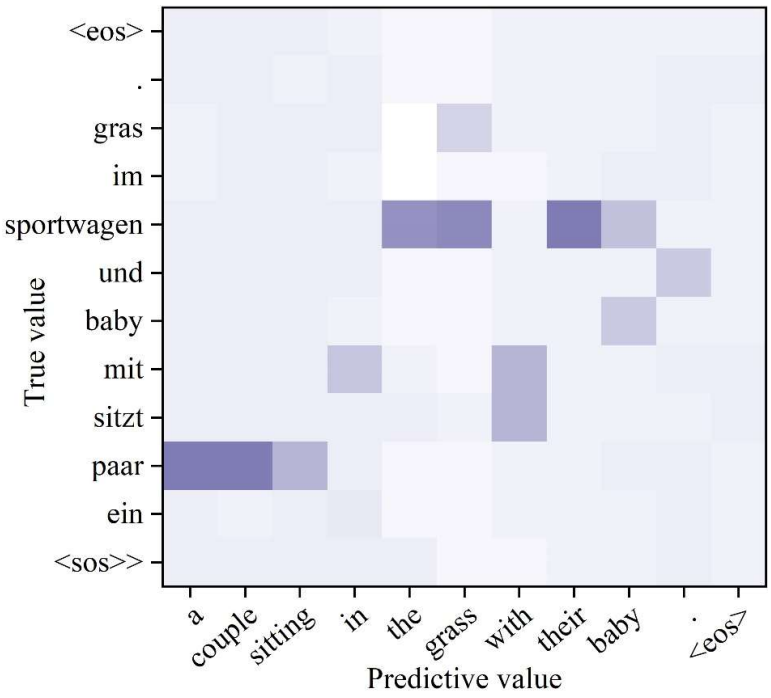

Fig. 3. The first head's cross-attention weight matrix

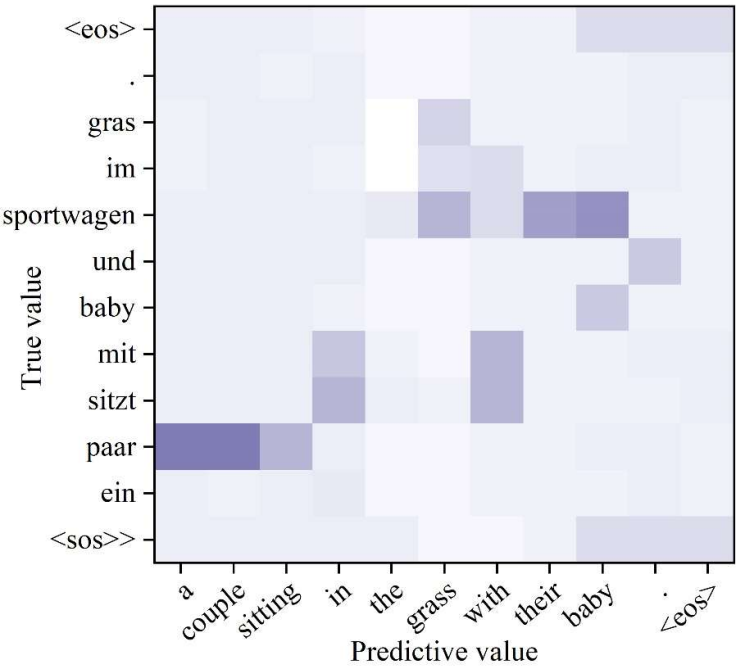

Fig. 4. The second head's cross-attention weight matrix

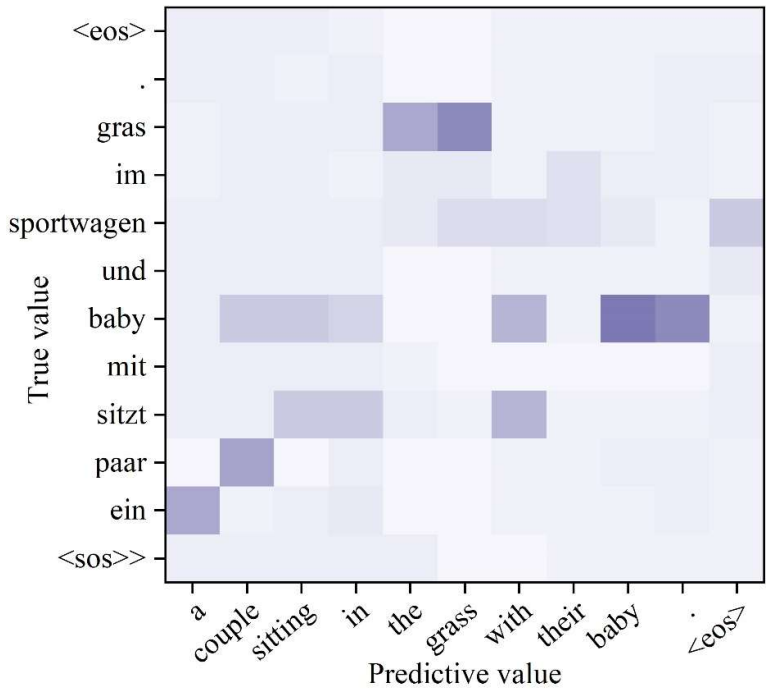

Fig. 5. The third head's cross-attention weight matrix

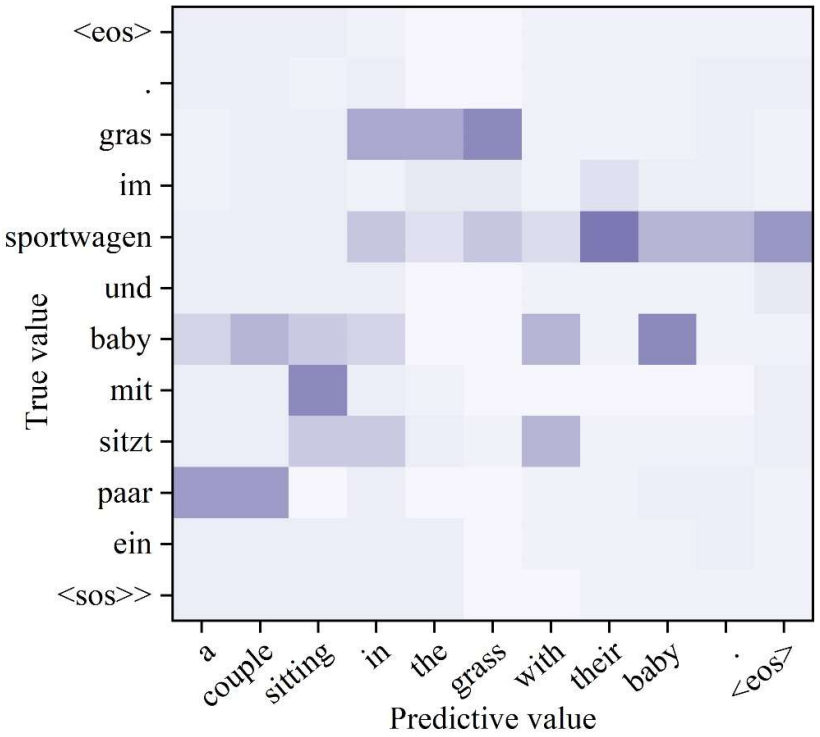

Fig. 6. The fourth head's cross-attention weight matrix

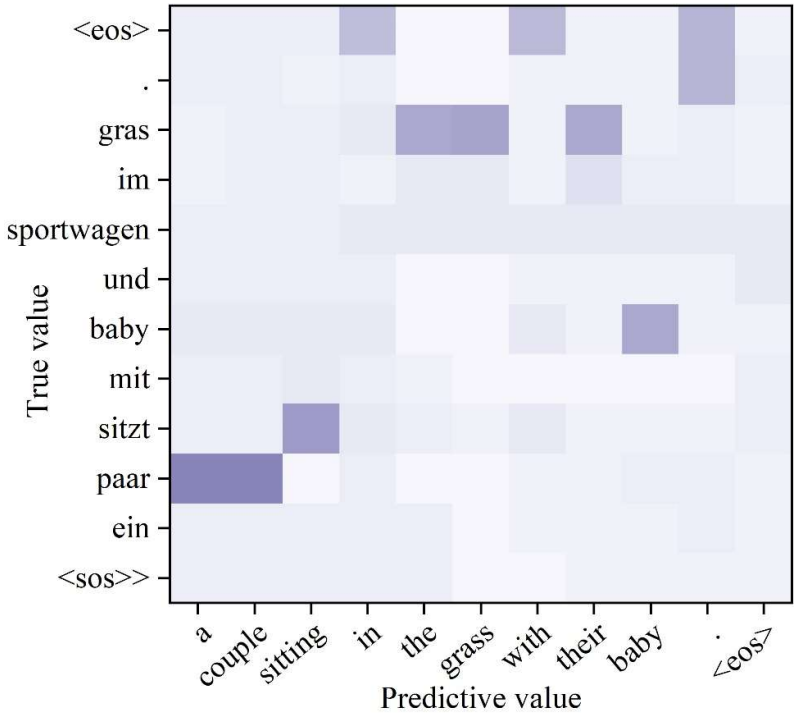

Fig. 7. The fifth head's cross-attention weight matrix

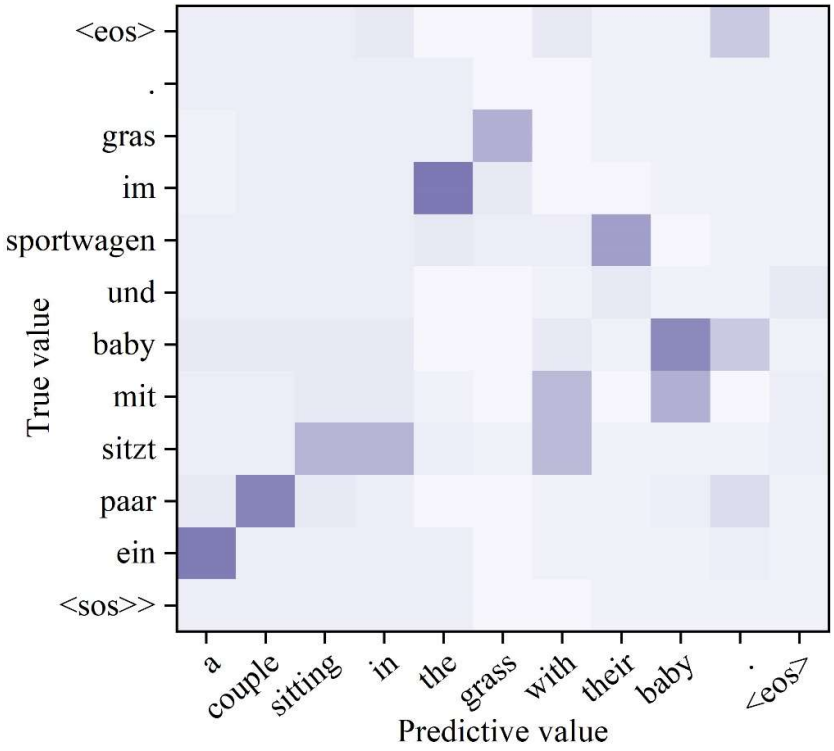

Fig. 8. The sixth head's cross-attention weight matrix

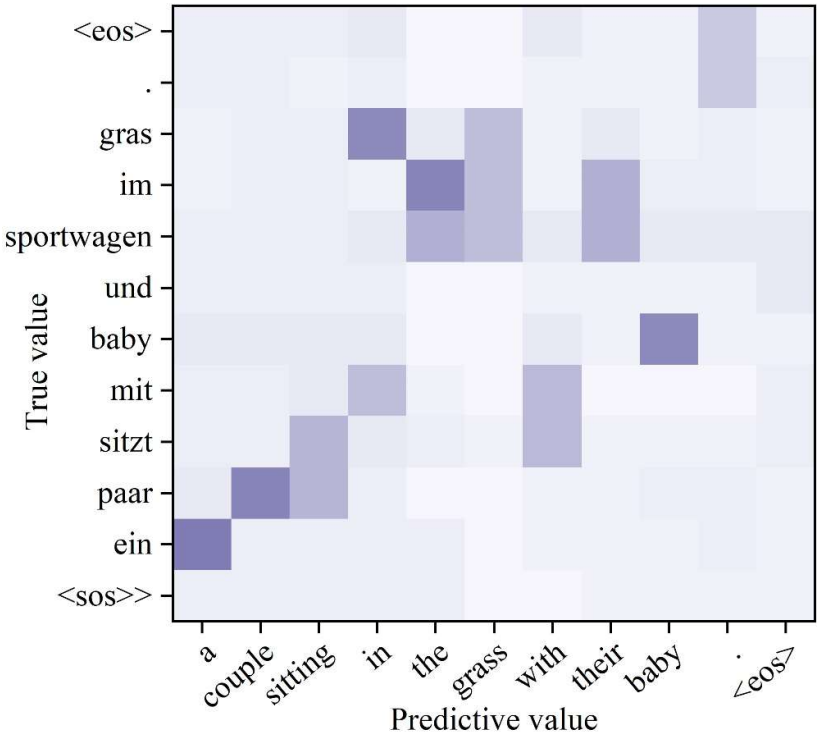

Fig. 9. The seventh head's cross-attention weight matrix

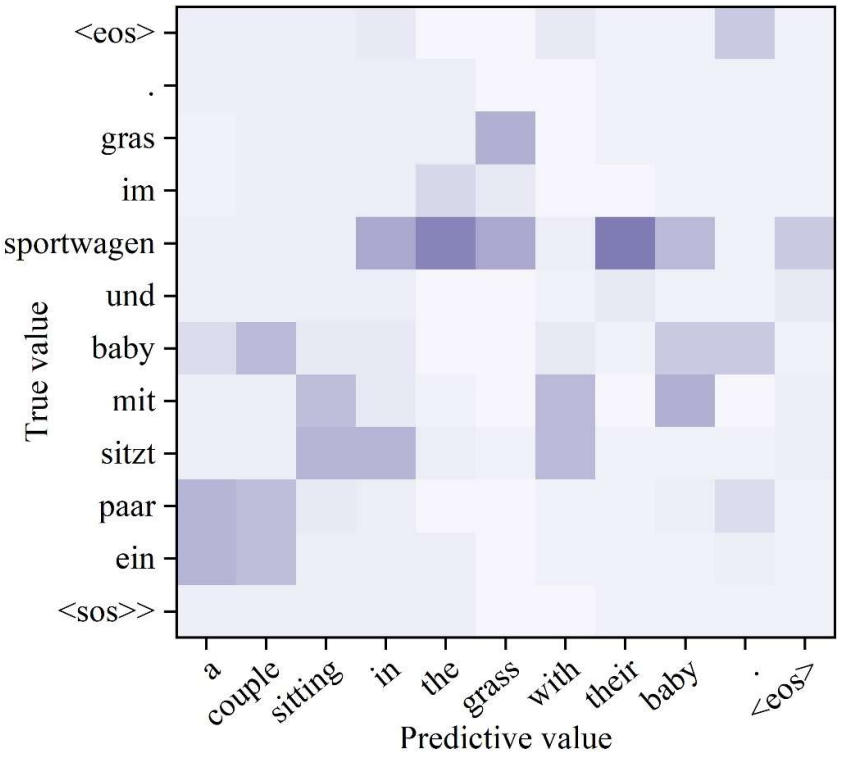

Fig. 10. The eighth head's cross-attention weight matrix

The trend of loss curves during training is similar, so in this subsection a comparative analysis will be performed based on the loss curve graphs of the De→En terminology translation task. In the comparative analysis, the baseline model proposed in this paper and the proposed optimized model are used for comparison. The comparative analysis shows more clearly the improvement effect of the optimization method on the model. Since the stabilized curves are basically the same, in order to better

see the difference in model performance, the loss values located in the front part are intercepted and plotted, and the loss values are shown in Figure 11.

Located at the bottom is the loss curve graph of the optimized model, and the other curve is the change graph of the loss curve of the base model. From the trend of the graph, it can be seen that the change of the optimized model curve will be a little steeper than the change curve of the base model, which indicates that the learning rate of the optimized model will be a little faster than the base model. Therefore, with the same model parameters and hardware facilities, the optimized model can learn more in a shorter period of time, and when the model converges, the optimized model will converge faster, the optimized model converges at about 1500 steps, while the base model converges at 2000 steps. Overall, the introduction of syntactic knowledge and terminology improves the efficiency of the model's translation learning, allowing the model to learn the same content faster in the same amount of time, thus increasing the rate of terminology translation.

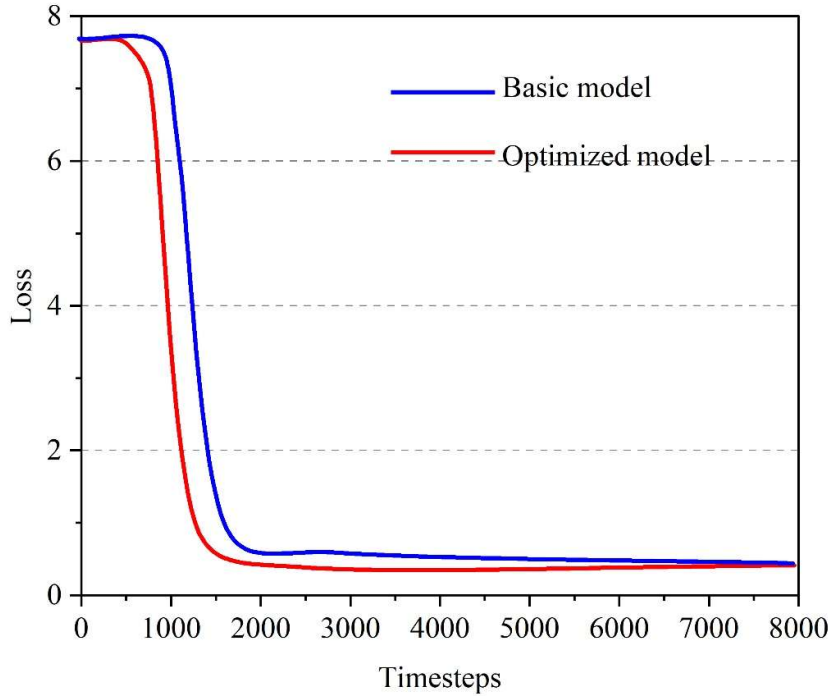

Fig. 11. Model loss comparison

The model of this paper is trained and tested based on the collected datasets of German-English (De-En) and English-German (En-De), and the output translation results are compared and analyzed with the base model, and the recorded results are shown in Table 2. The introduction of syntactic knowledge and terminology will improve the translation quality of the model, and the accuracy is improved on each dataset, which side by side shows richer text information, which is favorable for the model to obtain more effective information, and will help the model to better predict the target text. From the optimization results of the model, syntactic keywords, lexemes, and terms acting on the model alone can improve the BLEU value of the base model, but the joint combination of keywords, lexemes, and terms with the base model has the most obvious effect on the improvement of the BLEU value of the model, and the optimized model of this paper improves the BLEU value of the model by 22.67 BLEU values in the task of translating the De→En terminology, relative to the base model.

Table 2. Translation quality analysis

Model	En→De	De→En
Basic model	53.45	57.45

+ syntax keywords	58.54	63.25
+ speech	59.64	68.77
+ term	62.48	72.54
+ syntax keywords + words + terms	69.58	80.12

5.4.Correlation Analysis of Translated Sentence Length

This paper further investigates the translation speed and translation quality of the final translation model and the autoregressive standard Transformer under different sentence lengths. First, this paper groups the test set of Multi30k dataset according to the source sentence length, and each group corresponds to a sentence length interval length of 10 (except for the last group where the upper limit of the interval is set to be the length of the longest source sentence), and the number of samples in each length interval is counted as shown in Table 3.

Table 3. Length interval sample number

Length interval	Sample number
(0, 10]	244
(10, 20]	985
(20, 30]	897
(30, 40]	524
(40, 50]	258
(50, 60]	135
(60, 70]	48
(70, 80]	15
(80, 90]	6
(90, 104]	3

Then, based on the statistical results, the intervals containing more samples (intervals of length 0~60) are regarded as common sentence length intervals in this paper, and experiments are carried out on the subgroups corresponding to these common sentence length intervals. Specifically, with respect to translation speed, the time to complete the translation of the whole group of data by the translation method of this paper and the Transformer method (with the bundle search size set to 4) is calculated in each group, and then the speed-up ratio between the two is calculated. Regarding the translation quality, the BLEU scores are calculated on each subgroup separately. The results are shown in Fig. 12, and in this paper, we will analyze the difference of the acceleration ratio on different length intervals. It should be noted that, due to the large difference in the number of samples between intervals, in order to make a fair comparison, this experiment sets a fixed batch_size=128, and then counts the average translation time of each batch on each subgroup, so as to be able to fully utilize the samples of various lengths for the statistics and to eliminate the effect of batch division.

As the sentence length increases, the speedup ratio of this method relative to the Transformer method increases, which means that the speedup advantage of this method expands with the increase of sentence length, and this result shows that the performance of the Transformer translation method is indeed significantly affected by the need for word-by-word generation. In addition, observing the BLEU results of each group, it can be found that the fluctuation of translation quality with sentence length is small, which indicates that the optimization method in this paper is more robust to sentence length.

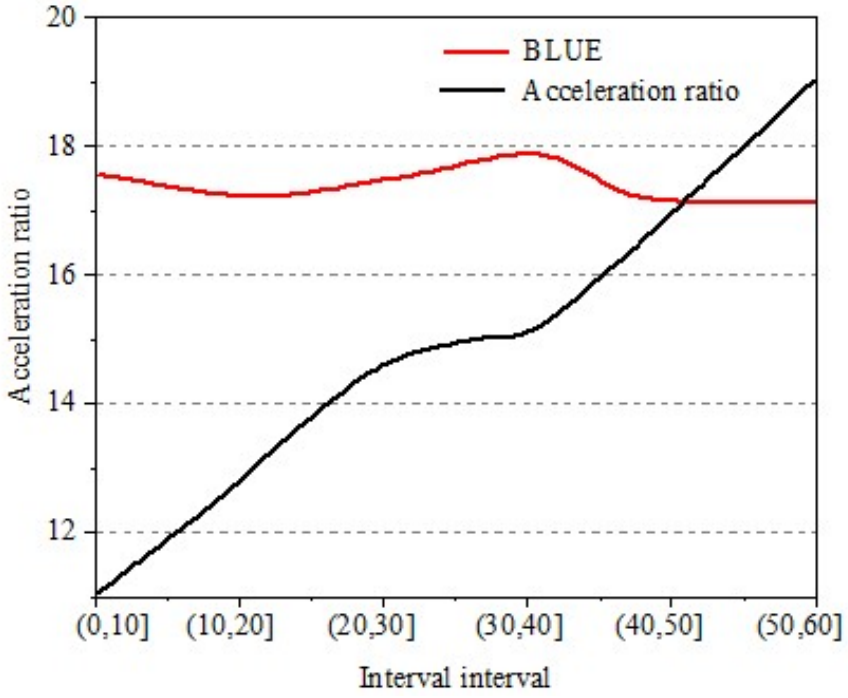

Fig. 12. The translation speed and translation quality of different sentences

6. Conclusion

In order to make full use of the specialized terminology in the English translation process and improve the accuracy of machine translation, this paper proposes a machine translation method that incorporates English syntactic knowledge and terminology.

The Multi30k dataset is used for experiments. It is proved that a training number of about 10 times should be appropriate in the subsequent experiments, and the model uses the static position encoding method for English translation better than the dynamic position encoding method. Finally, a single German sentence was translated using the English translation model of this paper, and the cross attention weight matrix of the 8 heads of the last layer of the cross attention matrix was plotted, according to which it was confirmed that the accuracy of the optimized translation model of this paper for the English translation of technical terms was at a more excellent level.

The optimized translation model makes full use of the textual information in translation based on the fused English syntactic knowledge and terminology. Through a large number of experiments, it is proved that the optimized model of professional terminology English translation proposed in this paper improves the translation performance, which not only accelerates the learning of the model, but also optimizes the performance of the model. It makes the translation model of this paper improve 22.67 BLEU values on the De→En translation task, which is far better than the base model. In addition, the fluctuation amplitude of the BLEU value of this paper's method with the sentence length is also observed, and the overall fluctuation range is small, which indicates that the robustness of this paper's optimization method on the sentence length also performs extremely well.

Funding

This research was supported by the 2024 Key Project of Humanities and Social Sciences Research in Anhui Province's Universities: “Research on the External Promotion and Translation of Cultural Heritage in Maanshan Area from the Perspective of Ecological Translation Studies” (2024AH053311).

References

[1] Specia, L., Scarton, C., & Paetzold, G. H. (2018). Quality estimation for machine translation. Morgan & Claypool Publishers.

[2] Lee, S. M. (2023). The effectiveness of machine translation in foreign language education: a systematic review and meta-analysis. Computer Assisted Language Learning, 36(1-2), 103-125.

[3] Freitag, M., Foster, G., Grangier, D., Ratnakar, V., Tan, Q., & Macherey, W. (2021). Experts, errors, and context: A large-scale study of human evaluation for machine translation. Transactions of the Association for Computational Linguistics, 9, 1460-1474.

[4] Castilho, S., Doherty, S., Gaspari, F., & Moorkens, J. (2018). Approaches to human and machine translation quality assessment. Translation quality assessment: From principles to practice, 9-38.

[5] Bowker, L., & Fisher, D. (2010). Computer-aided translation. Handbook of translation studies, 1, 60-65.

[6] Zanettin, F. (2017). Issues in computer-assisted literary translation studies. Intralinea. Special Issue: Corpora and Literary Translation Issues in Computer-Assisted Literary Translation Studies 2017.

[7] Shadiev, R., Chen, X., & Altinay, F. (2024). A review of research on computer‐aided translation technologies and their applications to assist learning and instruction. Journal of Computer Assisted Learning, 40(6), 3290-3323.

[8] Karpińska, P. (2017). Computer Aided Translation–possibilities, limitations and changes in the field of professional translation. The Journal of Education, Culture, and Society, 8(2), 133-142.

[9] Ningrum, D. K. S., Sofyan, R., & Saragih, E. (2024). Exploring Linguistic Competence in Academic Text Translation by Professionals. Language Literacy: Journal of Linguistics, Literature, and Language Teaching, 8(1), 301-314.

[10] Tursunovich, R. I. (2022). Linguistic and cultural aspects of literary translation and translation skills. British Journal of Global Ecology and Sustainable Development, 10, 168-173.

[11] Vieira, L. N., O'Hagan, M., & O'Sullivan, C. (2021). Understanding the societal impacts of machine translation: a critical review of the literature on medical and legal use cases. Information, Communication & Society, 24(11), 1515-1532.

[12] Way, A. (2018). Quality expectations of machine translation. Translation quality assessment: From principles to practice, 159-178.

[13] Cadwell, P., o'Brien, S., & Teixeira, C. S. (2018). Resistance and accommodation: factors for the (non-) adoption of machine translation among professional translators. Perspectives, 26(3), 301-321.

[14] Awadh, A. N. M. (2023). Utilization of Computer-Assisted Translation Tools Among Arab Translators: Scope, Challenges, and Solutions. Theory and Practice in Language Studies, 13(11), 2743-2754.

[15] Çetiner, C. (2018). Analyzing the attitudes of translation students towards cat (computer-aided translation) tools. Journal of Language and Linguistic Studies, 14(1), 153-161.

[16] Xiang, Y., Chen, Y., Fan, W., & Ye, H. (2024). Enhancing computer-aided translation system with BiLSTM and convolutional neural network using a knowledge graph approach. The Journal of Supercomputing, 80(5), 5847-5869.

[17] Zhao, T., & Alias, M. B. (2024). Automated programming approaches to enhance computer-aided translation accuracy. PeerJ Computer Science, 10, e2396.

[18] Zhu, M., & Jiang, L. (2024, September). Effects of the model of combining computer-aided translation with post editing on the English translation of Cantonese Opera terms. In Proceedings of the 2024 International Symposium on Artificial Intelligence for Education (pp. 89-93).

[19] Popel, M., Tomkova, M., Tomek, J., Kaiser, Ł., Uszkoreit, J., Bojar, O., & Žabokrtský, Z. (2020). Transforming machine translation: a deep learning system reaches news translation quality comparable to human professionals. Nature communications, 11(1), 1-15.

[20] Dugonik, J., Bošković, B., Brest, J., & Sepesy Maučec, M. (2019). Improving statistical machine translation quality using differential evolution. Informatica, 30(4), 629-645.

[21] Park, C., Park, K., Moon, H., Eo, S., & Lim, H. (2021). A study on performance improvement considering the balance between corpus in neural machine translation. Journal of the Korea Convergence Society, 12(5), 23-29.

[22] Shuqiang Huang,Cuiyu Tan,Jinzhen Zheng,Zhugu Huang,Zhihong Li,Ziyin Lv... & Qiuxia Yan. (2024). Identification of RNMT as an immunotherapeutic and prognostic biomarker: From pan-cancer analysis to lung squamous cell carcinoma validation.. Immunobiology(5),152836.

[23] Sadeghian Ehsan,Dragomirescu Elena & Inkpen Diana. (2025). Damage Detection for a Cantilevered Steel I-Beam through Deep-Learning Methods: LSTM, Multivariate Time-Series Transformer, and LSTM-Based Autoencoder. Journal of Computing in Civil Engineering(2),

[24] Alessio Palmero Aprosio & Giovanni Moretti. (2016). Italy goes to Stanford: a collection of CoreNLP modules for Italian.. CoRR