Computers, Materials & Continua
DOI: 10.32604/cmc.2023.032732
Article

Neural Machine Translation by Fusing Key Information of Text

Shijie Hu[1], Xiaoyu Li[1,*], Jiayu Bai[1], Hang Lei[1], Weizhong Qian[1], Sunqiang Hu[1], Cong Zhang[2], Akpatsa Samuel Kofi[1], Qian Qiu[2,3], Yong Zhou[4] and Shan Yang[5]

[1]School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu, 610054, China
[2]Science and Technology on Altitude Simulation Laboratory, Sichuan Gas Turbine Establishment Aero Engine Corporation of China, Mianyang, 621000, China
[3]School of Power and Energy, Northwestern Polytechnical University, Xi'an, 710072, China
[4]School of Computer Science, Southwest Petroleum University, Chengdu, 610500, China
[5]Department of Chemistry, Physics and Atmospheric Sciences, Jackson State University, Jackson, MS 39217, USA
*Corresponding Author: Xiaoyu Li. Email: xiaoyuuestc@uestc.edu.cn
Received: 27 May 2022; Accepted: 12 July 2022

Abstract: When the Transformer proposed by Google in 2017, it was first used for machine translation tasks and achieved the state of the art at that time. Although the current neural machine translation model can generate high quality translation results, there are still mistranslations and omissions in the translation of key information of long sentences. On the other hand, the most important part in traditional translation tasks is the translation of key information. In the translation results, as long as the key information is translated accurately and completely, even if other parts of the results are translated incorrect, the final translation results' quality can still be guaranteed. In order to solve the problem of mistranslation and missed translation effectively, and improve the accuracy and completeness of long sentence translation in machine translation, this paper proposes a key information fused neural machine translation model based on Transformer. The model proposed in this paper extracts the keywords of the source language text separately as the input of the encoder. After the same encoding as the source language text, it is fused with the output of the source language text encoded by the encoder, then the key information is processed and input into the decoder. With incorporating keyword information from the source language sentence, the model's performance in the task of translating long sentences is very reliable. In order to verify the effectiveness of the method of fusion of key information proposed in this paper, a series of experiments were carried out on the verification set. The experimental results show that the Bilingual Evaluation Understudy (BLEU) score of the model proposed in this paper on the Workshop on Machine Translation (WMT) 2017 test dataset is higher than the BLEU score of Transformer proposed by Google on the WMT2017 test dataset. The experimental results show the advantages of the model proposed in this paper.

Keywords: Key information; transformer; fusion; neural machine translation

1 Introduction

Neural machine translation (NMT) is a task of translating text from source language to target language. Neural machine translation was first proposed by Nal et al. [1], Sutskever et al. [2] and Cho et al. [3]. Unlike the traditional phrase-based translation system (Koehn et al. [4]) which consists of many small sub-components that are tuned separately, neural machine translation is one of the ultimate goals of artificial intelligence that committed to help people complete the translation task, and gradually replaces human beings to complete the complicated and time-consuming translation work.

As early as the 1930s and 1940s, people began the research on machine translation. With continuous breakthroughs in research, the research on machine translation technology has gradually shifted from a translation system (based on vocabulary, grammar and other rules) to a statistical-based machine translation, and then to the current research, the neural machine translation which is on the hot. The task of neural machine translation is mainly to use neural network related methods and a large amount of data for training and get a general translation model [5]. After the model is trained, we only need to input the source language sentence into the given model, and the model can get the corresponding translation result by performing the calculation.

In the current field of machine translation, neural machine translation has become the mainstream method and paradigm in researches and applications. The proposal of transformer has detonated this field. In 2017, Vaswani et al. [6] proposed the Transformer model, compared with sequence to sequence model, this model has better experimental performance in NMT, and compared with traditional Recurrent Neural Network (RNN) [7], it has greatly improved training efficiency. Since Transformer was put forward, it has been keeping attracting attention. So far, Transformer has been adopted by a variety of natural language processing (NLP) models, and many researchers have also made many innovative improvements on this basis. For instance, Sukhbaatar et al. proposed the Adaptive-Span Transformer [8] which optimizes the calculation efficiency of transformer.

In recent years, syntactic-based neural machine translation [9] has become a hot topic in neural machine translation research. Existing works [10–14] have shown that incorporating linguistic information into the translation model can greatly improve the performance of the model. Although neural machine translation has made great achievements, there are also translations that are fluent but not faithful enough [15], difficult to process rare words, poor performance in low-resource languages, poor cross-domain adaptability, low utilization of prior knowledge, mistranslations and missed translations [16], etc. Inspired by the classic statistical machine translation research, it has become a hot topic in the field of neural machine translation research that using existing linguistic knowledge, incorporating linguistic information into the neural machine translation model [17], alleviating the inherent difficulties faced by neural machine translation, and improving translation quality [18].

Among these issues, this paper has carried out a research which focusing on mistranslations and omissions. When the traditional machine translation model completes the translation work, there are often mistranslations and missing translations of the keywords in the source text. This problem greatly reduces the quality of the results translation. Due to the lack of interpretability of neural networks, it is difficult to explain how these omissions and mistranslations occur and how to design methods to eliminate them.

Specifically, in the translation process under the "seq2seq + attention" framework, the "attention" of translating the current vocabulary and the "attention" of translating the vocabulary before it are independent, so the current operation cannot obtain alignment-related [19] information from the previous translation information, this has led to the problems of "over translation" and "missing translation", and this problem becomes more prominent as the length of the source text increases. In this regard, we have made an adjustment to the traditional seq2seq structure and Transformer. In view of the current machine translation model's problems in the translation of the key information of long sentences, based on the Transformer model, this paper proposes a model improvement method by fusing key information, that is, the key information of each source language sentence is encoded by an independent encoder based on the self-attention mechanism, then give the corresponding weight, combine this weight with the Transformer model. Compared with the Transformer model, this method has a better improvement, and improves the BLEU [20] score of the model in the WMT Chinese-English translation task, which proves the effectiveness of the model.

The main work of this paper is as follows:

1) In order to solve the problem of mistranslation and omission in the translation of the key information of long sentences in the existing machine translation model, a method of encoding by fusing the keyword information in the source language text is proposed, which can effectively improve the accuracy of the model's translation of long sentences.
2) Based on the transformer model, a new model structure is proposed, in brief, an encoder for encoding key information of the source language text is added on the basis of transformer. Besides, this paper proposes a new way of fusing key information, in which the source language text and key information are separately encoded and then fused, and the fused input is used to train the model. The model performs better than traditional models in translation tasks.
3) Design related experiments to verify the performance of the model proposed in this paper on the public data set, and compare it with other benchmark models.

2 Model Structure Fusion Key Information

The classic NMT model usually uses a sequence to sequence model which has an encoder and a decoder, and the input is the word sequences of the source language text $S = [s_1, s_2, \ldots, s_m]$, the output is the word sequences of the target language text $T = [t_1, t_2, \ldots, t_n]$. The encoder stack of the NMT model that integrates key information proposed in this paper is shown in Fig. 1, the left branch of fusion encoder is an encoder for key information, and the right branch is an encoder for source language text. As shown in Fig. 1, the stack takes the key information of the source language text and the source language text as input respectively. After encoding by fusion encoder, the outputs are fused and input into the decoder.

Inspired by Google's multi-head attention model, in order to associate key information with the source language text, we used N layers of multi-head self-attention to encode key information ($N = 6$ in the experiment), do the same for the source language text, then calculate the result of the correlation between the two. Finally, the association result is combined with the source language text encoding result that as the input of the multi-head attention in the decoder which is associated with the coding information of the target language text to complete the fusion of key information. The detailed model structure proposed in this paper is shown in Fig. 2.

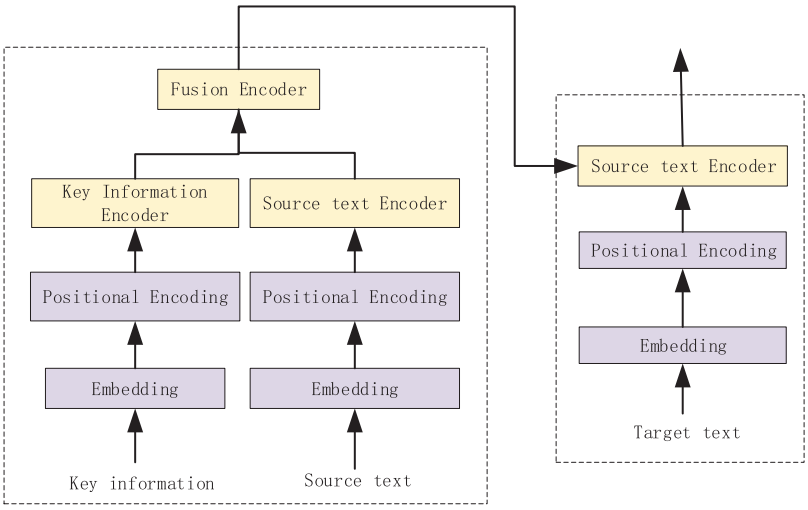

Figure 1: General structure of model

The specific calculation process of multi-head attention is shown in Eq. (1), in practice, we compute the $MultiHead(Q, K, V)$ and $head_i$ in the same way as Transformer [6].

$$MultiHead(Q, K, V) = Concat(head_1, \ldots, head_i)W^o \tag{1}$$

In Eq. (1), the sub-header's head is calculated as shown in Eq. (2).

$$head_i = Attention(QW_i^{\mathrm{Q}}, KW_i^k, VW_i^V) \tag{2}$$

In Eqs. (1) and (2), $Concat(*)$ is the concatenation operation for vectors or matrices, $W^o \in R^{d_h \times d_h}$ is the matrix to be trained, $Q, K, V \in R^{n \times d_h}$ are the matrices used as input, d_h is the dimension of input, $W_i^Q, W_i^K, W_i^V \in R^{d_h \times d_k}$ are all the linear projection matrices, and about $d_k = d_h/h$, the h is the number of heads that need to be calculated in parallel.

The output of the entire encoder is K, V, and the upper layer input of the decoder is used as calculating vector Q, which is put into the decoder stack for calculation. The complete model is shown in Fig. 2:

2.1 Key Information Extraction

For the extraction of the key information of the source language text, our approach is to choose an appropriate method to obtain the key information, and integrate the obtained key information into the model structure. There are many ways to obtain key information. This paper uses the method of extracting keywords to obtain key information. The number of keywords is 4.

In many keyword extraction algorithms (TF-IDF, TextRank, YAKE [21], KP-Miner, etc.), through comparison, the experimental results on many data sets have shown that the effect of YAKE is better than other methods, so we use YAKE, which has a better somatosensory effect, as the baseline of this paper.

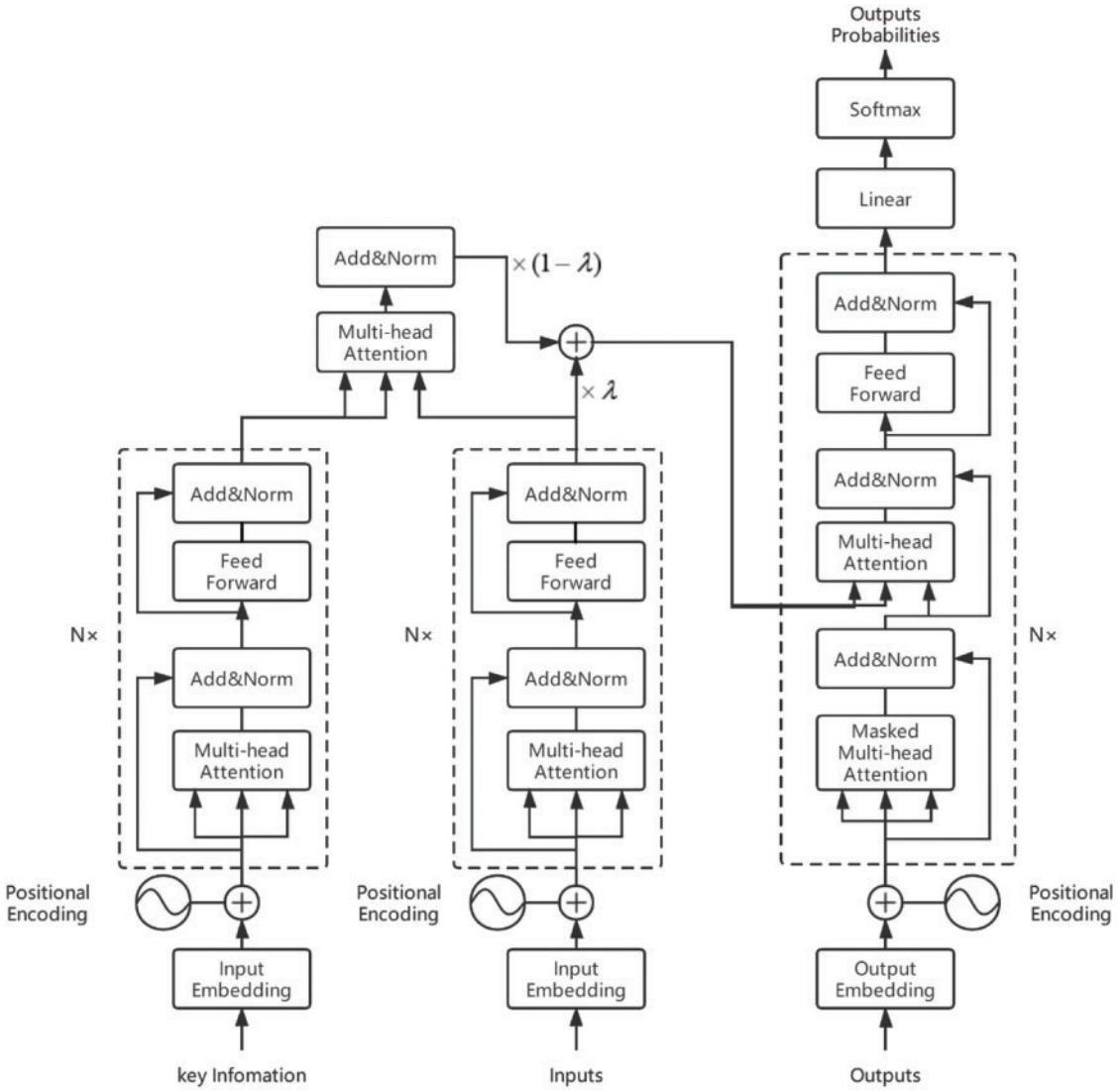

Figure 2: Completed structure of model

2.2 Embedding and Positional Encoding

The neural network model cannot directly process the text sequence, so the text needs to be expressed as a vector in order to be processed by the model. The key information and source language text belong to the high-level expression. The task of Embedding is to map the high-dimensional raw data (sentence) to the low-dimensional data, so that the high-dimensional original data becomes separable after being mapped to the low-dimensional data, and this mapping is called embedding [22,23]. The role of the embedding layer is to convert the input text into a vector form so that it can be processed by the model. Before word embedding, the text sequence needs to be segmented, that is, a paragraph of text is represented as a set of characters or an ordered list of words. Assume $K = [key_1, key_2, \ldots, key_l]$ as the sequence of key information after word segmentation, $S = [s_1, s_2, \ldots, s_m]$ as the sequence of source text after word segmentation, and $T = [t_1, t_2, \ldots, t_n]$ as the sequence of

target text after word segmentation. In these sequences, key_i represents keywords, s_i represents words in the source text, and t_i represents words in the target text. Although the key information we extract is composed of multiple keywords, they may not have a relative position relationship, in order to take the position relationship into consideration, the model ensure their order in the original text. Until now, the model maps the sentence composed of words to a representation vector. Define E_{key}, E_s, E_t as the results of key information, source language text and target language text after embedding.

About positional encoding, use the method that add position encoding into token embedding for key information, source language text and output target language text. On the basis of the built vocabulary, define $I = (i_1, i_2, \ldots, i_n)$ as the results of representation of ID, and input them into an embedding layer (that is, make a linear transformation: $E_{key} = W \cdot I$), after that, add the position information of each token which defined as $\vec{P}_t$ into it, so the results E'_{key} can be calculated, $E'_{key} = \vec{P}_t + E_{key}$. Regarding the calculation of positional encoding, given an input sequence of length n, define t denote the position of the word in the sequence, $\vec{P}_t \in R^d$ as the vector which represents position t, d is the dimension of vector, λ_k represents the frequency, $f : N \rightarrow R^d$ is the function of generated positional vector $\vec{P}_t$, which defined in Eq. (3) as:

$$\vec{p}_t^{(i)} = f(t)^{(i)} := \begin{cases} \sin(\lambda_k \cdot t), & \text{when } i = 2k \\ \cos(\lambda_k \cdot t), & \text{when } i = 2k+1 \end{cases} \tag{3}$$

$$\lambda_k = \frac{1}{10000^{\frac{2i}{d_{\text{model}}}}} \tag{4}$$

With this approach, we get E'_{key}, then put it into the encoder.

2.3 Encoder That Incorporates Key Information

The whole encoder is composed of two sub-encoders. This paper use a single encoder to encode the key information. The two sub-encoders have the same structure. The sub-encoders are stacked with the same $N = 6$ layers, and each encoder layer is composed of two sub-layers, the first layer is a multi-head self-attention model, and the second is a simple fully connected feedforward neural network. Each of the two sub-layers uses a residual connection, and then the normalization operation is performed. The output of every single sublayer is $LayerNorm \cdot (x + Sublayer(x))$, $Sublayer(x)$ is the output of the sublayer itself, the detail is shown in Fig. 3.

As shown in Fig. 3, in the preprocessing stage, the positional encoding data and the embedding data are summed and input into the multi-head self-attention to learn the internal relationship between the source sentence and the keyword. Encoder consists of $N(N = 6)$ identical layers, and residual connection and normalization are used between every two sublayers. Each layer consists of two sub-layers. The first sublayer implements multi-head self-attention, and the second sublayer is a simple position-wise fully connected feedforward network.

In the encoder, define H^i as the output of layer i, and use the output of the previous layer as the input of the next layer. Make the results of word embedding E as the input of the encoder at the first layer. The encoder layer consists of 4 parts: Multi-Head Attention, Add & Norm, Feed-Forward, Add & Norm. The formulas are as follows:

$$C^{i-1} = MultiHead(H^{i-1}, H^{i-1}, H^{i-1}) \tag{5}$$

$$Z^{i-1} = AddNorm(C^{i-1}) \tag{6}$$

$$H^i = AddNorm(FNN(Z^{i-1})) \tag{7}$$

In Eq. (5), *MultiHead*() is multi-head attention, which defined in Eqs. (1) and (2). The *AddNorm*() actually consists of two parts, one is Residual Connection, the other is Normalization, the purpose is avoid vanishing gradient. The *FNN*() in Eq. (7) is the Feed-Forward neural network, this layer is here to increase the nonlinear characteristics of the model.

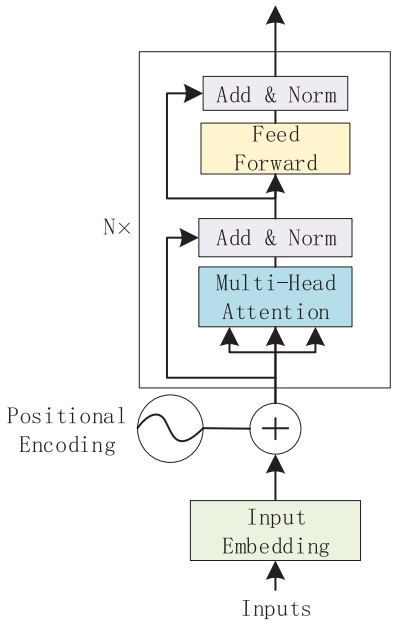

Figure 3: Completed structure of encoder

It should be noted that the positional encoding and Feed-Forward layers mainly provide nonlinear transformations. The second sub-layer is a fully connected layer, and the reason why using positional encoding is the transformation parameters of each position are the same when passing through the linear layer.

Each sub-layer has added Residual Connection and Normalization module, so the output of the sub-layer can be expressed as:

$$sub\_layer\_output = LayerNorm(x + (SubLayer(x))) \tag{8}$$

The above is the calculation process of the key information and source language text sequence encoder. Define H^i_{key} and H^i_S as the output of the key information encoder and the source text encoder, both encoders are at layer i, let $E_{key} = H^0_{key}$, $E_S = H^0_S$.

2.4 Key Information Fusion

In this work, we merge the key information with the encoded result of the source language text, however, the multi-head attention used in the fusion is different from the multi-head attention of the encoder part. The encoder part uses self-attention to calculate the correlation of each word in the sequence, and the fusion calculation part we let the encoding result of the source language text $H^N_S \in R^{m \times d_k}$(in which N is the total numbers of layers, m is the length of source text sequence, d_k is the dimension of word embedding) as the vector Q in calculating vectors, then let the encoding result

of key information $H_{key}^{N} \in R^{l \times d_k}$ as the vector K and vector V in calculating vectors, the output is $A_{key-S} \in R^{m \times d_k}$. Also, to avoid vanishing gradient, as shown in Eq. (9), put the output into the layer of $AddNorm()$.

$$A_{key-S} = AddNorm(MultiHead(H_S^N, H_{key}^N, H_{key}^N) \tag{9}$$

In addition, in order to prevent the fused information from causing excessive interference to the decoder, we use λ to control, the value of λ is $(0 \sim 1)$, the output after fusion is defined in Eq. (10).

$$Y_{key-s} = \lambda H_S^N + (1 - \lambda) A_{key-S} \tag{10}$$

2.5 Decoder

As same as the encoder, the decoder is also composed of $N(N = 6)$ identical layers. Besides the two sub-layers in each encoder layer, the decoder also inserts a third seed layer to implement "multi-headed" attention on the output of the encoder stack. Similar to the encoder, we use residual connections at both ends of each sub-layer to realize short-circuit, and then normalize the layer.

After calculating, let Y_{key-S} as the vector K and vector V in calculating vectors of multi-Head attention in the decoder. The multi-layer attention calculation of the decoder part is shown in Eqs. (11) and (12). In Eq. (11), H_t^i is the decoder's output at layer i. Take the output of the decoder of the previous layer as the input of the next layer, and let the output result of the embedded text sequence of the target language $E_t = H_t^0$.

$$G^{i-1} = AddNorm(MaskedMultiHead(H_t^{i-1}, H_t^{i-1}, H_t^{i-1})) \tag{11}$$

$$Z_t^{i-1} = AddNorm(MultiHead(G^{i-1}, Y_{key-S}, Y_{key-S})) \tag{12}$$

$$H_t^i = AddNorm(FNN(Z_t^{i-1})) \tag{13}$$

The difference between the decoder part and the encoder part is that the decoder introduces the masked multi-head attention in order to prevent the token at the current moment from paying attention to the "future" token. After that, model pass the calculation result of the last layer of the decoder through a linear layer, and then normalize the calculation through the *Soft* max function to obtain the distribution of the output, as shown in Eq. (14).

$$p(y_i | y_{<i}, x) = soft\max(H_t^N W^o) \tag{14}$$

In Eq. (14), $H_t^N \in R^{n \times d_k}$ is the output of the last layer in decoder, n is the length of output sequence, d_k is the dimension of word embedding, $W^o \in R^{d_k \times V}$ is a linear mapping matrix, V is target language vocabulary size.

2.6 Loss Function

The model we designed uses cross entropy [24] as the loss function.

$$L_c = -\frac{1}{N} \sum_{y \in D} \sum_{i=1}^{n} \sum_{j=1}^{m} y_{ij} \log \hat{y}_{ij} \tag{15}$$

In Eq. (15), N is the number of all the training samples, $y \in R^n$ is the real label of input samples, $\hat{y} \in R^n$ is the prediction label of input samples, D is the set of samples, n is the length of generated target text, and m is here to represent target language vocabulary size.

3 Experiment Settings

3.1 Dataset

In recent years, research on neural machine translation prefer to use WMT data set as experimental data set, and use BLEU as evaluation criteria to verify the effectiveness of the method. WMT is one of the top international evaluation competitions in the field of machine translation. Over these years, almost all research institutions will use the WMT dataset as experimental data when publishing papers on new methods of machine translation, and use the BLEU score to measure the effectiveness of the method, giving a quantitative and comparable translation quality assessment, therefore, the WMT dataset has become a recognized mainstream dataset in the field of machine translation. This paper uses the English and Chinese parallel data (News Commentary v12) provided by WMT 2017 as the translated Chinese-English data set. The statistics of this data set are shown in Tab. 1. Among them, the Chinese-English data set includes 227568 Chinese-English text pairs.

Table 1: WMT 17 zh-en dataset

Dataset	Pairs
zh-en	227568

In experiment, set the maximum length of Chinese language text sequence $max\_source\_text\_len = 60$, and set the maximum length of English language text sequence $max\_target\_text\_len = 60$.

This paper uses the Chinese-English data sets in news2017dev and news2017test as the verification set and test set respectively. Both the validation set and the test set contain about 2000 Chinese and English data pairs.

3.2 Evaluation

For the translation results of machine translation, we can use manual methods to evaluate the results, but this method is inefficient and everyone's evaluation criteria are different. Therefore, researchers hope to propose a universal machine translation evaluation standard. And BLEU is the widely used machine translation evaluation standard.

The BLEU method was originally proposed by IBM. This method believes that if the machine-generated translation is more similar to the result of human translation, the higher the translation quality. This method calculates the similarity between the generated translation and the reference translation by counting the $n-gram$ overlap between the two.

First, it counts the number of $n-gram$ that appear in both the generated translation and the reference translation. Then divide the number of co-occurring $n-gram$ by the total number of generated translations. The specific calculation is shown in Eq. (16).

$$p_n = \frac{\sum_{C\in\{Candidates\}} \sum_{n-gram\in C} Count_{clip}(n-gram)}{\sum_{C'\in\{Candidates\}} \sum_{n-gram'\in C'} Count(n-gram)} \tag{16}$$

In Eq. (16), $Count_{clip}(n-gram)$ is different from $Count(n-gram)$, it calculates the maximum number of occurrences of each $n-gram$ in the candidate translation sentence, and $Count(n-gram)$ is the total number of occurrences of all $n-gram$ segments in the reference translation sentence. Finally, the accuracy p_n of each order $n-gram$ is calculated. At the same time, in order to prevent p_n from becoming higher as the sentence length becomes longer, the length penalty factor Brevity Penalty (BP)

is introduced in the calculation of BLEU, and the calculation process is shown in Eq. (17).

$$BP = \begin{cases} 1, & \text{when } c > r \\ e^{1-r/c}, & \text{when } c \leq r \end{cases} \tag{17}$$

4 Results and Discussion

To compare the performance of the model proposed in this paper with other models, this paper also uses Transformer to train on the same experimental data. During the experiment, we use the English and Chinese parallel data (News Commentary v12) provided by WMT 2017 as the translated Chinese-English data set, and the number of keywords used is four. To make a convincing conclusion, the test set is divided to different parts, then compared the BLEU scores of different sentence lengths.

In this work, we mainly compare and analyze the BLEU evaluation scores of the designed neural machine translation model and the benchmark system, and the translation examples generated by our machine translation model on the test set. The BLEU scores of the model proposed in this article and the benchmark model are shown in Tab. 2.

Table 2: BLEU scores of models

SRC TEXT LEN	(0, 10]	(10, 20]	(20, 30]	(30, 40]	(40, 50]	(50, +)
BLEU of KI-Fusion model	**11.40**	**22.06**	**25.73**	**18.92**	**6.56**	**2.33**
BLEU of transformer	11.33	19.51	22.80	15.18	3.96	1.97

As shown in Tab. 2, we list the BLEU scores of our designed model and benchmark system on the testing set. According to the experimental data, the length of the source sentence is in the interval (0, 10), (10, 20), (20, 30), (30, 40), (40, 50) and (50, +). The BLEU scores of the translation results are shown in Tab. 1. The following conclusions can be drawn from the data in the Tab. 1:

1) The proposed method has the most obvious improvement in the translation effect of source text which has long sentences. The BLEU score of the method in the interval (0, 10) is increased by about 0.07, and the BLEU score in the interval (10, 20] is increased by about 2.55. In addition, the BLEU score of the method in the interval (20, 30) and (30, 40) The score increased by about 2.93 and 3.47 respectively.
2) The overall performance of the model proposed in this paper is better than Transformer base, but when the source sentence length exceeds 50, the BLEU scores of the two methods are both very low. On one hand, it is because the number of text pairs in this interval in the test set is small, on the other hand, with the sentence length increasing, the translation becomes more difficult.
3) Compared with Transformer, the model proposed in this paper has improved BLEU values on translations of different lengths at the source end.

As shown in Fig. 4, the neural machine translation model we have achieved that integrates key information has been significantly improved on the basis of Transformer. Therefore, by comparing with the BLEU evaluation score of Transformer, it can be concluded that the method we designed based on the Transformer model to fuse key information can indeed improve the quality of machine

translation. By comparing with the benchmark model of Transformer model, we can see that our model has achieved better results.

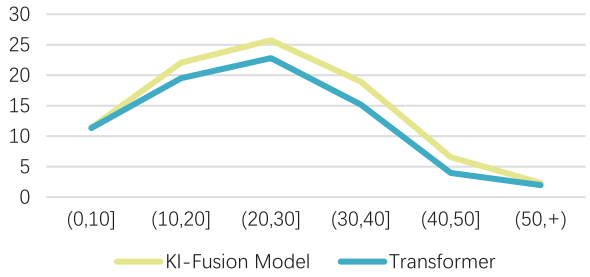

Figure 4: BLEU scores of models

Tab. 3 takes a translation result on the test set as an example, and compares the translation results of a long sentence between the benchmark model Transformer and the model proposed in this article.

Table 3: Translation samples

Source text	This week, China's General Administration of Customs released statistics show that the first 7 months of this year, China's import and export value of 13.21 trillion yuan, down 3% over the same period last year.
Reference	本周，中国海关总署公布的统计数据显示，今年前 7 个月，中国进出口总值 13.21 万亿元人民币，比去年同期下降 3%。
Translation of Transformer	BLEU
本周，中国机构政府的数字表明去年的第七个月，中国进口和投资价值为 13.21 万亿元人民币，比去年同期下降了 3%.	0.293
Translation of KI-Fusion Model	BLEU
本周，中国关税公布的数字显示，今年最初 7 个月，中国的进口和出口价值 13.12 万亿元人民币，比去年同期下降了 3%。	0.359

As shown in Tab. 3, the BLEU score is also used as the standard, and it can be seen that the BLEU score of the model proposed in this paper is significantly higher than the score of Transformer, while the translation quality is higher, too. This shows that the neural machine translation model fused with key information can obtain higher translation quality by comprehensively using key information to adjust the input of the encoder, thereby improving the performance of the model.

Based on the above results and analysis of the results, the neural machine translation model proposed in this paper has achieved a satisfactory result on the Chinese and English data sets, and compared with the benchmark model, the model proposed in this paper has significant improvement, which shows the effectiveness of the method proposed in this paper.

5 Conclusion

The model proposed in this paper combines the Transformer proposed by Google Brain, via fusing key information, this neural machine translation model not only improves the stability of the base model, but also makes the model more accurate in translation tasks.

The entire encoder part of the model fuses the key information with the encoded result of the source language text, and the complete information after the fusion is used as the K and V in the multi-head attention in the decoder to participate in the multi-layer decoding calculation. In this way,

the coding of key information is added to the model training, which improves the accuracy of the translation model in the translation task, to a certain extent, it reduces the mistranslation and omission of the neural machine translation model when translating sentences.

In the future, we hope to add other structures, such as recurrent neural networks, to the convergence between the two stages of key information encoding and source language text encoding to continue processing the low-dimensional features and high-dimensional features in the key information and source language text to further improve the performance of neural machine translation models that integrate key information on public data sets.

Acknowledgement: Special thanks to School of Information and Software Engineering and the members of the Information of Physics Computation Center for their help and support for this research. We would also like to thank everyone from the Institute of Logistics Science and Technology for their technical and experimental support.

Funding Statement: This work was supported by Major Science and Technology Project of Sichuan Province [No. 2022YFG0315, 2022YFG0174]; Sichuan Gas Turbine Research Institute stability support project of China Aero Engine Group Co., Ltd. [No. GJCZ-2019-71].

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] K. Nal and P. Blunsom, "Recurrent continuous translation models," in *Proc. of the 2013 Conf. on Empirical Methods in Natural Language Processing*, Seattle, Washington, USA, pp. 1700–1709, 2013.

[2] L. Sutskever, O. Vinyals and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. of the 27th Int. Conf. on Neural Information Processing Systems*, pp. 3104–3112. MA, USA, MIT Press, 2014.

[3] J. Lee, K. Cho and T. Hofmann, "Fully character-level neural machine translation without explicit segmentation," *Transactions of the Association for Computational Linguistic*, vol. 5, no. 1, pp. 365–378, 2017.

[4] P. Koehn, F. J. Och and D. Marcu, "Statistical phrase-based translation," in *2003 Conf. of the North American Chapter of the Association for Computational Linguistics on Human Langauge Technology*, Edmonton, Canada, pp. 48–54, 2003.

[5] H. Ji, S. Oh, J. Kim, S. Choi and E. Park, "Integrating deep learning and machine translation for understanding unrefined languages," *Computers, Materials & Continua*, vol. 70, no. 1, pp. 669–678, 2022.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones *et al.,* "Attention is all you need," in *Proc. of the 31st Int. Conf. on Neural Information Processing Systems*, pp. 6000–6010. NY, United States, Curran Associates Inc., 2017.

[7] S. K. Mahata, D. Das and S. Bandyopadhyay, "Mtil2017: Machine translation using recurrent neural network on statistical machine translation," *Journal of Intelligent Systems*, vol. 28, no. 3, pp. 447–453, 2014.

[8] S. Sukhbaatar, D. Ju, S. Poff, S. Roller, A. Szlam *et al.,* "Not all memories are created equal: Learning to forget by expiring," in *Int. Conf. on Machine Learning*, Long Beach, California, USA, pp. 9902–9912, 2019.

[9] W. Guo, J. Fan and K. Zhang, "Advance research on neural machine translation integrating linguistic knowledge," *Journal of Frontiers of Computer Science and Technology*, vol. 15, no. 7, pp. 1183–1194, 2021.

[10] X. Kang and C. Zong, "Fusion of discourse structural position encoding for neural machine translation," *Chinese Journal of Intelligent Science and Technology*, vol. 2, no. 2, pp. 144, 2020.

[11] D. Zheng, Z. Ran, Z. Liu, L. Li and L. Tian, "An efficient bar code image recognition algorithm for sorting system," *CMC-Computers, Materials & Continua*, vol. 64, no. 3, pp. 1885–1895, 2020.

[12] M. Wang, B. Niu and L. Ma, "Transformer model improvement method by word-level weights," *Journal of Chinese Computer Systems*, vol. 40, no. 4, pp. 744–748, 2019.
[13] X. Li, J. Ma and S. Qin, "Image attention fusion for multimodal machine translation," *Journal of Chinese Information Progressing*, vol. 34, no. 7, pp. 68–78, 2020.
[14] A. Gillioz, J. Casas, E. Mugellini and O. A. Khaled, "Overview of the transformer-based models for NLP tasks," in *2020 15th Conf. on Computer Science and Information Systems (FedCSIS)*, Sofia, Bulgaria, pp. 179–183, 2020.
[15] Y. Peng, X. Li, J. Song, Y. Luo, S. Hu *et al.,* "Verification mechanism to obtain an elaborate answer span in machine reading comprehension," *Neurocomputing*, vol. 466, no. 1, pp. 80–91, 2021.
[16] W. Luo, "Analyzing the problems of vocabulary in Japanese-Chinese neural network machine translation," *Computer Science and Application*, vol. 10, no. 3, pp. 387–397, 2020.
[17] J. Zhang, J. Liu and X. Lin, "Improve neural machine translation by building word vector with part of speech," *Journal on Artificial Intelligence*, vol. 2, no. 2, pp. 79–88, 2020.
[18] S. Hu, X. Li, Y. Deng, Y. Peng, B. Lin *et al.,* "A semantic supervision method for abstractive summarization," *Computers, Materials & Continua*, vol. 69, no. 1, pp. 145–158, 2021.
[19] T. Alkhouli, G. Bretschner, J. T. Peter, M. Hethnawi, A. Guta *et al.,* "Alignment-based neural machine translation," in *Proc. of the First Conf. on Machine Translation: Volume 1, Research Papers*, Berlin, Germany, pp. 54–65, 2016.
[20] K. Papineni, S. Roukos, T. Ward and W. J. Zhu, "Bleu: A method for automatic evaluation of machine translation," in *Proc. of the 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, PA, USA, pp. 311–318, 2002.
[21] R. Campos, V. Mangaravite, A. Pasquali, A. Jorge, C. Nunes *et al.,* "YAKE! keyword extraction from single documents using multiple local features," *Information Sciences Journal*, vol. 509, no. 1, pp. 257–289, 2020.
[22] S. Ghannay, B. Favre, Y. Estève and N. Camelin, "Word embedding evaluation and combination," in *Proc. of the Tenth Int. Conf. on Language Resources and Evaluation (LREC'16)*, Portorož, Slovenia, pp. 300–305, 2016.
[23] O. Levy and Y. Goldberg, "Neural word embedding as implicit matrix factorization," in *Proc. of the 27th Int. Conf. on Neural Information Processing Systems*, pp. 2177–2185. MA, USA, MIT Press, 2014.
[24] P. T. De Boer, D. P. Kroese, S. Mannor and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Annals of Operations Research*, vol. 134, no. 1, pp. 19–67, 2005.