International Journal for Housing Science and Its Applications
Publish August 10, 2025. Volume 46, Issue 4 Pages 4447-4461

https://doi.org/10.70517/ijhsa464290

Optimizing the quality of long English text translation: Paradigm improvement driven by self-attention mechanisms

Xiaoyan Li[1,2], Wei Chen[2,*], Ruonan Wang[3] and Jingjing Zhang[1,2]
[1] School of Education, Hefei University, Hefei, Anhui, 230616, China
[2] School of Foreign Languages, Bengbu University, Bengbu, Anhui, 233030, China
[3] Xidian University, Xi'an, Shaanxi, 710068, China

Corresponding authors: (e-mail: *davidchenlxy7579@163.com).

Abstract Machine translation technology plays an important role in the process of globalization, but traditional translation systems often face semantic breaks and lack of coherence when dealing with long texts. Although existing neural machine translation models perform well at the sentence level, they are still deficient in cross-sentence semantic understanding and contextualization. In this study, an optimization model based on the multi-head self-attention mechanism is constructed to address the problem of lack of semantic coherence in English long text translation. Methodologically, a context-dependent semantic coherence computation model is designed by adopting an encoder-decoder architecture combined with the multi-head attention mechanism, extracting sentence features through convolutional neural networks, and fusing document topic information and semantic matching strategies. The replication mechanism and gating mechanism are introduced into the encoder to improve the accuracy of vocabulary generation. The results show that after integrating the multi-head attention mechanism, the model achieves a BLEU value of 22.0885 on the Chinese-English translation task, which is improved by 0.7885 compared with the baseline model; in the semantic coherence analysis task, the accuracy rate reaches 60.2485%, with an F1 value of 49.4955%; and the Pearson's correlation coefficient with the manual scoring is 0.7498.The conclusions show that the multi-head self-attention mechanism can effectively capture global semantic relations in long texts, significantly improve translation quality and semantic coherence, and provide a feasible technical path for English long text translation.

Index Terms multi-head self-attention mechanism, English long text translation, semantic coherence, encoder-decoder, convolutional neural network, machine translation

I. Introduction

In modern society, English translation plays an increasingly important role, and with the increase of economic and cultural ties worldwide, the demand for English translation is growing [1], [2]. English translation not only needs to accurately convey meanings and details, but also needs semantic coherence, especially in the translation of long texts, which is the key to ensure that the translated text is fully understood [3]-[5]. Semantic coherence refers to the connections and consistency between individual sentences or paragraphs in a text [6]. These connections can be realized through logical articulation, grammatical consistency and lexical repetition [7]. Semantic coherence in the translation of long English texts is one of the key factors in ensuring that readers are able to understand the translated text; if the text is incoherent, readers will be confused and may misunderstand it [8]-[10].

Semantic coherence can be achieved in several ways: (1) Logical articulation: is the expression of connections between different contents or ideas in a text [11]. Logical articulation can be realized through the use of logical connectives or phrases, which can help translators organically combine different parts of a text to form a coherent whole [12]-[14]. (2) Text consistency: it is the consistency between individual sentences or paragraphs in a text [15]. Textual consistency can be achieved by maintaining consistency in subject, tense, person and tone, which can help readers understand the text better and reduce the possibility of misunderstanding during reading [16], [17]. (3) Lexical repetition: lexical repetition refers to the repeated use of the same words in a text for coherence purposes [18]. This can help the translator to combine the parts of the text and reduce the possibility of misunderstanding by the reader during the reading process [19], [20]. And with the development of artificial intelligence, the optimization of semantic coherence of English long text translation can be achieved based on the model of multi-head self-attention mechanism, which can solve the problem of information dilution that exists in the traditional machine translation, and achieve the semantic coherence of English long text translation [21]-[24].

This study proposes to construct a semantic coherence optimization model for English long text translation based on multi-head self-attention mechanism. First, the encoder-decoder architecture is designed to incorporate the multi-

head self-attention mechanism in the encoder, which enhances the model's ability to capture long-distance dependencies. Second, a context-dependent semantic matching model is constructed to extract sentence features by convolutional neural network and combine them with document topic information for semantic fusion. Then, the semantic coherence degree calculation method is designed to analyze text coherence using frequent subgraph patterns. Finally, the effectiveness of the model is verified by multiple datasets, and the impact of different components on translation quality is analyzed.

II. Construction of English Long Text Translation Model Based on Multiple Attention Mechanisms

II. A. Encoder-Decoder Model for Synthesizing Chinese-English Code Conversion Texts

II. A. 1) CS text synthesis based on the encoder-decoder model

In this section, a recurrent neural network-based encoder-decoder model is used to construct a generator for generating bilingual CS text data. This generator implicitly learns the linguistic constraint rules of CS from a limited number of CS texts, implicitly learns the linguistic constraint rules within a language from a large number of monolingual parallel corpora, and then utilizes the monolingual parallel corpora to generate bilingual CS text data.

Figure 1 shows the CS text generator based on the encoder-decoder model, which consists of an encoder, a decoder, and an attention mechanism. In this paper, we use a bi-directional long short-term memory network (BLSTM) as an encoder, a unidirectional long short-term memory network (LSTM) as a decoder, and a content vs. location based approach as an attention mechanism.

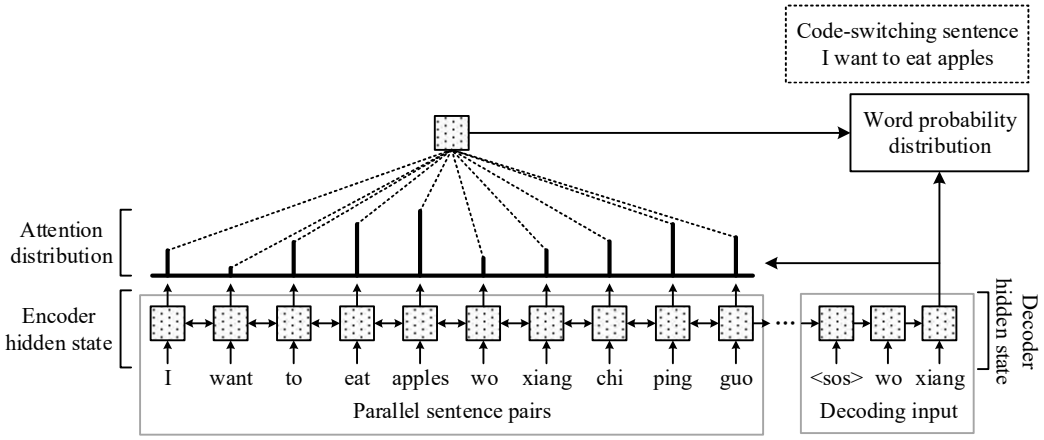

Figure 1: CS text generator based on the encoder-decoder model

The encoder inputs a sequence of words $X=[x_1,\cdots,x_L]$, L is the length of the input word sequence, and the word sequences include five kinds of Chinese-English parallel sentence pairs, English-Chinese parallel sentence pairs, Chinese sentences, English sentences, and Chinese-English CS sentences. The encoder encodes the word sequences into a sequence of encoded vectors $H=[h_1,\cdots,h_L]$, as shown in equation (1):

$$H=BLSTM(X) \tag{1}$$

The attention mechanism receives the implicit state s_{t-1} of the decoder at each output time step t and computes the attention weight vector $a_t=[a_{t,1},\cdots,a_{t,L}]$ and acts on the sequence of coding vectors to produce the context vector c_t for the t th output time step as shown in Equation (2):

$$c_t=\sum_{i=1}^{L}a_{t,i}h_i \tag{2}$$

The attention weights learned by the model denote intra-linguistic linguistic constraint rules, and cross-linguistic linguistic constraint rules.

The decoder receives the context vector c_t with the output word of the previous output time step $t-1$ and combines it with the decoder's implicit state s_{t-1} to obtain the decoder's current implicit state s_t, which is then

predicted by the output-layer mapping to predict the probability distribution of the word of the current label $P_{voc}(w_t)=\left[p_{voc}(w_t^1),\cdots,p_{voc}(w_t^v)\right]$, V is the size of the output vocabulary list as shown in equation (3):

$$p_{voc}(w_t^v)=LSTM(w_{t-1}^*,[s_{t-1},c_t]) \tag{3}$$

In the training phase, w_{t-1}^* is the reference label, and in the synthesis phase, w_{t-1}^* is the output of the decoder from the previous time step.

The training objective function of this generator is the cross-entropy between the reference sequence and the predicted sequence, as shown in Equation (4):

$$Loss=-\frac{1}{T}\sum_{t=1}^{T}p_{voc}(w_t^*) \tag{4}$$

The types of reference sequences inputted by the decoder include five kinds: Chinese-English parallel sentence pairs, English-Chinese parallel sentence pairs, Chinese sentences, English sentences, and Chinese-English CS sentences.

II. A. 2) CS synthesis based on band replication mechanism

Text generators based on the encoder-decoder model result in synthesized word sequences that are subject to fewer intra-linguistic linguistic constraints versus cross-language linguistic constraints, i.e., the synthesized text has a low degree of naturalness due to the fact that the decoder does not show to receive timely linguistic knowledge guidance during the decoding process. To solve this problem, based on this, this subsection introduces a replication mechanism for the encoder-decoder. On top of the CS text generator based on the encoder-decoder model, a gating is added which determines whether the next word produced by the generator is predicted from the decoder or copied from the input source text of the encoder. The gating probability $p_{gen}\in[0,1]$ denotes the probability that the current word selects the word predicted by the decoder (from the predicted vocabulary list distribution), while $1-p_{grn}$ denotes the probability that the current word selects the copied text word.

The p_{gen} is jointly computed from the encoder's context vector c_t, the decoder's implicit state s_t, and the decoder's current input, i.e., the previous output w_{t-1}^*:

$$p_{gen}=\sigma(W_c^T c_t+W_s^T s_t+W_w^T Emb(w_{t-1}^*)) \tag{5}$$

where W_c,W_s,W_w are trainable parameter matrices, and $Emb(w_{t-1}^*)$ is the embedding vector of the word w_{t-1}^*.

The final glossary output probability distribution $P(w_i)\neq[p(w_i^1),\cdots,p(w_t^v)]$ produced by the decoder is the glossary probability distribution $P_{vc}(w_t)$ with all the inputs of the encoder as $x_i=w_i^v$. The corresponding attentional weight summation is obtained after gated summation:

$$p(w_t^v)=p_{gen}p_{vec}(w_t^v)+(1-p_{gen})\sum_{i:x_i=n_i}a_{t,i} \tag{6}$$

The training objective function of this generator is the cross-entropy of the reference sequence and the predicted sequence, as shown in Equation (7):

$$Loss=-\frac{1}{T}\sum_{t=1}^{T}p(w_t^*) \tag{7}$$

II. B. Machine Translation Model Based on Encoder-decoder Architecture

II. B. 1) Encoders based on multiple attention mechanisms

Text generators based on the encoder-decoder model result in synthesized word sequences that are subject to fewer intra-linguistic linguistic constraints versus cross-language linguistic constraints, i.e., the synthesized text has a low degree of naturalness due to the fact that the decoder does not show to receive timely linguistic knowledge guidance during the decoding process. To solve this problem, based on this, this subsection introduces a replication mechanism for the encoder-decoder. On top of the CS text generator based on the encoder-decoder model, a gating is added which determines whether the next word produced by the generator is predicted from the decoder or copied from the input source text of the encoder. The gating probability $p_{gen}\in[0,1]$ denotes the probability that the current word selects the word predicted by the decoder (from the predicted vocabulary list distribution), while $1-p_{grn}$ denotes the probability that the current word selects the copied text word.

The p_{gen} is jointly computed from the encoder's context vector c_t, the decoder's implicit state s_t, and the decoder's current input, i.e., the previous output w_{t-1}^*:

$$p_{gen} = \sigma(W_c^T c_t + W_s^T s_t + W_w^T Emb(w_{t-1}^*)) \tag{8}$$

where W_c, W_s, W_w are trainable parameter matrices, and $Emb(w_{t-1}^*)$ is the embedding vector of the word w_{t-1}^*.

The final glossary output probability distribution $P(w_i) \neq [p(w_i^1), \cdots, p(w_t^v)]$ produced by the decoder is the glossary probability distribution $P_{vc}(w_t)$ with all the inputs of the encoder as $x_i = w_i^v$. The corresponding attentional weight summation is obtained after gated summation:

$$p(w_t^v) = p_{gen} p_{vec}(w_t^v) + (1 - p_{gen}) \sum_{i:x_i = n_i^{'}} a_{t,i} \tag{9}$$

The training objective function of this generator is the cross-entropy between the reference sequence and the predicted sequence, as shown in Equation (10):

$$Loss = -\frac{1}{T} \sum_{t=1}^{T} p(w_t^*) \tag{10}$$

In this section, the encoder based on multi-head attention mechanism is used to capture the key parts of text sequence features and spatial features.

The output results after processing by the embedding layer are used as input data for the encoder. In this section, the word sound and word shape embedding representations in the previous section are vectorially connected with the word element embedding results, and the obtained results are fed into the fully connected layer and summed with the positional embedding and segmentation embedding, and finally the embedded results are obtained through the regularization layer and dropout layer. Among them, the role of the regular layer is to add a regular term in the loss function, which is used to penalize the weight values that are too large or too small, and prevent overfitting or underfitting from occurring. The dropout layer is used to randomly discard a part of the neurons during the training process, in order to reduce the dependence between the neurons, and to enhance the network's ability of generalization, as well as avoiding the occurrence of overfitting. The specific process is shown in Eqs. (11)-(13):

$$E_{concat} = FC(E_{pinyin} \oplus E_{font} \oplus E_{token}) \tag{11}$$

$$E_{total} = E_{concat} + E_{pos} + E_{seg} \tag{12}$$

$$H_v = Dropout(Norm(E_{total})) \tag{13}$$

$\oplus$ denotes the vector concatenation operation (concat), $E_{concat} \in \mathbb{R}^{l \times d_e}$ denotes the result of concatenation of the word sound, word shape and word embedding layers, $E_{total} \in \mathbb{R}^{l \times d_e}$ denotes the final result after summing E_{concat} with positional, segmentation embedding, and the output after regular and dropout layers is $H_v \in \mathbb{R}^{l \times d_h}$, d_h denotes the hidden layer dimension, $d_h = d_e$.

Due to the complexity and diversity of Chinese variant characters, it brings some challenges to mine the core semantic information, while the attention mechanism can accurately capture the key parts of the vectors. In this paper, we adopt the multi-head attention mechanism in the model to expand the model's ability to focus on different positions, and at the same time give the model multiple representation subspaces. The Query, Key, and Value weights corresponding to the i th head are $W_i^Q \in \mathbb{R}^{d_h \times d_q}, W_i^K \in \mathbb{R}^{d_h \times d_k}, W_i^V \in \mathbb{R}^{d_h \times d_v}$, the Query matrix $Q \in \mathbb{R}^{l \times d_h}$ is calculated and the Key matrix $K \in \mathbb{R}^{l \times d_h}$, Value matrix $V \in \mathbb{R}^{l \times d_h}$, and then based on all the $head_i$ and output weights $W^o \in \mathbb{R}^{d \times d_h}_v$ to get MutiHead, in this paper, we use the number of attention heads as $h = 8, d_q = d_k = d_v = d_h / h$. Finally, the output result of the encoder is obtained through the residual network, regularization layer and feedforward network, in which the residual layer is used to solve the degradation problem of deep neural networks and improve the training efficiency by learning the residual function between the input and the output, as shown in Eqs. (14)-(16):

$$Q_i = H_v W_i^Q, K_i = H_v W_i^K, V_i = H_v W_i^V \tag{14}$$

$$head_i(Q_i, K_i, V_i) = softmax(\frac{(Q_i K_i)^T}{\sqrt{d_k}}) V_i \tag{15}$$

$$MutiHead(Q, K, V) = (head_1 \oplus head_2 \oplus ... \oplus head_n) W^o \tag{16}$$

II. B. 2) English Long Text Encoder

A text decoder is a network structure used in conjunction with an encoder to transform data for the current word based on the encoder's feature representation of the current word and the prediction result of the previous word. The text decoder used in this section consists of an input layer, an attention mechanism and a generator.

The input of the text decoder consists of two parts: the decoder output of the previous time step and the last encoder output of the current time step, which is obtained through the residual linkage, regularization layer, feedforward layer, and the multi-head attention mechanism as shown in Eqs. (17)-(18):

$$H_t^{enc} = FW(Add\ \&\ Norm(MultiHead_{enc}^{t}(Q,K,V))) \tag{17}$$

$$H_{t-1}^{dec} = FW(Add\ \&\ Norm(MultiHead_{dec}^{t-1}(Q,K,V))) \tag{18}$$

where H_{t-1}^{dec} is the output of the decoder at time step $t-1$ and H_t^{enc} denotes the output of the encoder at time step t.

Unlike the encoder, in the decoder's attention mechanism, the input to the Query matrix Q_t^{dec} at time step t comes from the output of the decoder at the previous time step $t-1$, H_{t-1}^{dec}. The inputs of the Key matrix K_t^{dec} and the Value matrix V_t^{dec} The output H_t^{enc} from the current time step t of the encoder is shown in Equation (19). Through the attention mechanism, we can focus on the connection between the predicted normal characters and the corresponding variant characters, mine deeper semantic information, and improve the model's learning ability on the Chinese variant character task:

$$Q_t^{dec} = H_{t-1}^{dec}W_t^{Q}, K_t^{dec} = H_t^{enc}W_t^{K}, V_t^{dec} = H_t^{enc}W_t^{V} \tag{19}$$

The generator in this paper consists of a linear layer and a LogSoftmax layer to solve the problem of overflow and underflow of Softmax results and improve the efficiency of model training. The generator calculates the lexicon probability distribution through the LogSoftmax layer and the linear layer based on the attention score to get the corresponding predicted words, as shown in Equation (20):

$$P_{vocab} = Logsoftmax(W_1\alpha_t + b_1) \tag{20}$$

where P_{vocab} is the lexicographic probability distribution of the task, α_t denotes the attention score at time step t, and W_1 and b_1 are learnable parameters.

II. C. English Long Text Translation Based on Multiple Attention Mechanisms

Transformer is a deep learning model based on an attention mechanism, and like RNNs, Transformer is designed to process data whose inputs are sequences, such as in natural language processing for tasks such as translation and text summarization. However, Transformer does not process data in the order of the sequence itself. For example, if the input data is a natural language sentence, instead of processing it from the beginning of the sentence to the end of the sentence, the Transformer recognizes the meaning of each word in the sentence in context, and then builds the corresponding attention relation for each word.This working mechanism of the Transformer model determines that it can perform more parallelized computations, which achieves the effect of decreasing training time. Prior to Transformer, most of the best performing NLP models relied on RNN structures, such as LSTMs and Gated Recurrent Units (GRUs), with added attention mechanisms, while the Transformers model, which is based on an attention mechanism, has topped several NLP tasks in recent years, which proves the conclusion that --The model based on the attention mechanism itself has the performance of RNN with the efficiency of attention.

Figure 2 shows the structure of the Transformer model, which consists of two parts, the encoder group and the decoder group, both of which are stacked by 6 identical encoders and decoders. The design of this 6-layer stacked structure was developed by Google's research team after a lot of experiments, and does not have a strict logical meaning in itself.

Each encoder in Transformer consists of two main modules: a self-attention mechanism and a feed-forward neural network. The self-attention mechanism inputs the coded vectors from the previous layer of encoder outputs and computes the correlation between them to generate the output coded vectors. The feed-forward neural network further processes each output coding vector and then passes these output coding vectors as inputs to the next encoder as well as to the decoder. The first layer encoder takes the Embedding and positional information of the input sequence as its input.

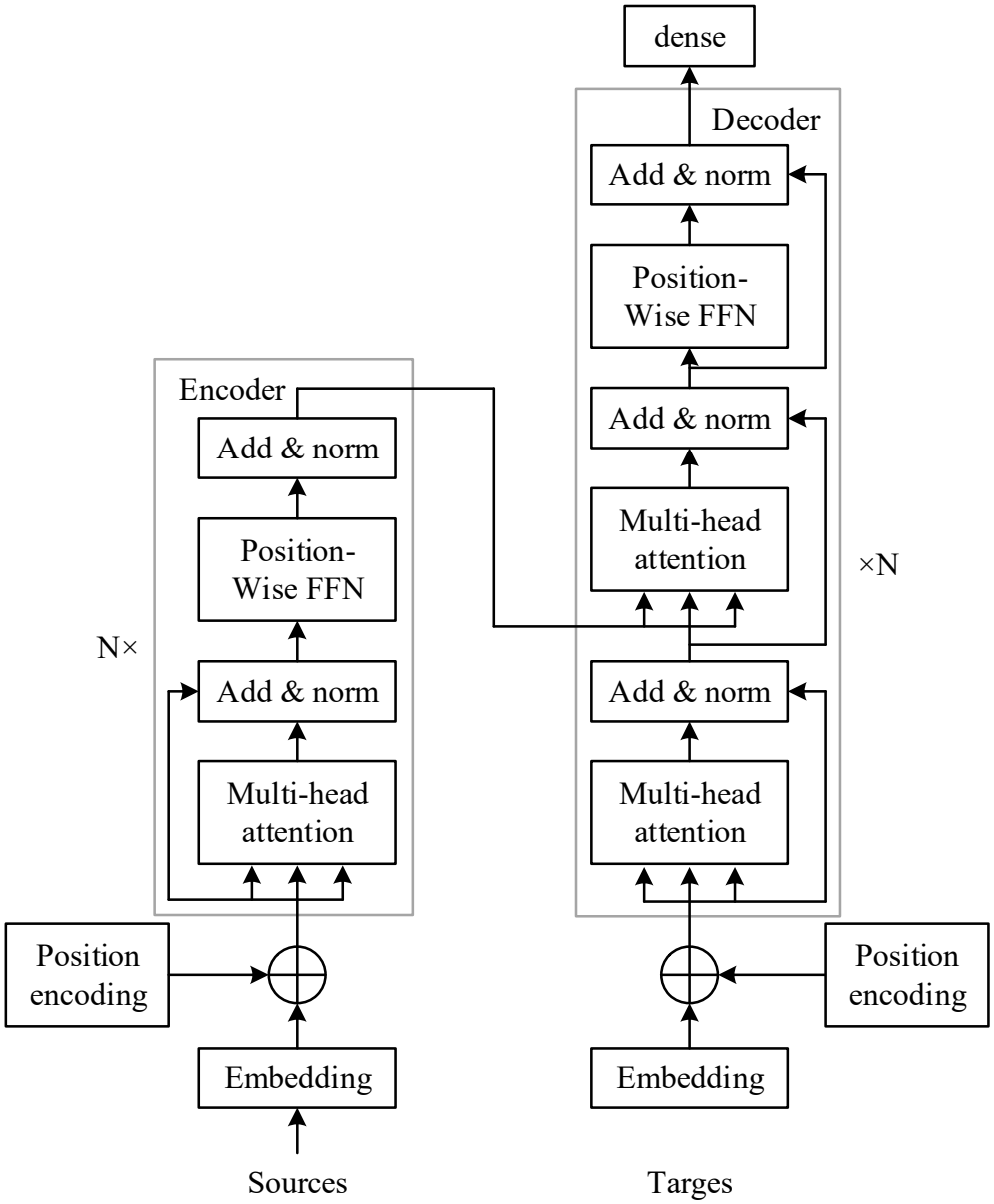

Figure 2: Structure of the Transformer model

Each decoder of Transformer consists of three main modules: a self-attention mechanism, an encoding attention mechanism, and a feed-forward neural network. The decoder functions similarly to the encoder, with the difference that an encoding attention mechanism is added; the added attention mechanism is to correlate the correspondence between the encoded vectors and the decoder output vectors. The first layer decoder takes as its input the Embedding and location information of the output sequence, which is the same as the input of the first layer encoder. Considering that the decoder can only predict future outputs based on the current and previous outputs, the decoder needs to mask part of the output sequences during the training process to prevent the model from "spying" on the reference answer in advance. The last decoder is connected to the softmax layer, which is used to generate the predicted probability of the output sequence.

In the Transformer model, not only the sequences are Embedding, but also the positional information of the sequences, i.e., the positional encoding (PE) of the sequences, which is used to represent the position of the elements in the sequences. This is due to the fact that the Transformer model does not process the sequences in the order of the sequences themselves as RNN does, and although it pays attention to the connections between the elements by virtue of the attention mechanism, it does not pay attention to the positional information of the elements in the sequences, which is quite necessary in natural language processing. Therefore the Transformer

model uses the position encoding method of sine and cosine functions in order to preserve the position information of the elements in the sequence, which is calculated as follows:

$$PE_{(pos,2i)} = sin(pos / 10000^{2i/d}) \tag{21}$$

$$PE_{(pos,2i+1)} = cos(pos / 10000^{2i/d}) \tag{22}$$

where pos denotes the position of the element in the sequence, d denotes the dimension of the positional encoding (same as the dimension of the sequence Embedding), PE denotes the positional encoding of the sequence, $2t$ denotes the even dimension of the sequence, and $2i+1$ denotes the odd dimension of the sequence.

III. Optimizing Semantic Coherence in English Long Text Translation

III. A. Context-dependent Semantic Matching Models

III. A. 1) Convolutional neural network based sentence feature extraction

In this paper, we use convolutional neural networks to learn semantic representations of sentences and candidate target phrases. For the input sentence, the words in the sentence are first initialized into pre-trained word vectors to obtain the word vector matrix of all the words in the sentence. Second, the convolutional neural network performs convolutional operations on all possible windows of the input sentence and pools the features obtained from the convolution for selection. After several convolution and pooling operations, the semantic feature representation of the sentence is finally obtained. Assuming that the length of the input sentence is n and $x_i \in R^k$ is the word vector of the i th word in the sentence (with dimension k), the source language sentence can be viewed as a matrix of $n \cdot k$ dimensions. For ease of presentation, the sentence is represented in the form shown in Eq. (23):

$$x_{1:n} = x_1 \oplus x_2 \oplus ... \oplus x_n \tag{23}$$

where $\oplus$ denotes the concatenation operator and $x_{i:l+j}$ denotes connecting the word vectors of the words $x_i, x_{i+1}, ...x_{i+j}$ of word vectors are connected.

In the first layer (Layer-1), the convolutional layer of the neural network takes as input a matrix of word vectors for a source language sentence f or a candidate phrase $\tilde{e}$, and operates a sliding window of length h to perform a convolutional transformation of all possible combinations of neighboring word sequences using a filter $w \in R^{h \cdot k}$. For each possible combination of word sequences, a new feature representation is generated, which is finally combined into a feature vector of the sentence or phrase. This is shown in equation (24):

$$c_i^{(1,j)} = f(w^{(1,j)} \cdot x_{i:i+h-1} + b^{(1,j)}) \tag{24}$$

where $c_i^{(1,j)}$ denotes the feature generated by the word window $x_{i:i+h-1}$ after convolution of the j th filter, $w^{(1,l)}$ denotes the parameter matrix corresponding to the j th filter in the first layer, $b^{(1,J)} \in R$ denotes the bias term, and f denotes the nonlinear activation function, and in this paper we use the ReLu function as the activation function.

In order to distinguish phrases and their contexts, this paper adds a new dimension after the word vector to distinguish whether the current word is within the scope of a phrase pair, where 1 means the word is inside the phrase pair and 0 means other context words. The purpose of adding this dimension is to allow the model to take more information about the current phrase into account during the training process. Since the lengths of sentences and target candidate phrases are variable, in this paper, we fill in the beginning of sentences or target candidate phrases to reach the maximum length of sentences in the training set, so as to ensure that the input sequences are of the same length.

In the second layer (Layer-2), its input is the feature vector obtained from the convolution of the previous layer. In this paper, all adjacent non-overlapping convolutional features are pooled and selected. The details are shown in Eq. (25):

$$c_i^{(2,j)} = \max\{c_{2i}^{(1,j)}, c_{2i+1}^{(1,j)}\} \tag{25}$$

After several convolution and pooling operations, this paper obtains the feature vectors of source language sentences and target phrases.

III. A. 2) Sentence Semantics and Text Theme Fusion Strategies

In this paper, we use the Wikipedia document set to train the LDA topic model and use the generated model to reason about the topic vectors of all documents in the translation system training set. The topic vectors of the

documents represent the probability that a document belongs to each topic. This topic information will be used to assist translation decisions.

In addition, the topic information of a document can be represented by the words in the document, so this paper also obtains the vector representation of a document by weighting the word vectors of different words in the document. In this paper, we use the TF-IDF metric to measure the importance of words in a document, where the IDF value indicates how many documents the current word has appeared in the training set, and the TF value indicates the frequency of the word in the current document. Assuming that the document is represented as an ordered combination of all word word vectors, i.e., $d = \{d_1, d_2 ... d_n\}$. Then the vector representation of the document is shown in Equation (26):

$$V = \frac{\sum_{i=1}^{n} w(i) d_i}{\sum_{i=1}^{n} w(i)} \tag{26}$$

where V denotes the semantic vector of the document, $w(\cdot)$ denotes the weight of the i th word in the document, in this paper, we use the TF-IDF metrics as the weight function, and d_i denotes the word vector of the i th word in the document.

III. A. 3) Semantic Coherence Calculation

Based on the above method, this paper obtains the fused semantic representations of source language phrases and context information as well as the semantic representations of target candidate phrases. On this basis, this paper uses a multilayer perceptron to calculate their semantic matching degree, which indicates the probability of a source language phrase being translated into a target candidate phrase in a given context.

In this paper, the semantic feature vectors of the source and target languages are spliced together as inputs to the multilayer perceptron. The first layer of the multilayer perceptron first nonlinearly varies the spliced feature vectors to get the hidden layer state. The details are shown in Equation (27):

$$h_c = \phi(w_c \cdot [s_{\hat{f}_i} : t_{\hat{e}_j}] + b_c) \tag{27}$$

where $s_{\hat{f}_t}$ denotes the fused semantic features of the source language phrase in context, and $t_{\hat{e}_j}$ denotes the semantic features of the target candidate phrase.

The second layer of the multilayer perceptron takes the above hidden layer as input, and gets a new hidden layer after a nonlinear transformation and then undergoes a linear transformation to reach the output layer. The output layer has only one node, which represents the semantic matching score of the phrase pair in the context. The specific formula is shown in (28) below:

$$score(s,t) = W_{l2}[\phi(w_{l1} \cdot h_c + b_{l1})] + b_{l2} \tag{28}$$

III. A. 4) Semantic Coherence Model Training Strategy

The training goal of the context-dependent phrase-pair semantic matching model proposed in this paper is to assign high matching scores to correct translation results of source language phrases in specified sentence and document contexts, and low matching scores to incorrect translation results. Based on the idea of pairwise ranking learning, this paper transforms the problem of ranking all candidate translations of a source language phrase into a problem of ranking two-two candidate translation pairs. First, this paper constructs two pairs of comparable candidate translations, i.e., the correct translation result of a source language phrase and an incorrect target translation result in a specified context. Second, the above context-dependent phrase pair semantic matching model is adopted to obtain the semantic features of the source language phrase fused with the context and the target candidate phrase, which are used to construct the ternary (s, t^+, t^-), where s denotes the semantic feature vector of the source language phrase fused with the context, t^+ the feature vector of the correct target translation, and t^- the feature vector of the incorrect target translation. Finally, the pairwise ranking loss function is adopted as the loss function of the model, as shown in Equation (29):

$$L_\theta = max\{0, 1 + score(s,t^-) - score(s,t^+)\} \tag{29}$$

where $score(s,t)$ denotes the semantic matching degree of the defined phrase pairs according to Eq. (29). θ deotes all the parameters of the neural network model, including the sentence and target phrase convolutional

network (two-layer convolution, two-layer pooling), the shallow network for fusion of sentence and document information, and the parameters of the multilayer perceptron.

In this paper, we use the above methods to construct the training dataset and minimize the target loss function training to obtain the context-dependent phrase pair semantic matching model. The loss function on the full training set is defined as shown in Equation (30):

$$L = \sum_{(s,t^+,t^-)\in C} L_g(s,t^+,t^-) \tag{30}$$

In this paper, we use mini-batch gradient descent algorithm to optimize the parameters of the neural network model. When translating and decoding, the trained model and the context in which the phrase is located are used to determine the semantic match between the current phrase and all target candidate phrases, and are incorporated into the machine translation system as new features.

III. B. Optimized Evaluation of Semantic Coherence in English Long Text Translation

Coherence quality analysis is the most important part of all coherence analysis models, and each coherence analysis model has a different approach. Most of the traditional sentence graph models use the feature of average outgoingness to measure the quality of coherence of English texts, but such an approach its experimental results are not good enough to accurately capture the coherence information of English texts. An important assumption of this paper is that a coherent text follows a specific logic and coherence pattern between words or sentences within its discourse, based on which we adopt the method of frequent subgraphs to capture the coherence pattern in the text, which is an important foundation of the model of this paper for the analysis of the semantic coherence quality of English texts. In the sentence semantic graph, the coherence information of the text is reflected as the connection patterns between sentence nodes and the distributional differences of the weight values of sentence edges, so this paper analyzes the coherence quality of English texts by capturing the frequency of these frequent subgraph patterns and the semantic values of the subgraphs. Considering the different overall distributions of frequent subgraph frequencies, graph signatures and subgraph semantic values in coherent and incoherent English texts, the coherence quality of English texts is analyzed by mining the frequently occurring three-node and four-node subgraph patterns in the semantic graph of sentences and treating their probabilities, subgraph semantic values, etc. as coherent features. The specific analysis process is as follows:

(1) Use a large number of English texts with good coherence as a training set to train, and count the frequency of all three-node and four-node subgraphs appearing in the training set;

(2) Set the frequency coefficients to filter out the frequent subgraph patterns that appear frequently in the training set, and calculate the occurrence probability of each frequent subgraph pattern to generate our frequentist subgraph model, which is used as the frequent subgraph distribution feature of English texts with good coherence quality;

(3) Extract the graph signature and subgraph semantic value information in the sentence semantic graph representation of the English text to be analyzed:

(4) Combining the frequent subgraph related calculation method of LeoBorn et al. to design an algorithm to analyze the English text's coherence quality by using the distributional features of frequent subgraphs in the sentence semantic graph. Its calculation formula is as follows:

$$CoherenceScore(G) = \frac{\sum_{j=1}^{n}\lambda_j \sum_{i=1}^{m} P(sg_i)\times\phi(sg_i,G)\times SemanticValue(sg_i)}{SentenceNum(G)} \tag{31}$$

In Equation (31), m is the total number of frequent subgraphs of k nodes in the English text to be analyzed, n is the number of values of k, which has the value of 2 in this paper, $P(sg_i)$ is the frequency of frequent subgraphs for the i th k node in the English text to be analyzed, $\phi(sg_i,G)$ is the number of times that the frequent subgraph sg_i in the graph G, $SemanticValue(sg_i)$ is the subgraph semantic value of frequent subgraphs, and $SemtenceNum(G)$ is the number of nodes of the semantic graph of a sentence. Finally, in this paper, the maximum-minimum normalization algorithm is used to normalize the coherence quality score $CoherenceScore(G)$ of English texts between 0 and 1.

The final value obtained is used to represent the coherence quality of English text. The closer the value is to 1, the better the quality of coherence of the English text.

IV. Coherence Analysis of English Long Text Translation with Multi-attention Mechanisms

IV. A. Model parameterization

IV. A. 1) Experimental results

In this paper, the model uses Moses model to label the data and remove more than 80 labeled sentences from the source language side and the target language side with the following steps:

(1) Data screening. Firstly, the sentences with more than 80 markers are deleted, after that the data with garbled codes are deleted, and finally the sentences with errors are deleted by manual screening.

(2) Segmentation. For Chinese this paper uses jieba participle.

The model of this paper is optimized based on the Transformer model, using PyTorch 0. 4. 1 in the Fairseq toolkit to implement the model of this paper. The BLEU value is utilized as the evaluation metric of this paper through the method of small lattice network search. In this paper, Bi-Dependency and Pascal are chosen as the benchmark models for comparison experiments, and the experiments are based on the Transformer architecture, respectively. All experiments are performed on a single NVIDIA RTX 2070 SUPER GPU. In this paper, 3800 warm-up optimizations were performed according to the learning schedule using the hyperparameter settings in the latest Tensor2Tensor. A label smoothing rate of 0.15 is used during training. a beam search with beam size 5 and length penalty of 0.5 is used for validation. The learning rate used in this paper is 0.0008, batch size max-tokens is 4093 and dropout is 0.25. In the experiments on compressed data, 7800 warm-up optimizations are used in this paper.

In this paper, experiments were conducted on Newsdev2023 dataset, CWMT dataset, IWSLT14 dataset, and compressed Chinese-Thai and Chinese-English datasets, respectively, and Table 1 shows the BLEU results of translations with different models.

Transformer's BLEU values for Chinese-English and English-Chinese translations on CWMT are higher than Newsdev2023 by 6.7199 and 14.9453 respectively, and higher than IWSLT14 by 16.834 and 25.0154.This paper has improved the translation quality of English long text by incorporating the multi-head attention mechanism, which indicates that the proposed multi-head self-attention mechanism is effective.

Table 1: BLEU results of different model translations

Model	Newsdev2023		CWMT		IWSLT14	
	Chinese-English	English-Chinese	Chinese-English	English-Chinese	Chinese-English	English-Chinese
Bi-Dependency	21.2966	19.1655	28.3485	34.3488	10.9845	9.3485
Pascal	21.6333	19.5636	28.6366	34.6348	11.3448	9.6152
Transformer	22.0789	19.8486	28.7988	34.7939	11.9648	9.7785

IV. A. 2) The effect of multiple attention on translation results

In order to verify the reasonableness of the Chinese-English neural machine translation method based on multi-head self-attention mechanism proposed in this paper, experiments on the influence of two-way dependency information, fusion of two-way dependency information in different layers of multi-head attention mechanism and Gaussian weight function on the translation effect of the model were designed respectively.

In order to verify the effectiveness of fusing source language bidirectional dependency information, this paper conducts experiments on the effectiveness of fusing bidirectional dependency information on the Chinese-English dataset. The definition of “Transformer+ Multiple-focus” represents the model framework of this paper, and the definition of “Pascal” represents the fusion of only the information of the direction from the child word to the parent word in the dependency knowledge. Define “Bi-Dependency” to mean that only the information from the parent word to the child word direction in the dependent knowledge is fused. Table 2 shows the comparison of the BLEU values of the fused single/multiple-attention mechanism.

Transformer+Multiple-focus achieves the best results: on the Chinese-English translation task, the BLEU improves by 0.7885 and 0.7 compared to the Bi-Dependency model, and by 0.4488 and 0.799 compared to the Pascal model.On the compressed dataset, the BLEU value of Transformer+Multiple-focus also has a large improvement. The Transformer+Multiple-focus translation model proposed in this paper achieves the highest BLEU value and the best translation effect on the bi-directional Chinese-English translation task, indicating that the integration of the multiple-attention mechanism at the source language side is of greater help to the neural machine translation task.

Table 2: Contrast of the bleu value of a fusion single/long attention mechanism

Model	Chinese-English	English-Chinese	Chinese-Englishsmall	English-Chinesesmall
Bi-Dependency	21.2969	19.1486	10.9566	9.3488
Pascal	21.6366	19.0496	10.9966	9.6363
Transformer+Multiple-focus	22.0854	19.8486	11.9625	9.7856
Transformer	21.6788	19.5699	11.3165	9.6548

IV. A. 3) Fusion in different attention layers

In this paper, we also conducted self-attention mechanism layer experiments on the Chinese-English dataset at different layers of the multi-head attention mechanism to verify at which layer fusing semantic knowledge is more effective, and the experimental results are shown in Table 3.

The Transformer model achieved the best results in fusing semantic knowledge at the first layer of the multi-head attention mechanism, with BLEU values of 22.0885, 19.8215, 11.9655, and 9.7485, respectively.Compared with the lowest results, on the Chinese-English translation task, the BLEU values were improved by 0.6187 and 0.9679, respectively. After compressing the data, 1.4799 and 0.37 BLEU values were boosted, respectively. The performance of the model on the test set decreases significantly when the Transformer is placed in a lower layer. Such a result confirms that more attention in the first layer is focused only on the word itself that needs to be translated, rather than its context. It can be inferred that incorporating syntactic correlation in the first layer can be effective in learning word representations, thus further improving the translation accuracy of the Transformer model.

Table 3: Contrast of BLEU values in different attention-level

Attention layer	Chinese-English	English-Chinese	Chinese-Englishsmall	English-Chinesesmall
1	22.0885	19.8215	11.9655	9.7485
2	21.4698	18.9358	11.1248	9.6485
3	21.7496	19.4088	10.4856	9.6748
4	21.4936	18.8536	10.9466	9.3785
5	21.6336	18.9485	11.1648	9.7299
6	21.4899	18.9699	10.9485	9.7293

IV. B. Semantic Coherence Analysis Experiments and Analysis

IV. B. 1) Experimental data sets

In the experiment of analyzing text semantic coherence three corpora are used to solve different sub-tasks, which are GCDC dataset, CLEC corpus and TECCL corpus, and the fusion model of Transformer with the multi-head self-attention mechanism is trained and tested on the corpora. The components of the discourse parse tree include structural segments, nuclear tags and relational markers, in order to observe the effects of these three components on the semantic coherence of the text, an ablation experiment is conducted by removing structural segments, nuclear tags and relational markers from the Transformer model in the sub-datasets of the GCDC corpus, Yahoo, Clinton, Enron, and Yelp, respectively. .

IV. B. 2) Ablation experiments

The accuracy of the specific model is shown in Table 4 and the F1 value evaluation metrics are shown in Table 5 (all data in the table are in the form of percentage (%)).

From the experimental results, it is concluded that the Transformer model achieves an average accuracy of 56.8495% and an average F1 value of 45.5499% in each domain dataset. The overall model embedding the basic discourse units into the discourse parse tree compared to only the topology of the discourse relation tree (t) into the neural network model, the kernel sex labeling model (ns) with the addition of the discourse parse tree, and the relational labeling model (re) with the addition of the discourse parse tree in the categorization task compared to the categorization task the accuracy was improved by 9.387 percentage points, 7.8037 percentage points, respectively, 4.5 percentage points. Therefore, all components of the discourse parse tree are crucial for semantic coherence analysis, and the more significant influences are nuclear labels and discourse relations, the removal of which leads to a decrease in model performance. Since the Transformer model can only analyze semantic

information from the bottom-up, and the part-of-speech parser also parses from the sentence perspective, the model is a local coherence study in the overall view. Therefore, a hierarchical network model based on the multi-attention mechanism is integrated on the basis of the Transformer model, and the model better captures the global semantic coherence at the three levels of sentence paragraph and paragraph document from the perspective of the coherent structure of the paragraph or the overall text, so that the accuracy of the semantic coherence categorization task is improved to 60.2485%, and the F1 value reaches 49.4955%.

Table 4: The accuracy rate of ablation experiments conducted by the semantic coherence model

Model	t	ns	re	e	Yahoo	Clinton	Enron	Yelp	Overall
Transformer	√				38.1845	53.6158	44.2485	53.7485	47.4625
Transformer	√	√			44.8486	54.5696	44.7498	52.1635	49.0458
Transformer	√	√	√		48.3485	56.3158	47.4985	56.9552	52.3495
Transformer	√	√	√	√	53.9663	59.8969	54.8485	58.6155	56.8495
Multiple-focus					58.9345	62.1325	56.9315	58.5366	59.1348
Transformer+Multiple-focus	√			√	60.2486	60.9486	56.4985	57.5152	58.8985
Transformer+Multiple-focus	√	√		√	61.2978	61.8415	56.4936	58.7985	59.5485
Transformer+Multiple-focus	√	√	√	√	61.7899	63.7984	56.5596	58.9694	60.2485

Table 5: The F1 value of the ablation experiment conducted by the semantic coherence model

Model	t	ns	re	e	Yahoo	Clinton	Enron	Yelp	Overall
Transformer	√				29.7485	39.2489	34.0495	41.7966	36.2455
Transformer	√	√			34.1648	40.7498	35.5352	41.4955	37.9485
Transformer	√	√	√		42.9485	42.5988	35.9542	43.5486	41.2488
Transformer	√	√	√	√	46.5299	44.7498	44.9485	45.8496	45.5499
Multiple-focus					46.6252	54.6486	45.6489	46.0866	48.2485
Transformer+Multiple-focus	√			√	46.8485	52.5969	44.9485	45.8595	47.5485
Transformer+Multiple-focus	√	√		√	49.6486	53.0469	45.4985	46.2496	48.6315
Transformer+Multiple-focus	√	√	√	√	51.8998	54.5645	45.1348	46.0495	49.4955

IV. B. 3) Accuracy

The accuracies of the four models for classifying text semantic coherence on the four sub-datasets are shown in Fig. 3. The EGridConv model is to represent the English text in the form of a solid grid and analyze the local coherence of the text through a convolutional neural network, and the CohLSTM model is to use recurrent neural networks to capture the textual contextual information, and to cumulate the states of the two most similar recurrent networks in the sentence to obtain the semantic The CohLSTM model uses recurrent neural network to capture the textual contextual information, and accumulates the two most similar recurrent network states in the sentence to get the semantic information, and then encodes the change pattern of the semantic information by convolutional neural network to represent the coherence of the text.

It can be seen that no model consistently gives the best results in each dataset, as the data from the four domains have different contexts involved, leading to differences in the criteria learned by the models when it comes to semantic understanding. Nevertheless, the fusion model proposed in this paper has the best average performance in the classification task, with an improvement of 8.029 percentage points compared to the EGridConv model, 6.1759 percentage points compared to the CohLSTM model, and 0.9661 percentage points compared to the Avg-XLNet model. For the labels low coherence, average coherence and high coherence, the lower performance of all models in the three classifications is due to the fact that the models have difficulty in correctly classifying the text with average coherence, which is caused by the fact that the number of training samples in this category is difficult to learn its features with a small number of training samples.

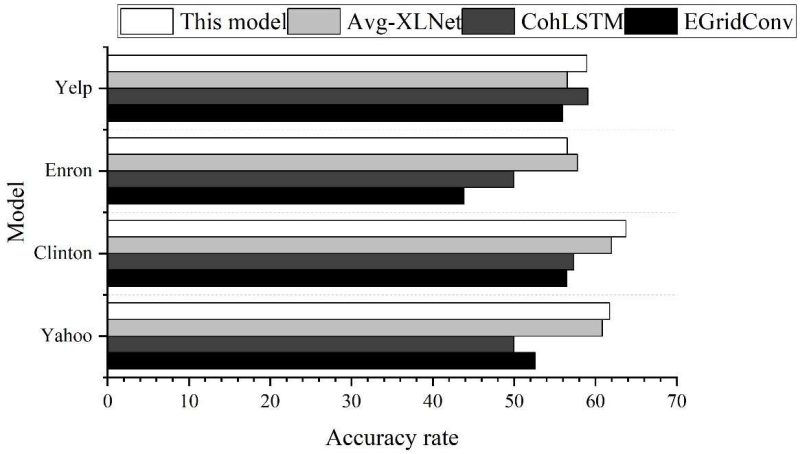

Figure 3: Text semantic coherence classification accuracy

IV. B. 4) Semantic coherence modeling vs. manual scoring

By comparing the machine scoring with the manual scoring, on the one hand, we can analyze the effect of the wrong words in the translated text on the semantic coherence of the text, and verify that this paper's model corrects the words before understanding the semantic coherence is scientific, and on the other hand, we test the feasibility of the present model in the practical application. The 700 translated texts were selected from the CLEC corpus containing wrong word annotations for experiments, and the Pearson's correlation coefficient of the unsemantic coherence model was 0.6848, and the Pearson's correlation coefficient of the semantic coherence model with the addition of word correction algorithms was 0.7498, and the difference of the Pearson's correlation coefficients was about 0.065 when comparing the addition of semantic coherence model and the human teacher scoring, the difference of Pearson's correlation coefficients was about 0.065 for texts with the inclusion of wrong words in the model. The occurrence of erroneous words in the text has a greater impact on the semantic coherence of the text, and the inclusion of word spelling errors as a reference factor in the analysis model helps to assess the coherence of the text more accurately. For the feasibility analysis of the model, 700 compositions from the TECCL corpus were selected for model testing, and the scatter plot of the experimental comparison between the automatic model scoring and manual scoring by teachers is shown in Figure 4.

The overall level of machine scoring is higher than that of manual scoring, and there is a large gap between the model and the teacher's scoring results in this paper, but the overall view is relatively similar, after all, semantic coherence is an abstract concept, and the manual correction of translated compositions will be affected by a lot of subjective factors, such as different degrees of coherence requirements, too many incorrect words that make it difficult for the teacher to understand the content of the author's expression, etc. The second corpus is also a manual process, which is not a simple process. Secondly, the corpus is also manually labeled, and the existence of a small number of large discrepancies in the marking points is acceptable. The average absolute error between the scores of the English compositions scored by the model and the scores of the compositions corrected by the teachers is 3.1685, the error value is less than 4, and there is not much difference between the results of manual correction and the results of machine correction, and the Pearson correlation coefficient of the scores of the two scores is 0.6848, the value of which ranges from 0.6 to 0.8 and belongs to the strong correlation. In summary, this model has good practical application value.

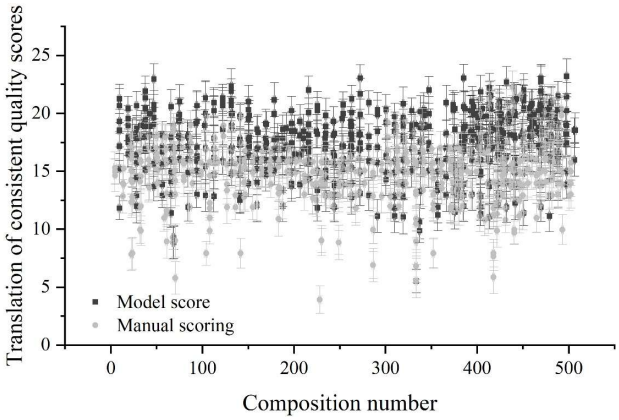

Figure 4: The comparison of the scores

V. Conclusion

The semantic coherence optimization model for English long text translation based on multi-head self-attention mechanism has achieved remarkable results in multiple dimensions. The experimental results show that the model incorporating the multi-head attention mechanism achieves a BLEU value of 28.7988 for Chinese-English translation on the CWMT dataset, which is significantly improved compared with the traditional Transformer model. In the semantic coherence analysis task, the accuracy of the full model is 60.2485%, which is 9.387 percentage points higher than the base model using only the topology, proving the importance of the components of the discourse parse tree. The ablation experiments further validate the contribution of each component of the model, where the removal of nuclear and relational labels leads to a significant performance degradation. The multi-head attention mechanism works best when fusing semantic knowledge in the first layer, with a BLEU value of 22.0885, validating the effectiveness of early semantic fusion. The correlation analysis between the model and manual scoring shows that the Pearson correlation coefficient is 0.6848 and the average absolute error is 3.1685, indicating that the model has good practical value. The method of capturing text coherence features through frequent subgraph patterns effectively improves the overall quality of long text translation and provides new ideas and methods for the development of neural machine translation technology. The model not only makes a breakthrough in translation accuracy, but also shows superior performance in semantic coherence maintenance, which lays a solid foundation for practical application.

Funding

This work was supported by Anhui Philosophy and Social Sciences Planning Project: Comparative Study of the Chinese Translation of the Western Philosophical Classic The Enneads (AHSKY2020D83), Bengbu University Research Team (BBXYKYTDxj13).

References

[1] House, J. (2014). English as a Lingua Franca and Translation 1. In English as a Lingua Franca (pp. 279-298). Routledge.
[2] Vinall, K., & Hellmich, E. (2022). Do you speak translate?: Reflections on the nature and role of translation. L2 Journal: An electronic refereed journal for foreign and second language educators, 14(1).
[3] Jun, X. (2018). A study on semantic coherence in discourse translation from the perspective of thematic progression patterns. Journal of Literature and Art Studies, 8(6), 892-896.
[4] Deng, N., Wang, Y., Huang, G., Zhou, Y., & Li, Y. (2023). Semantic Coherence Analysis of English Texts Based on Sentence Semantic Graphs. EAI Endorsed Transactions on Scalable Information Systems, 10(5).
[5] Hadla, L. (2015). Coherence in translation. Research on Humanities and Social Sciences, 5(5), 178-184.
[6] Jing, H. (2023). Textual Cohesion and Coherence in Translation of Historical Text: An EC Translation Case Study of The Revolution of 1848 (Excerpts). London Journal of Research In Humanities and Social Sciences, 23(23), 21-32.
[7] Al-Kharabsheh, A., & Hamadeh, N. (2017). Shifts of Cohesion and Coherence in the Translation of Political Speeches. Advances in Language and Literary Studies, 8(3), 100-112.
[8] Mehri, A., Farokhipour, S., & Sajjadi Dehkharghani, S. M. (2020). On the Impact of Global Coherence on Translation Comprehension of the Holy Quran: A Case Study of PhD-ESP Learners. Linguistic Research in the Holy Quran, 9(2), 31-36.
[9] Najafi, E., Valizadeh, A., & Darooneh, A. H. (2022). The effect of translation on text coherence: a quantitative study. Journal of Quantitative Linguistics, 29(2), 151-164.
[10] Guo, P. (2024). Construction of Semantic Coherence Diagnosis Model of English Text based on Sentence Semantic Map. Scalable Computing: Practice and Experience, 25(1), 327-339.
[11] Károly, K. (2020). Logical relations in translation: The case of Hungarian-English news translation. In Contemporary Approaches to Translation Theory and Practice (pp. 93-113). Routledge.

[12] Onn, W. J. (2018). The Semantics of Logical Connectors: therefore, moreover and in fact. Russian Journal of Linguistics, 22(3), 581-604.
[13] Li, X., & Kim, M. (2021). A descriptive study on Chinese-English translation choices for logical meanings. Systemic Functional Linguistics and Translation Studies, 123-142.
[14] Ma, Y. (2022). THE INFLUENCE OF COLLEGE ENGLISH TRANSLATION AND INFORMATION TEACHING INNOVATION ON COLLEGE STUDENTS'THINKING LOGIC OBSTACLES. Psychiatria Danubina, 34(suppl 1), 683-685.
[15] Yang, J., Husin, N., & Yusof, A. M. (2025). Exploring the Consistency between Translation Style Attitudes and Practices. International Journal of Language Education and Applied Linguistics, 15(1), 40-51.
[16] Prieto Ramos, F. (2021). Ensuring consistency and accuracy of legal terms in institutional translation: The role of terminological resources in international organizations. In Institutional Translation and Interpreting. Taylor & Francis.
[17] Ferro, M. J. (2025). Corpus Linguistics and Literary Translation: A Contribution to Consistency. In Exploration of the Intersection of Corpus Linguistics and Language Science (pp. 327-356). IGI Global Scientific Publishing.
[18] Klaudy, K., & Károly, K. (2017). The text-organizing function of lexical repetition in translation. In Intercultural Faultlines (pp. 143-159). Routledge.
[19] Hassan, A. J. (2015). Translating Arabic verb repetition into English. Arab World English Journal (AWEJ), 6(2), 144-153.
[20] Gunawan, H. (2020). The Effectiveness of Repetition in Increasing Students Vocabulary. SIMETRIS, 14(2), 49-52.
[21] Qiu, D., & Yang, B. (2022). Text summarization based on multi-head self-attention mechanism and pointer network. Complex & Intelligent Systems, 1-13.
[22] Liu, T., Zou, B., He, M., Hu, Y., Dou, Y., Cui, T., ... & Wang, D. (2023). LncReader: identification of dual functional long noncoding RNAs using a multi-head self-attention mechanism. Briefings in bioinformatics, 24(1), bbac579.
[23] Cui, H., Iida, S., Hung, P. H., Utsuro, T., & Nagata, M. (2019, November). Mixed multi-head self-attention for neural machine translation. In Proceedings of the 3rd Workshop on Neural Generation and Translation (pp. 206-214).
[24] Lihua, Z. (2022). Analysis of English Translation Model Based on Artificial Intelligence Attention Mechanism. Mathematical Problems in Engineering, 2022(1), 9669152.