ingegneria sismica
International Journal of Earthquake Engineering
Anno XLIII - Num. 1 - 2026

A Context-Aware English Translation Accuracy Improvement Strategy Based on Deep Reinforcement Learning

Ping Yin[1] *and Juanyin Liu*[1,*]

[1] Teaching Department of Basic Courses, Hebei Vocational University of Industry and Technology, Shijiazhuang, Hebei, 050091, China

SUMMARY: *The paper presents a neural machine translation system with a context-sensitive mechanism. It uses a recursive self-encoder to learn the representations of English sentences, which are then contextualized by topic distribution. Furthermore, the parameters of the model are optimized by applying a deep reinforcement learning algorithm, which enhances the accuracy of the model further. Experiments are performed on different datasets to assess the performance of the proposed model. It was found that the model exhibits its highest level of translation performance at the cosine similarity threshold of 0.9, and that there are notable improvements in the quality of translations following fine-tuning of the parameters utilizing deep reinforcement learning. Bleu scores increase by 2.00 - 4.84 points, which indicates the efficiency of the fine-tuning stage. Besides, the incorporation of the context-sensitive bilingual-constrained recursive autocoder boosts the bleu scores up to 5.23 - 7.76 points over the baseline and variant models. On the whole, the addition of deep reinforcement learning and the context-sensitive method makes the model much more effective at producing English translations that are correct.*

KEYWORDS: *Recursive self-encoder; Neural machine translation model; Deep reinforcement learning; Parameter optimization; English translation*

1 Introduction

In the classical English translation, the individual sentences are usually translated separately, disregarding their contextual connection with the rest of the text. Nevertheless, since English is an informational language, with connections between all the sentences, translation of separate sentences might lead to a semantic disconnection and miss-alignment. Such a strategy can be ineffective in accurately reflecting the purpose of the original text. The solution to this problem was to develop a great deal of interest in improving the quality and accuracy of English translation by using context awareness.

The concept of context awareness in the field of translation involves considering the contextual data around a sentence and using the same to enhance the precision of its translation in machine translation systems [1]. Context-awareness integration enables machine translation systems to have a deeper insight into the semantic and logical structure of the overall text and consequently provide more precise and fluent translations [2, 3]. In English translation, Guo noted the fact that ambiguous sentences tend to generate mistranslations. To overcome this, a context-aware translation system with deep learning has been developed, which combines an attention mechanism, context embedding and grammatical-semantic analysis to eliminate ambiguities. The results of experiments prove that this method is much more effective in terms

*eiessay777@126.com
https://doi.org/10.65102/is2026476
Published: 30 April, 2026

of translation accuracy and fluency especially in cases of low-resource and idiomatic texts [4]. Document-level machine translation has been found useful in enhancing the quality of translation by Gete et al. Nevertheless, it is still unclear how context-aware models can be used in ways that are efficient. Using an input splicing model, they analyzed the role of data bias, syntactic role of previous words and the complexity of the context in English-German and English-French translations. Their results indicate that although the model performs better than the baseline in context, its sensitivity to regular syntactic constructions like subject-pronoun relations is present and the performance drops when the structures are significantly different [5]. Rikters et al. noted that the majority of machine translation research work deals with sentence-based strategies, which are quite successful in resource-rich languages but experience difficulties when translating through sentence boundaries. They constructed a Japanese-to-English conversation corpus which contains contextual knowledge and manually annotated important areas that require context to attain a correct translation. Their research showed how contextual information could help enhance translation quality using training models [6].

In addition, Vu et al. emphasized the problems of context-aware machine translation models especially in the case of long contexts or complicated models where improper usage of features may cause poor performance. In order to overcome this problem, they suggested a model that is able to enhance translation choices through the prediction of co-referential properties of the input. The model was evaluated using different datasets like the English-German and English-Russian datasets and it was found to be significantly better (more than 1 points in the BLEU score) than the other models [7]. Su et al. created a topic model depending on context that was aimed at improving lexical choice by representing the connection between local context and global topic. The authors used the Gibbs sampling algorithm to perform inference, and their experiments showed that this method was superior to the conventional ones. Moreover, the association simulation also enhanced translation quality [8]. Hu et al. proposed the context-aware evaluation metric Cont-COMET based on the COMET framework, while considering the pre- and post-sentence contexts and aligning them with the manual annotation settings, and extracting the key information through the content selection method, which was experimentally shown to be improved at both the systemic and segmental levels [9]. Chen et al. showed that effective modeling of global context is the key challenge in document-level machine translation, and proposed to model the source language context from sentence and lexical two-layer level: the sentence level extracts the global information related to the current sentence, and the lexical level calculates the associations with intra-sentence words, and fuses the two into the Transformer, and verified that this layered context modeling can significantly improve the translation performance, and that the two levels of context are complementary [10]. However, context-awareness in practical applications also faces some challenges that lead to the problem of low translation accuracy. First of all, it is difficult to effectively capture the contextual relationships between sentences. There may be differences in linguistic structure and logical relationships between different languages [11, 12]. Second, the problem of long dependencies between sentences is also a key challenge. In long text translation, the semantic relationship between sentences may appear more complex [13, 14]. The significant progress in deep reinforcement learning in the field of natural language processing has created new opportunities to improve the context-sensitive translation quality in English translation.

One of the most significant areas of machine learning is deep reinforcement learning, which combines the strong representational capacity of deep learning with the decision-making performance of reinforcement learning. It allows the model to learn and optimize its decision-making strategy on its own in complex situations through the simulation of the human learning process. Consequently, it has become possible to produce more efficient and accurate translation results [15-18]. Deep reinforcement learning makes it possible to dynamically optimize the translation strategy by casting the English translation process as a sequential decision-making problem. The given strategy enhances the awareness of context, which in turn leads to the development of more accurate translations.

The presented in this article neural machine translation model starts with using a recursive autoencoder to obtain English phrase representations. This contextual information is subsequently incorporated into these representations by modeling the topic context of every phrase. The composite representations of phrases are sent through a neural network to produce context-sensitive translations. Deep reinforcement learning is used to optimise the network parameters of the model more, particularly deep reinforcement learning is utilised to overcome problems like under-translation and omissions through the use of reward and policy gradient mechanisms. The experiments are aimed at examining the effect of deep reinforcement learning and context-sensitive approaches on the accuracy of English translation with different points of view.

2 Context-aware translation model based on deep reinforcement learning

The present section presents a bilingual-constrained recursive self-encoder, which is one of the contextual methods suggested to improve a neural machine translation model. Also, deep reinforcement learning algorithm is used to optimize the network parameters of the model to enhance the quality of its translation.

2.1 Context-aware strategies empowering neural machine translation models

Phrases are used as the building blocks of phrase-based machine translation. With this paper, introducing a context-aware bilingual-constrained recursive self-encoder (TBARE) can be used to help neural machine translation models learn more about the syntactic and grammatical properties of phrase-to-phrase translation rules. This model is aimed at facilitating the acquisition of bilingual phrase representations and making the translation easier. The method builds upon the bilingual recursive autoencoder (BRAE) by considering some contextual data during learning of representations. The overall architecture of the TBRAE model is shown in Fig. 1. This model features the yellow section as the phrase representation model using a recursive autoencoder (RAE) and the lavender section indicates the context modeling module. Three major parts make up the model:

a. Two RAEs for learning the phrase representation at the source and target.

b. Two contextual representation models for modeling the topic context of the phrases at the source and target ends.

c. Bidirectional semantic constraints for minimizing the semantic distance of phrase pairs.

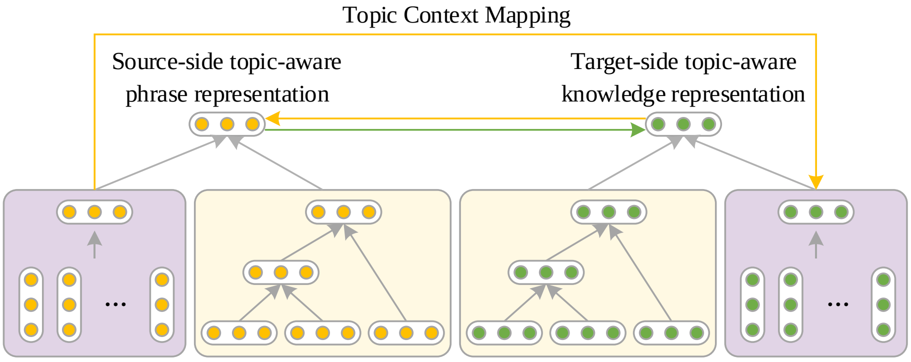

Figure 1: TBRAE model frame

2.1.1 Phrase representation modeling

In the TBRAE model, each word in a word list V is represented by an n-dimensional vector, and all the vectors form a word embedding representation matrix $L_w \in R^{n\times|V|}$. Considering that a phrase is a meaningful combination of its word sequences, RAE is used to learn the vector representation of the phrase. Suppose the input phrase consists of three words $(x_1, x_2, x_3)^*$, which are fed into the RAE to obtain its phrase representation. First for x_1 and x_2, its parent node y_1 can be obtained:

$$y_1 = f\left(W^{(1)}\left[x_1; x_2\right] + b^{(1)}\right) \tag{1}$$

$W^{(1)} \in R^{n\times 2n}$ and $b \in R^{n\times 1}$ are the parameters of the model. f is the activation function, which is used in the text as $\tanh(\cdot)$. The obtained y_1 is also a vector of dimension n, which agrees with the dimension of the input word vector. Here y_1 can be viewed as a new word combining x_1 and x_2. In order to measure the quality of the learned y_1, i.e., how much of the original information is retained, the x_1 and x_2 are reduced using y_1 as in the case of the autocoder:

$$\left[x_1', x_2'\right] = f\left(W^{(2)} y_1 + b^{(2)}\right) \tag{2}$$

Here x_1', x_2' correspond to the reduced x_1, x_2, $W^{(2)} \in R^{2n\times n}, b^{(2)} \in R^{2n\times 1}$ respectively. Similarly, for y_1 and x_3 apply the formula to obtain their vector representations y_2.

The above process is repeated until finally only one node remains. In order to obtain the optimal representation for each phrase, a greedy strategy is used to minimize the reduction loss, which is defined for a phrase p by the RAE:

$$E_{rec}(p;\theta) = \sum_{(c_1,c_2)\in T(p)} \frac{1}{2}\left\|\left[c_1; c_2\right] - \left[c_1'; c_2'\right]\right\|^2 \tag{3}$$

where $T(p)$ is the tree obtained by greedy strategy and θ is the parameter of RAE model.

In order to make a reasonable distribution of the weights of all the nodes, the nodes are given the corresponding weights when calculating the reduction loss, assuming that the leaf nodes corresponding to c_1, c_2 are n_1, n_2 respectively, then the reduction loss for node y_1 is calculated as follows:

$$E_{rec}(y_1;\theta) = \frac{n_1}{n_1+n_2}\left\|c_1 - c_1'\right\|^2 + \frac{n_2}{n_1+n_2}\left\|c_2 - c_2'\right\|^2 \tag{4}$$

Similarly, for the problem where the value of the intermediate node may be small, the same normalization solution is used. Specifically, after obtaining y_1, $y_1' = \frac{y_1}{\|y_1\|}$ is used as the final representation of y_1. And so on, this is done for all the representations of the internal nodes of the tree.

2.1.2 Contextual representation modeling

Following the lead of topic-based machine translation, the topic distribution in a document is used to get the contextual details of the phrases in that document. Nevertheless, one of the main issues with neural networks is finding the right data representation. To make it easy to compute, every topic is considered as a word and every topic is represented as an n-dimensional vector. In analogy with words being vectors, the vectors of all topics taken together comprise a topic embedding matrix that we denote as $L_z \in R^{n\times|Z|}$.

Since the topic representation of a document is a probability distribution over all the topics, the vector representation of topics is weighted and summed according to the probability of the topics to get the topic representation of the document:

$$dc = \sum_z p\left(z\,|d\right)\cdot \vec{z} \tag{5}$$

where z and $\vec{z}$ correspond to the topic and the embedded representation of the topic, respectively. dc denotes the topic of the document.

At the time of training, the contextual representations of phrase pairs (f, e) can be obtained using the method mentioned above, where dc_f and dc_e are used to represent the topic contexts of phrases f and e. Since the acquisition of context relies on monolingual documents, additional corpus can be utilized to better train the topic model.

In order to obtain the topic distribution at the target end, a transformation relation of the topic contexts from the source to the target end can be learned. Specifically, for parallel phrase pairs (f, e) and their topic contexts (dc_e, dc_f), the topic context transformation relation is learned by minimizing the change loss:

$$E_{tcm}\left(f\,|e;\theta\right) = \frac{1}{2}\left\|dc_e - f\left(W_{f2e}^{(3)}\cdot dc_f + b_{f2e}^{(3)}\right)\right\|^2 \tag{6}$$

where $W_{f2e}^{(3)} \in R^{n\times n}$ and $b_{f2e}^{(3)} \in R^{n\times 1}$ are the parameters to be learned.

2.1.3 Bilingual semantic constraints

First, the topic context-aware phrase representation p_{dc} is obtained through a layer of fully connected networks:

$$p_{dc} = g\left(W^{(4)}\left[p; dc\right] + b^{(4)}\right) \tag{7}$$

Here, $W^{(4)} \in R^{n\times 2n}$ and $b^{(4)} \in R^{n\times 1}$ are network parameters with learning. Where p is the phrase representation learned using RAE, dc is the topic context corresponding to the obtained phrase, and p_{dc} is the phrase representation with topic context information.

There exist two directions for the computation of semantic distance, and the semantic loss is defined as the mapping of the source to the target, for example:

$$E_{sem}\left(f\left|e;\theta\right.\right) = \frac{1}{2}\left\|e_{dc} - f\left(W_{f2e}^{(5)} f_{dc} + b_{f2e}^{(5)}\right)\right\|^2 \tag{8}$$

Here, $W_{f2e}^{(5)} \in R^{n\times n}$ and $b_{f2e}^{(5)} \in R^{n\times 1}$ are the parameters to be learned.

Then, the semantic loss is also computed by maximizing the semantic spacing, which is obtained for positive samples (f, e) and negative samples (f, e'):

$$E_{sem}^{*}\left(f\left|e;\theta\right.\right) = \max\left\{0, E_{sem}\left(f\left|e;\theta\right.\right) - E_{sem}\left(f\left|e';\theta\right.\right) + 1\right\} \tag{9}$$

Here e' is a negative sample, another translation of f or obtained by replacing the words in e at random. Similarly, $E_{sem}^{*}\left(e\left|f;\theta\right.\right)$ can be obtained.

2.2 Model parameter optimization based on deep reinforcement learning

The research uses deep reinforcement learning to optimize and fine-tune the model parameters and specifically focuses on sentence-level BLEU score. This technique can be applied to alleviate the problems of overfitting and underfitting. Also, with the help of the strategy gradient algorithm, the model will increase the probability of producing better quality sentences in the process of iterative optimization, which finally leads to higher quality translations.

2.2.1 Estimation of reward awards

The probabilistic formulation of non-autoregressive translation has the following form:

$$P\left(Y\left|X,\theta\right.\right) = \prod_{i=1}^{T} p\left(y_i\left|X,\theta\right.\right) \tag{10}$$

In this case, X is taken as the input to the translation model, Y is used as the target output sentence, T is the length of the target sentence, and θ is used as the parameters of the neural network. i can be defined as the position i within the sentence, y_i as the predicted word at the position i, and $P(\ )$ as the probability function.

The expected loss gradient of the reinforcement learning approach described is:

$$\nabla_\theta L_\theta = -\sum_{Y} \nabla_\theta \prod_{i=1}^{T} p\left(y_i\left|X,\theta\right.\right)\cdot r\left(Y\right) \tag{11}$$

where $r(\ )$ denotes the reward computation function, whose input is the whole sentence Y and the output is the bleu value of this sentence. The ∇_θ denotes the gradient of the neural network parameter θ, and Y denotes the sentence obtained from reinforcement learning sampling.

The final objective of the optimization is to change the output of the neural machine translation model so that it gradually matches a greater BLEU score. The reward is computed as the mean of the instantaneous rewards achieved through various sampling iterations.

In the equation given, the predicted probability of the word i at each position is being updated through a gradient that is affected by the reward of the full sentence $r(Y)$, particularly the BLEU score of the sentence. Non-autoregressive models are simplified to the form:

$$\nabla_\theta L_\theta = -\sum_{i=1}^{T} \sum_{y_i} \nabla_\theta p\left(y_i \middle| X, \theta\right) \cdot r\left(y_i\right) \tag{12}$$

where $r(y_i)$ is the expected reward when the vocabulary y_i is fixed, as:

$$r\left(y_i\right) = \mathop{\mathbb{E}}_{y_{i+1}} \mathop{\mathbb{E}}_{y_{i+1:T}} r\left(Y\right) \tag{13}$$

$r(y_t)$ estimated by Monte Carlo sampling, i.e., words y_t in fixed positions t, other words $y_{1:t-1}$ and $y_{t+1:T}$ are obtained by sampling N times from a probability distribution $p\left(\cdot \middle| X, \theta\right)$, Y denotes the whole sentence, $r(y_t)$ the estimate is the average of the prizes $r(Y_1), r(Y_2), r(Y_3), \ldots, r(Y_N)$ of the N sampled sentences, and T denotes the number of words in the sentence. $p\left(y_t \middle| X, \theta\right)$ is the strategy, then replace the word in this position and repeat the sampling to evaluate the REWARD. after all the content in this position has been evaluated, $t+1$ looks at the next position and continues the same sampling and estimation.

2.2.2 Estimation of the expectation gradient

Finally the time complexity of the whole algorithm is $O\left(T_k \cdot n\right)$. The space complexity is $S(n) = O(1)$.

The gradient $\nabla_\theta L_\theta$ is estimated using the REINFORCE formula as:

$$\nabla_\theta L_\theta = -\sum_{t=1}^{T} \mathop{\mathbb{E}}_{y_t} \left[\nabla_\theta \log\left(p\left(y_t \middle| X, \theta\right)\right) \cdot r\left(y_t\right) \right] \tag{14}$$

The described approach is related to a gradient estimation algorithm which involves sampling the target word y_t and computing the expected gradient based on the log-probability gradient of y_t, which is multiplied by the reward r. Although the gradient estimate is not biased, it has a relatively large variance.

In order to solve this, the unbiased estimation is done by going through the top-k words followed by the estimation of the rest of the words using a single sampling iteration, as demonstrated in the following diagram:

$$\nabla_\theta L_\theta = -\sum_{t=1}^{T}\left(\sum_{y_t\in T_k}\nabla_\theta p\left(y_t\middle|X,\theta\right)\cdot r\left(y_t\right)+\left(1-P_k\right)\right. \\ \left.\cdot\mathop{\mathbb{E}}_{y_t\sim\tilde{p}}\left[\nabla_\theta \log\left(p\left(y_t\middle|X,\theta\right)\right)\cdot r\left(y_t\right)\right]\right) \tag{15}$$

Here, $p\left(\cdot / X,\theta\right)$ is the output probability distribution of the target word generated by the decoder at position t, T_k is the probability set of a series of top-k target words, and P_k is the sum of the probabilities in T_k. $\tilde{p}$ is the normalized probability distribution after removing the word probabilities from T_k.

The algorithm takes as input the output probability distribution $p\left(\cdot\middle|X,\theta\right)$, the traversal count k, and the number of samples n, and outputs the gradient estimate of step t.

The gradient estimation process of step t is first divided into two parts: traversal and sampling.

The algorithm first constructs the set T_k using the probabilities of the first K words, and then Algorithm 1 estimates the expected reward of the words in T_k. The cumulative gradient in T_k is obtained by traversing the words in T_k and accumulating the gradients of their probability functions, using the corresponding rewards as weights.

After a traversal process of gradient accumulation in T_k, the algorithm estimates the expected gradient of the words in the sampling process that do not belong to T_k. The algorithm obtains the probability distribution $\tilde{p}$ of the remaining words by masking the probabilities of the words in T_k. The words y_t from the distribution $\tilde{p}$ are sampled to compute the gradient of the log probability of y_t, and then the return of $r(y_t)$ is estimated. The weight of this estimate is $1-P_k$, where P_k is the sum of the probabilities in T_k. Finally, the estimated gradient is the sum of the gradients of the Top-k words and the sampled words.

The algorithm introduced in the paper focuses on tracing the gradients of major words because they have a leading role in the process of estimating gradients and estimating the gradients of minor words by a single sampling iteration.

After obtaining the gradient $\nabla_\theta L_\theta$, the neural network parameter θ is updated based on the following equation and fine-tuned to obtain the new neural network parameter θ_{new}:

$$\theta_{new} = \theta + \alpha\nabla_\theta L_\theta \tag{16}$$

In this case, α denotes the learning rate. The next step is to repeat the loop to conduct Monte Carlo sampling, estimate the reward, compute the expected gradient, and update the model parameters.

3 Translation accuracy experimental design and analysis of results

This part of the paper discusses the experimental verification of the context-sensitive translator model using deep reinforcement learning. It gives an exhaustive description of the experimental data, experimental procedures, measures of assessment, and outcomes.

3.1 Experimental data

In order to assess the quality of the suggested approach, the experiments mainly focused on using parallel datasets of low-resource languages, which were published by the International Workshop on Spoken Language Translation (IWSLT). Datasets used to train the translation system were IWSLT17, IWSLT15 and IWSLT13, which were chosen after careful screening. These datasets comprised seven languages namely; English, French Arabic, Czech, Vietnamese, Turkish and Polish and they had six language pairs of parallel data. Also, a parallel set of Chinese-English data, the CEEC dataset, was made to give a better assessment of the translation quality of the context-sensitive model using deep reinforcement learning. In Table 1 the statistics of the parallel datasets are shown in detail.

Table 1: Parallel data set statistics

Data set	Language pair	Data set size
IWSLT 17	French ↔ English	230k
	Arabic ↔ English	230k
IWSLT 15	Czech ↔ English	100k
	Vietnamese ↔ English	130k
IWSLT 13	Turkish ↔ English	130k
	Polski ↔ English	140k
CEEC	Chinese ↔ English	40k

3.2 Experimental setup

The learning rate change strategy during model training is inverse sqrt, using Adam as the optimizer. All the experimental data were processed with word splitting, special symbols processing, effective length sentence screening, parallel sentence ratio set to 2, effective length 1-200. the processing scripts involved are from the open source project Moses, each language limits the maximum number of characters to 6000 for the training batch, and the training is performed on dual GPU GeForce GTX 1080 Ti. The model with the best bleu value on the validation set is saved, and the saved models are model averaged to obtain the final model. To perform model testing, the translations are generated using a cluster search algorithm, with the size of the cluster set to 3. The translations are then removed from the subword sequence flags and reduced to the participle, and finally the bleu scores obtained from the models are computed using Sacre bleu, with a retained accuracy of two decimal places.

3.3 Translation quality indicators

The research paper utilizes the BLEU score as an indicator of assessing the translation accuracy of the neural machine translation model. Bleu is an automated measure of performance that was proposed by IBM to assess machine translation systems. The fundamental concept behind it is to determine the similarity of n-grams between the machine-produced translation and the reference translation. More overlapping implies greater quality of translation. With a reference translation $S_1, S_2, \cdots, S_m$, the n-gram accuracy of the machine translation can be computed with this formula:

$$P_n = \frac{\sum_{i \in n\_gram} \min\left(h_i(C), \max_{j \in m} h_i(S_j)\right)}{\sum_{i \in n\_gram} h_i(C)} \tag{17}$$

where i is an n-element grammar in the machine translation C, $h_i(C)$ denotes the number of times n-element grammar i appears inside the machine translation, and $h_i(S_j)$ denotes the number of times n-element grammar i appears in the j th reference translation. number of times. In order to address scoring bias, BLEU adds a penalty factor (BP) to the final score, which is defined as.

$$BP = \begin{cases} 1 & l_c > l_s \\ e^{1-\frac{l_s}{l_c}} & l_c \le l_s \end{cases} \tag{18}$$

Here l_c stands for the machine translation length and l_s stands for the acceptable length of the reference translation. Whenever there are several reference translations, the one nearest in length to the machine translation defaults to being used. When the machine translation is longer than the reference translation, the penalty factor BP is equal to 1 which means no penalty. The penalty factor is used only in a case where the machine translation is lower than the reference translation.

The entire definition of BLEU can be formulated by combining both the multivariate accuracy and the length penalty factor:

$$bleu = BP \times \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{19}$$

where p_n is the precision rate of the n meta-grammar and w_n is the weight of the n meta-grammar, which is generally set to be a uniform weight, i.e., $w_n = 1/N$ for any n.

3.4 Analysis of experimental results

3.4.1 Effect of Threshold Selection on Translation Performance

The model was trained with the help of a large-scale English corpus and the semantically similar words were chosen depending on the cosine similarity values that were higher than the defined threshold (≤ 1). The experiment threshold values were 0.5, 0.6, 0.7, 0.8, 0.9 and 1 and the semantic relatedness sets were produced based on these values. To evaluate how threshold choice affects the performance of the model in translation, a comparative study of the IWSLT17, IWSLT15, IWSLT13 and CEEC datasets was carried out. Five cross-validation tests were used to increase the reliability of the findings. The effect of the choice of the threshold on the process of translating is shown in Fig. 2 the shaded region is the error of the five tests. The model translation performance increased with increasing threshold values in all four datasets and peaked at the highest BLEU score of 0.9 after which the BLEU score started to decrease. This suggests that the best translation performance is attained when the threshold is fixed to 0.9. Moreover, reducing the threshold value leads to an increase in the number of semantically related words, which increases the size of the generated corpus. Under a threshold below 0.9, accuracy of the semantic word choice becomes the most important consideration in determining the quality of the translation. On the other hand, when the threshold is above 0.9, the size of the pseudo-parallel corpus generated is the key determinant of the performance of the translation model.

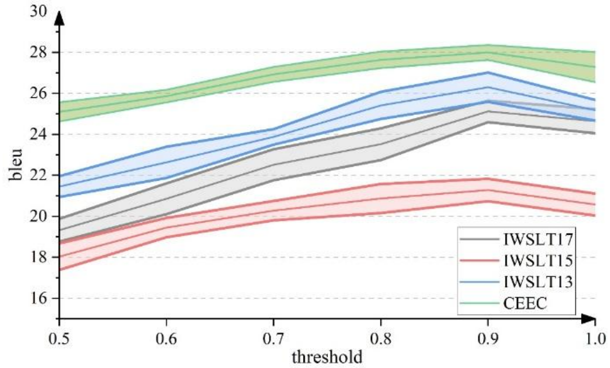

Figure 2: The effect of the valve value on the translation performance

In order to determine the effect of the last quantity of generated pseudo-corpora on translation quality, this part will train the translation model on various fractions (10, 20, 30, 40, 50, 60, 70, 80, 90 and 100) of the extra corpus. The outcomes displayed in Figure 3 demonstrate that with the growing amount of pseudo-corpus, the translation performance of the model takes the form of an inverted V: it initially improves and subsequently deteriorates. Optimal performance occurs at a selection of 30 percent of newly expanded corpus, which matches the results of earlier works showing that the performance of machine translation does not grow linearly with more new corpus being added.

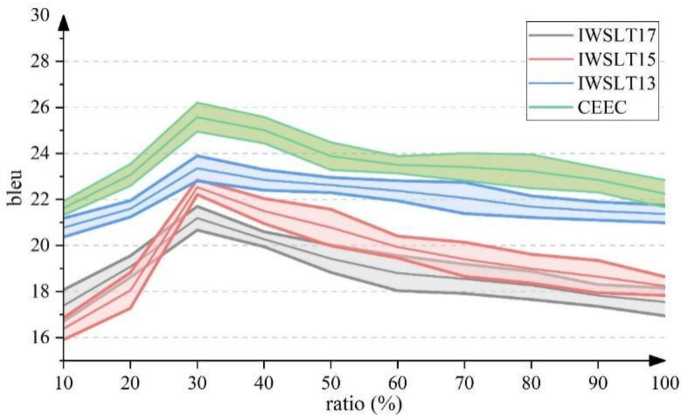

Figure 3: The effect of pseudo-language on the performance of model translation

3.4.2 Deep Reinforcement Learning to Improve Translation Quality

Tuning the parameters of the translation model has a great impact on the performance of the model. Directly fine-tuning the model to translate an English resource is a good approach. In this section, the BLEU scores of the translation model based on the four test sets discussed above will be assessed before fine-tuning, after direct fine-tuning and after the deep reinforcement learning fine-tuning of the model proposed in this study. To make sure that obtained results are correct, five cross-validation experiments were performed. The difference in the BLEU scores between the model in its original language pairs is shown in Table 2.

Direct fine-tuning of the translation model can successfully migrate the translation model to the target language pairs, but this comes at the expense of the performance of the other language pairs, with IWSLT17, IWSLT15, IWSLT13, and CEEC decreasing the bbu values by 2.81, 2.80, 4.27, and 4.88 bbu values, respectively. In contrast, deep reinforcement learning improves the bleu values by 2.51, 4.87, 2.00, and 4.30, respectively. deep reinforcement learning can find more suitable network parameters for the translation model through the reward mechanism and the expected gradient estimation, and it can take into account the fine-tuning of both the target language pair and the original language pair through the local parameter fine-tuning, and the fine-tuned model still has a better performance of multi-language translation.

Table 2: Change in bleu score of the original language of the model

	pre-adjust		direct-adjust		depth reinforcement-adjust	
	mean	sd	mean	sd	mean	sd
IWSLT17	29.53	0.48	26.72	0.32	32.04	0.61
IWSLT15	26.22	0.51	23.42	0.58	31.09	0.32
IWSLT13	21.11	0.68	16.84	0.26	23.11	0.44
CEEC	24.79	0.54	19.91	0.41	29.09	0.40

One disadvantage of direct fine-tuning is that there is a risk of overfitting. To explore this, the CEEC dataset was separated into a training set and a validation set with the ratio of 80:20. The paper also measured how loss values changed throughout training with both direct fine-tuning and deep reinforcement learning fine-tuning. The loss value changes obtained are depicted in Figure 4 where Direct means the direct fine-tuning method and Deep reinforcement means the method considered in this study. The figure depicts the loss value curves of the training as well as validation datasets of the CEEC dataset. With the direct fine-tuning, the loss value increases quickly in the validation set and becomes steady at about 4.75. This implies that direct fine-tuning needs proper checkpoint controls and early stopping to avoid overfitting. Even though the direct fine-tuning technique generates a lower validation loss than the deep reinforcement learning strategy, it has no better BLEU score on the test set. This means that direct fine-tuning can be easily overfitted and is not robust, particularly when resources are limited. In contrast, the local update method presented in this paper converges to a validation loss value of about 4.93. The deep reinforcement learning method is much more effective in solving the overfitting issue in translation models between language pairs compared to direct fine-tuning.

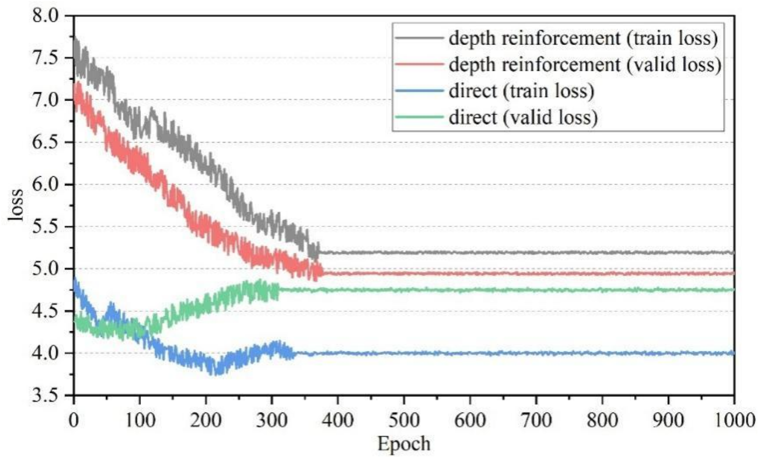

Figure 4: Change of the loss value of the fine-tuning process

3.4.3 Context-Aware Enhanced Translation Performance

The present subsection investigates how context-conscious TBRAE influences the performance of neural machine translation models in terms of their translation quality. Besides the fact that the BRAE was tested with the use of the baseline model, three versions of the TBRAE model were also tested, namely, TBRAE(cont) which only uses the topic context but does not consider word-topic constraints, TBRAE(ss) which has extra features depending on phrase similarity calculated on the source, and TBRAE(ts) which has extra features depending on phrase similarity calculated on the target.

The results of translation quality are given in Table 3, which is the outcome of five cross-validation tests performed on various datasets. The BLEU scores of the TBRAE model in this paper have increased by 5.23 to 7.76 points at all test sets over the baseline BRAE model. Also, the TBRAE variants demonstrate different levels of improvement. Particularly, TBRAE(cont) scored an increment of 0.99-3.43, TBRAE(ss) scored an increment of 2.07-4.08, and TBRAE(ts) scored an increase of 1.76-3.93. In comparison to the variants, the suggested framework with topic constraints and phrase similarity taken into consideration on the source and the target side causes a significant increase in performance. It implies that quality of neural machine translation is greatly improved when global context-awareness is incorporated.

Table 3: The translation quality of the model on different data sets

	BRAE		TBRAE(cont)		TBRAE(ss)		TBRAE(ts)		TBRAE	
	mean	sd	mean	sd	mean	sd	mean	sd	mean	sd
IWSLT17	24.06	0.45	28.81	0.34	29.66	0.22	29.97	0.29	31.73	0.49
IWSLT15	24.78	0.49	29.15	0.57	28.35	0.35	27.42	0.22	30.14	0.26
IWSLT13	25.21	0.57	29.83	0.46	27.89	0.39	29.04	0.46	32.97	0.56
CEEC	26.35	0.30	28.15	0.25	29.14	0.58	27.87	0.31	31.58	0.33

3.4.4 Translation sentence length correlation analysis

This paper further investigates the translation speed and translation quality of the final translation model with the autoregressive standard Transformer for different sentence lengths. Sentences in the [0,100] length interval in the test set of CEEC dataset are grouped according to the source sentence length, and each grouping corresponds to a sentence length interval of length 10. For translation speed, the time to complete the translation of the whole grouped data is calculated on each grouping for this paper's deep reinforcement learning-based context-aware translation model, the traditional translation model, and the bundle search size is set to 3. In turn calculate the speedup ratio between the two. For translation quality, bleu values are calculated on each grouping separately.

The figure 5 is showing the translation speed and quality depending on various sentence length, and this part is discussing the change in acceleration ratios across various length intervals. With the increase of sentence length, the acceleration ratio of the context-aware translation model, a deep reinforcement learning-based model, performs better than the conventional translation model. The acceleration ratio is 23.6 in the sentence length interval of (90,100), which means that the speed benefit of the proposed approach increases with the sentence length. It can be seen that this outcome confirms the hypothesis that the translation accuracy of the neural machine translation model is positively determined by the recursive self-encoder, which uses the contextual information, and the reward system built into deep reinforcement learning. In addition, the analysis of the BLEU scores of all the sentence length groups shows that the translation quality varies significantly less with shorter sentences, especially those in the [1,50] length range. Nevertheless, as the length of sentences becomes too long, the quality of translation starts decreasing at an accelerated pace, but its BLEU score remains high. Thus, it implies that the approach described in this paper is also resilient to the changes in the sentence lengths.

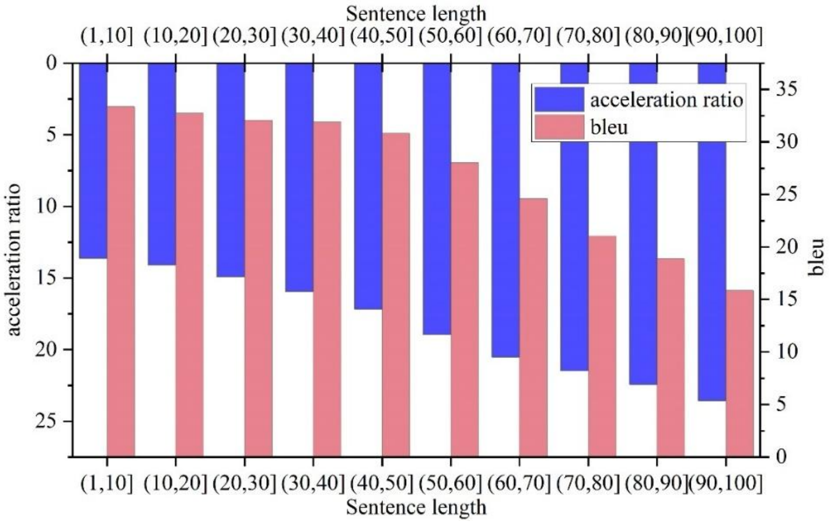

Figure 5: The translation speed and translation quality of different sentences

4 Conclusion

The research has the objective of improving the neural machine translation model accuracy of English translations by incorporating deep reinforcement learning and context-sensitive methods. This model generates English translations that are enriched with thematic contextual information by using phrase representations generated by a recursive self-encoder and thematic contextual representations. Deep reinforcement learning is subsequently used to maximize the parameters of the model by applying reward-based strategies and expectation gradient methods. The results of the experiment demonstrate the following main findings:

(1) With the rise in the level of cosine similarity of the model, the quality of translation first goes up, and then it goes down. The best translation quality is obtained at a threshold of 0.9.

(2) Deep reinforcement learning helps overcome the overfitting problem to a certain degree which causes an increment in the BLEU score by 2.00 to 4.87 across the four datasets after parameters optimization.

(3) Bilingual-restricted context-aware improves translation quality greatly and the BLEU score is improved by 5.23-7.76 compared with the baseline model. It also performs better than variant models such as TBRAE (cont) by 0.99-4.08 points.

(4) In case the length of sentences is in the range of (90,100], then the translation model with deep reinforcement learning and context-aware approach is 23.6 times quicker than the conventional translation model.

References

[1] Zhang, B., Xiong, D., Su, J., & Duan, H. (2017). A context-aware recurrent encoder for neural machine translation. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 25(12), 2424-2432.

[2] Jiang, C., & He, Y. (2025). Construction and Evaluation of Context Aware Machine Translation System. Procedia Computer Science, 261, 529-537.

[3] Wu, X., Xia, Y., Zhu, J., Wu, L., Xie, S., & Qin, T. (2022). A study of BERT for context-aware neural machine translation. Machine Learning, 111(3), 917-935.

[4] Guo, D. (2025). Deep learning-driven context-aware English translation for ambiguous sentences. International Journal of Information and Communication Technology, 26(15), 41-56.

[5] Gete, H., Etchegoyhen, T., & Labaka, G. (2023, June). What Works When in Context-aware Neural Machine Translation?. In Proceedings of the 24th Annual Conference of the European Association for Machine Translation (pp. 147-156).

[6] Rikters, M., Ri, R., Li, T., & Nakazawa, T. (2021). Japanese–English conversation parallel corpus for promoting context-aware machine translation research. Journal of Natural Language Processing, 28(2), 380-403.

[7] Vu, H. H., Kamigaito, H., & Watanabe, T. (2024). Context-aware machine translation with source coreference explanation. Transactions of the Association for Computational Linguistics, 12, 856-874.

[8] Su, J., Xiong, D., Liu, Y., Han, X., Lin, H., Yao, J., & Zhang, M. (2015, July). A context-aware topic model for statistical machine translation. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers) (pp. 229-238).

[9] Hu, X., Yin, X., & Wan, X. (2023, December). Exploring context-aware evaluation metrics for machine translation. In Findings of the Association for Computational Linguistics: EMNLP 2023 (pp. 15291-15298).

[10] Chen, L., Li, J., Gong, Z., Zhang, M., & Zhou, G. (2022). One type context is not enough: Global context-aware neural machine translation. ACM Transactions on Asian and Low-Resource Language Information Processing, 21(6), 1-14.

[11] Menezes, M., Farajian, A., Moniz, H., & Graça, J. (2025). Context is everything: a context-aware annotation typology for dialogue translation quality assessment. Tradumàtica tecnologies de la traducció, (23), 383-416

[12] Chauhan, S., Daniel, P., Saxena, S., & Sharma, A. (2022). Fully unsupervised machine translation using context-aware word translation and denoising autoencoder. Applied Artificial Intelligence, 36(1), 2031817.

[13] Dinarelli, M., Niaouri, D., Lopez, F., Gonzalez-Saez, G., Nakhlé, M., Esperança-Rodier, E., ... & Ballier, N. (2024). Context-Aware Neural Machine Translation Models Analysis And Evaluation Through Attention. Traitement Automatique des Langues, 64(3), 67-91.

[14] Sun, K., Qian, T., Chen, X., & Zhong, M. (2021). Context-aware seq2seq translation model for sequential recommendation. Information Sciences, 581, 60-72.

[15] Ladosz, P., Weng, L., Kim, M., & Oh, H. (2022). Exploration in deep reinforcement learning: A survey. Information Fusion, 85, 1-22.

[16] Heuillet, A., Couthouis, F., & Díaz-Rodríguez, N. (2021). Explainability in deep reinforcement learning. Knowledge-Based Systems, 214, 106685.

[17] Wang, J., Meng, F., & Zhou, J. (2026). Deeptrans: Deep reasoning translation via

reinforcement learning. Transactions of the Association for Computational Linguistics, 14, 47-63.

[18] Guo, Q. (2025). Deep Reinforcement Learning Method for Optimizing English Translation Algorithms. International Journal of High Speed Electronics and Systems, 2540568.