Research Article

Yijun Wu* and Yonghong Qin

Machine translation of English speech: Comparison of multiple algorithms

https://doi.org/10.1515/jisys-2022-0005
received August 26, 2021; accepted November 15, 2021

Abstract: In order to improve the efficiency of the English translation, machine translation is gradually and widely used. This study briefly introduces the neural network algorithm for speech recognition. Long short-term memory (LSTM), instead of traditional recurrent neural network (RNN), was used as the encoding algorithm for the encoder, and RNN as the decoding algorithm for the decoder. Then, simulation experiments were carried out on the machine translation algorithm, and it was compared with two other machine translation algorithms. The results showed that the back-propagation (BP) neural network had a lower word error rate and spent less recognition time than artificial recognition in recognizing the speech; the LSTM–RNN algorithm had a lower word error rate than BP–RNN and RNN–RNN algorithms in recognizing the test samples. In the actual speech translation test, as the length of speech increased, the LSTM–RNN algorithm had the least changes in the translation score and word error rate, and it had the highest translation score and the lowest word error rate under the same speech length.

Keywords: English, machine translation, recurrent neural network, long short-term memory

1 Introduction

Globalization is a major trend in modern society. With the development of the economy, the division of labor in society is becoming ever more detailed, and cooperation and exchange between different countries are also increasing [1]. In communication, language communication is crucial, and the use of a conventional language that can be mutually understood can avoid misunderstandings and improve the efficiency of the division of labor. English is one of the most widely used common languages, but for nonnative English speakers, the cost of learning a new language is high, and it is difficult to reach the level of free communication [2]. In the wave of globalization, it is sufficient for face-to-face daily communication, but on formal occasions and when a large amount of information needs to be exchanged, it is difficult for a single human interpreter to meet the increasing demand for language translation. For example, translators should focus during simultaneous interpretation and often cannot work for long hours; thus, a translation tool is needed to replace human translation [3]. Machine translation uses computers and Chinese–English thesauri to perform batch translation, but it is too rigid. The emergence of intelligent algorithms has effectively contributed to the efficiency and quality of machine translation. Ashengo et al. [4] proposed a new method combining contextual based machine translation with recurrent neural network machine translation for translating English texts into Amharic texts, evaluated it by taking the New Testament Bible as a corpus, and verified the performance of the model. Lee et al. [5] used character-level convolutional networks to perform machine translation and found that the character-level convolutional network encoders

* **Corresponding author: Yijun Wu,** Department of Foreign Languages, Xi’an Jiaotong University City College, Xi’an, Shaanxi 710018, China, e-mail: y6w8yi@163.com
Yonghong Qin: School of Electrical Engineering, Southwest Jiaotong University, Chengdu, Sichuan 610031, China

Open Access. © 2022 Yijun Wu and Yonghong Qin, published by De Gruyter. This work is licensed under the Creative Commons Attribution 4.0 International License.

significantly outperformed the subword-level encoders in multilingual experiments. Koul and Manvi [6] put forward a model that could translate Sanskrit into English with a recurrent neural network and verified the effectiveness of the model with experiments. In the previous studies on translation of languages, some researchers choose to translate languages with words as the translation unit. Some use words as the translation unit, but construct the character arrangement as a two-dimensional flat image and use convolutional networks to translate the text. Some others focus on the training method of intelligent algorithms for the optimization of the translation model. This study took the advantage of recurrent neural networks (RNNs) in processing data with sequential meanings to translate English text. Compared with the translation model used in the previous studies, RNN fully considers the influence of word order on the translation. To further reduce the interference of invalid words on the translation, long short-term memory (LSTM) was used to improve the translation model. LSTM is also a RNN algorithm, but the "forgetting" mechanism introduced into LSTM could eliminate the influence of invalid words and highlight effective information. This study used the back-propagation (BP) neural network to recognize speech as text, also used LSTM as the encoding algorithm in the encoder of the translation algorithm and RNN as the decoding algorithm in the decoder of the translation algorithm, and compared the LSTM–RNN algorithm with two translation algorithms, BP–RNN and RNN–RNN algorithms, in simulation experiments.

2 Recognition of speech

Machine translation uses the performance of computers to achieve efficient translation of English texts; however, computers without vision and hearing need input text before translating English texts according to characters [7]. In practical applications, it is inefficient to use manual input when querying and translating English words or phrases, while direct input by voice is relatively convenient, and translation by voice means the possibility of simultaneous interpretation by computer. Before translating the English language input by voice, it is necessary to first recognize the voice and convert the audio into text characters.

In this study, a BP neural network was used for character recognition of speech. The reason for selecting the BP neural network is that the BP neural network is the most basic neural network and has relatively high generalizability. The basic process of speech recognition is shown in Figure 1. First, features are extracted from the collected English speech, and Mel-scale Frequency Cepstral Coefficients are selected to extract features of speech samples in this study. Then, the BP neural network is trained with the extracted features. In the training process, the samples whose features have been extracted are input into the BP neural network. The multilayer forward calculation of the extracted features is performed in the hidden layer using the activation function, and the results obtained after the layer-by-layer calculation are compared with the corresponding results of the training samples. The hyperparameters in the hidden layer are adjusted in the reverse direction according to the gap between the results [8]. Then, the layer-by-layer forward calculation is performed again, the calculated results are compared with the actual results, and the parameters are adjusted in reverse. The above steps are repeated until the gap between the calculated results and the actual results is reduced to the set threshold value. The speech samples for testing are input into the trained neural network model after feature extraction, and the recognition results are output after calculation. The gradient descent method is used for reversely adjusting parameters. The iterative formula for adjusting parameters is:

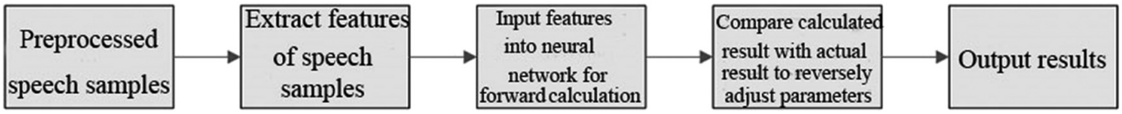
Preprocessed speech samples
Extract features of speech samples
Input features into neural network for forward calculation
Compare calculated result with actual result to reversely adjust parameters
Output results

Figure 1: The basic flow of speech recognition.

$$\begin{cases} \omega_{i+1} = \omega_i - \alpha \dfrac{\mathrm{d}L}{\mathrm{d}\omega_i}, \\ b_{i+1} = b_i - \alpha \dfrac{\mathrm{d}L}{\mathrm{d}b_i}, \end{cases} \tag{1}$$

where ω_i and b_i are the weight and bias term, ω_{i+1} and b_{i+1} are the weight and bias term after one iteration, L is a deviation, and α is a learning rate. Besides the application in the training of the BP neural network, equation (1) is also used in the training of RNN and LSTM.

3 Machine translation of English after recognition

3.1 Machine translation algorithm based on deep learning

For the recognized speech text, the traditional machine translation is to translate the speech text word-by-word with the help of the word bank of Chinese–English mutual translation [9]. Although this machine translation method is simple and fast, and the general meaning of the translated text can be guaranteed in conventional translation, the effect is poor when translating long English sentences. The first reason is that word-for-word translation often leads to grammatical confusion or even the complete opposite semantics because of grammatical differences between Chinese and English. The second reason is that some of the auxiliary words that do not have specific meanings in English statements are also translated, affecting the coherence of the translation [10].

The emergence of deep learning-based intelligent algorithms has remedied the above-mentioned defects of machine translation. The basic structure of intelligent algorithm-based machine translation methods is shown in Figure 2. The overall structure is divided into an encoder and a decoder. The specific algorithms used in the encoder and decoder are deep learning algorithms, including BP neural network, convolutional neural network, and so on. When the English text is translated, the English text is first converted into an encoding vector by the encoder, and then the encoding vector is converted to Chinese by the decoder.

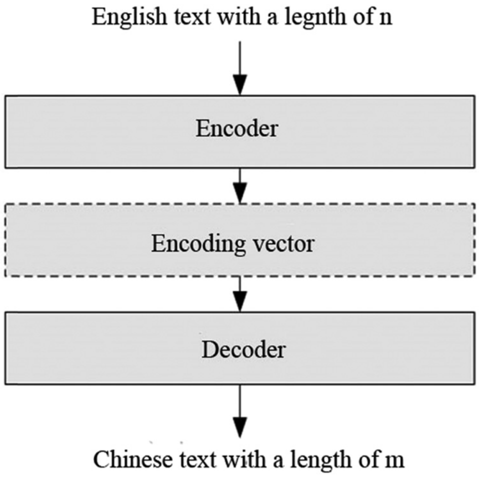
English text with a legnth of n
Encoder
Encoding vector
Decoder
Chinese text with a length of m

Figure 2: Basic structure of deep learning-based machine translation algorithm.

3.2 Machine translation algorithm with a LSTM encoder and a RNN decoder

A conventional BP neural network [11] is sufficient to convert a text with a length of n into an encoded vector string. The purpose of converting the text into a coded vector by the encoder in this study is to realize machine translation of the text in subsequent decoding; however, the meaning of the words in the utterance will be affected by the sequence, and the conventional BP network does not consider the influence of the word sequence when encoding the text. In the training and use of the RNN algorithm, the results at the current moment are influenced by the results at the previous moment, which fits well with the property of language that word sequences influence the meaning of words; therefore, the RNN algorithm is used for encoding and decoding machine translation.

However, the encoder faces the problem of gradient explosion when encoding English using the RNN algorithm, which makes the algorithm inefficient and less accurate in training and use. To remedy this deficiency of encoder in machine translation algorithm, this study adopts LSTM to encode English instead of the RNN algorithm. The LSTM algorithm is a kind of the RNN algorithm after introducing forgetting mechanism [12], and the input gate, forget gate, and output gate are the cores of the LSTM algorithm. The forget gate unit can round off the unimportant parts of the text when coding and converting it, highlighting the key points and reducing the number of operations while enhancing the accuracy. The forward calculation formula for encoding English text with LSTM is:

$$\begin{cases} \varepsilon = \{\varepsilon_0, \varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n\}, \\ x_t = e(\varepsilon_t), \\ f_t = \sigma(b_f + U_f x_t + W_f h_{t-1}), \\ s_t = f_t s_{t-1} + g_t \sigma(b + Ux_t + Wh_{t-1}), \\ g_t = \sigma(b_g + U_g x_t + W_g h_{t-1}), \\ h_t = \tanh(s_t) q_t, \\ q_t = \sigma(b_q + U_q x_t + W_q h_{t-1}), \end{cases} \tag{2}$$

where ε is the English text to be translated; ε_n is the $n + 1$th word in the English text; x_t is the word vector after processing ε_t with the vector transformation function $e(\cdot)$; t is the time step (every time step corresponds to the input moment of the corresponding sequence of words); h_{t-1}, h_t are hidden states of word vectors whose word sequences are $t - 1$ and t [13]; f_t is the output of the forget gate; b_f, U_f, and W_f are the bias term, input term weight, and forget gate weight in the forget gate, respectively; s_t is the output of the recurrent gate; b, U, and W are the bias term, input term weight, and recurrent gate weight in the recurrent gate, respectively; g_t is the external input gate unit; b_g, U_g, and W_g are the bias term, input term weight, and input gate weight in the input gate, respectively; q_t is the output gate unit; and b_q, U_q, and W_q are the output gate bias term, input term weight, and output gate weight, respectively.

After the forward operation of equation (2) in the encoder, every word ε_t in the English sentence gets the corresponding coding vector a_t, and the coding vector of every word is influenced by the previous word vector, which ensures that the coding vector of the whole sentence contains the influence of word order on the meaning of words. After that, the coding vector needs to be decoded to obtain the corresponding Chinese translation, and the decoding is carried out by the decoder. In order to guarantee the influence of word order on word meaning, the RNN algorithm is used to decode the coding vector, and the forward calculation formula in the hidden layer is

$$\begin{cases} y_{t-1} = e(Z_{t-1}), \\ z_t = \tanh(\omega z_{t1-} + Uy_{t-1} + b), \\ \hat{y}_t = \mathrm{softmax}(d + Vz_t), \\ Z_t = \arg\max \hat{y}_t, \end{cases} \tag{3}$$

where ω and U are the hidden state at time $t - 1$ and the weight matrix of the input ε_t word vector at time t; b is the bias term; Z_t is a word whose order is t in the translated sentence; y_{t-1} is the vector of a word whose order is $t - 1$ in the translated sentence; d is the bias term; V is the weight matrix; $\hat{y}_t$ is the probability distribution of different translated characters [14]; and z_t is the hidden state in the decoder.

4 Experimental analysis

4.1 Experimental environment

The experiments were conducted on a lab server [15] with the following configuration: Windows 7 system, I7 processor, and 16G memory.

4.2 Experimental data

In this study, the English data set was crawled from the Internet by a crawler software. The pronouncers involved in this data set cover most age groups from 12 to 70 years old. A total of 10,000 sentences with clean and clear pronunciation were selected, among which, 9,000 sentences were randomly selected as the training samples, and the remaining 1,000 sentences were used as the test sample. The experimenter read the sentences aloud and collected the speech feature parameters of the sentences. The sampling rate was set as 16 kHz, and 16-bit encoding was used. Some of the sentences that were read aloud are as follows.

(1) How are you?;
(2) How can I get to the airport, please;
(3) What’s the weather like today.

4.3 Experimental projects

4.3.1 Performance testing of speech recognition

The BP neural network for speech recognition was first trained using a training set. There were five hidden layers containing the rectified linear unit activation function. There were 1,024 nodes in every hidden layer. The maximum number of iterations was 500. The training set was for testing, and the testing results were compared with the results of artificial recognition.

4.3.2 Performance testing of machine translation algorithms

The LSTM used as the encoder of the machine translation algorithm had four hidden layers, and the number of nodes in every layer was 1,024; the RNN used as the decoder contained two hidden layers, and the hidden layer size was the same as that of the LSTM. To verify the effectiveness of the machine translation algorithm proposed in this study, it was compared with two other NMT algorithms. The difference between the two NMT algorithms and the machine translation algorithm proposed in this study was that they used BP neural network and RNN as the encoder respectively and RNN as the decoder. After training by the training set, the three machine translation algorithms were tested by the test set. The three machine translation algorithms in the comparative experiment all had a basic structure of “encoder + decoder,” and the decoder was RNN. The difference between the three algorithms was the intelligent algorithm, which was the BP neural network, RNN, and LSTM, respectively.

4.3.3 Real performance testing of machine translation algorithms for English speech

The main objective of this test was to examine the generalization performance of the machine translation algorithm in a real application environment after combining speech recognition and machine translation. In this test, ten volunteers were randomly invited. They read aloud English texts containing 100, 200, 300, 400, and 500 words. The audios were collected using a radio device and input into the trained machine translation algorithm to translate the volunteers' English speech.

4.4 Evaluation criteria

In this study, the translation was evaluated by the word error rate, and the calculation formula is

$$\mathrm{WER} = \frac{X + Y + Z}{P} * 100\%, \tag{4}$$

where X is the number of incorrect words substituted; Y is the number of incorrect words deleted; Z is the number of incorrect words inserted; and P is the number of all words in the test set. In addition to the word error rate, the evaluation criteria for the practical application of the machine translation algorithm also include the overall translation level of the sentence. Therefore, in this study, ten professional translators were invited to evaluate the translation, and the translation was scored according to the expression and grammatical structure. The total score was 100 points. The average score was taken as the final result.

4.5 Experimental results

As shown in Table 1, the word error rate of English speech recognition by artificial recognition was 7.53%, and the recognition time was 34 min; the word error rate of English speech recognition by BP neural network was 1.54%, and the recognition time was 8 min. The comparison of the speech recognition effect of the two methods showed that BP neural network had a lower word error rate and spent less time than artificial recognition. The reason for the above result was that computers were more efficient in terms of computational efficiency, and when faced with a large number of speech sounds in the test set, people could not maintain their attention for a long time, resulting in a higher word error rate and longer recognition time; however, BP neural network used computers to recognize speech sounds, which was not only computationally efficient, but also does not have the disadvantage of inattentiveness.

Table 1: The word error rate and recognition time of the BP neural network and the artificial recognition method

	Word error rate (%)	Recognition time (s)
Artificial recognition	7.53	34
BP neural network	1.54	8

It was seen from Table 2 that the translation word error rate of the machine translation algorithm with BP neural network for encoder and RNN for decoder was 25.4%; the translation word error rate of the machine translation algorithm with RNN for both encoder and decoder was 18.7%; the translation word error rate of the machine translation algorithm with LSTM for encoder and RNN for decoder was 3.2%. The BP–RNN algorithm had the highest error rate of translated words, the RNN–RNN algorithm was the second, and the LSTM–RNN algorithm had the lowest error rate of translated words.

Table 2: The word error rate of the three machine translation algorithms for the translation of English text

	BP–RNN	RNN–RNN	LSTM–RNN
Word error rate (%)	25.4	18.7	3.2

The two test results shown above were obtained from testing in the laboratory using a predetermined set of samples after training. All three machine translation algorithms utilized the neural network algorithm. The neural network, when learning with a limited number of training samples, will appear to have increasingly better performance on the samples, but once it steps out of the pre-prepared samples and tests on out-of-sample data, the performance decreases instead, i.e., it falls into an overfitting state, resulting in no generalization performance of the whole algorithm. In To test the generalization performance of the three machine translation algorithms after training, i.e., the actual application performance, this study selected ten volunteers to read aloud English scripts with different word counts and translated the speech read by the volunteers using the three machine translation algorithms that have been trained respectively. The word error rate and translation score of the final translation are shown in Table 3. As the length of the English speech to be translated increased, the scores of the translations obtained by the three translation algorithms decreased, and the BP–RNN algorithm had the largest decrease, the RNN–RNN algorithm had the second largest decrease, and the LSTM–RNN algorithm had the smallest decrease. Under the same speech length, the LSTM–RNN algorithm had the highest translation score, followed by the RNN–RNN algorithm and the BP–RNN algorithm.

Table 3: The actual application performance of the three machine translation algorithms for English speech

Word count	Score of translation (hundred-mark system)			Word error rate (%)		
	BP–RNN	RNN–RNN	LSTM–RNN	BP–RNN	RNN–RNN	LSTM–RNN
100	68.9	88.7	97.8	30.1	20.3	5.2
200	65.1	86.3	96.7	32.6	22.3	5.6
300	60.3	83.1	95.8	38.6	24.8	6.1
400	56.4	80.1	95.1	42.1	26.9	6.3
500	50.15	78.7	94.7	47.3	29.2	6.5

It was seen from Table 3 that the word error rate of the translations obtained by all three translation algorithms increased as the length of English speech increased; the BP–RNN algorithm had the largest increase, the RNN–RNN algorithm had the second largest increase, and the LSTM–RNN algorithm nearly had no increase. Under the same speech length, the BP–RNN algorithm had the highest word error rate, followed by the RNN–RNN algorithm and the LSTM–RNN algorithm.

5 Discussion

Although relying solely on manual translation of English is more secure in terms of accuracy and more flexible in translating slang or colloquialisms, manual translation has limited efficiency and is difficult to meet the needs of long text translation. Also in the case of simultaneous interpretation, the manual approach is hardly sustainable. The emergence of intelligent algorithms has enabled machine translation to be more widely used. The basic structure of the machine translation algorithm includes an encoder, which converts the text to be translated into a string of coded vectors, and a decoder, which converts the coded vectors into translated text. The algorithms used by the encoder and decoder are intelligent algorithms. As the meaning of words in a text is often affected by word order, RNN, which can consider the effect of order, is usually used as the algorithm for the encoder and decoder. In this study, to reduce the influence of invalid words on translation, LSTM was used as the algorithm of the encoder. Then, three machine translation algorithms, BP–RNN, RNN–RNN, and LSTM–RNN algorithms, were compared, and the results have been shown above. The difference between the three machine translation algorithms only lied in the algorithm used in the encoder, but the steps were nearly the same when they translated the English text. Taking "I love summer" as an example, its Chinese translation is "我爱夏天." When translating this English sentence, first, we encode "I love summer" in the encoder and split the sentence into a source sequence of "BOS:0, I:1, love:2, summer:3". "BOS" means beginning of sequence, and then the individual elements in the source sequence are converted into vectors to obtain a sequence of word vectors. The corresponding neural network algorithm is used in the encoder to convert the vector sequence into the context code. The context code of the word vector sequence is input to the decoder, and the RNN in the decoder is used to decode the context code. The probability distribution of the corresponding translated words is calculated. The word with the highest probability is selected to form the translated text. Finally, "我爱夏天" was obtained. In simple terms, the translated text corresponding to the context code is guessed by the decoder based on the training experience.

The performance of the BP neural network for speech recognition was examined first. The BP neural network was more accurate and faster compared with manual recognition. Then, it is the focus of this study, i.e., the effect of translating English texts. First, the word error rate of the three machine translation algorithms was compared. It was found that the LSTM–RNN algorithm had the smallest word error rate. The comparison results of recognizing and translating speech with different number of words showed that as the words of the speech to be translated increased, the translation score and word error rate of the LSTM–RNN algorithm were relatively stable; the translation score of the LSTM–RNN algorithm was higher than the other two machine translation algorithms; the word error rate of the LSTM–RNN algorithm was lower than the other two machine translation algorithms. The reason for the above results is as follows. The BP neural network algorithm encoded English directly word by word. Although it also mined the hidden laws, it could not accurately describe the influence of word order on word meaning, which led to an increase in word error rate. The RNN algorithm took into account the influence of the previous moment in the forward calculation, i.e., the influence of the previous word in this study; therefore, the word error rate was lower. The LSTM algorithm, as a variant of the RNN algorithm, could also summarize the influence of word order. The introduction of forget gate, input gate, and output gate units in the LSTM algorithm filtered the unimportant words to improve the accuracy.

6 Conclusion

This study briefly introduces the neural network algorithm for speech recognition and adopts LSTM instead of traditional RNN as the encoding algorithm for the encoder and RNN as the decoding algorithm for the decoder. Simulation experiments were carried out on the machine translation algorithm, and it was compared with two other machine translation algorithms. The results are as follows. (1) BP neural network had a lower word error rate and less recognition time compared with the artificial recognition method. (2) The LSTM–RNN algorithm had the lowest word error rate for English speech recognition results, the RNN–RNN-based machine translation algorithm had a higher word error rate, and the BP–RNN-based machine translation algorithm had the highest word error rate. (3) In practical speech translation applications, the translation scores obtained by the three translation algorithms decreased, and the word error rate increased with the increase of speech length; the LSTM–RNN algorithm had the smallest change; the LSTM–RNN algorithm had the lowest word error rate and the highest translation score under the same speech length.

Conflict of interest: Authors state no conflict of interest.

References

[1] Bayatli S, Kurnaz S, Ali A, Washington JN, Tyers FM. Unsupervised weighting of transfer rules in rule-based machine translation using maximum-entropy approach. J Inf Sci Eng. 2020;36(2):309–22.

[2] Ren Q, Su Y, Wu N. Research on Mongolian-Chinese machine translation based on the end-to-end neural network. Int J Wavel Multi. 2020;18(1):46–59.

[3] Herbig N, Pal S, Vela M, Krüger A, van Genabith J. Multi-modal indicators for estimating perceived cognitive load in post-editing of machine translation. Mach Transl. 2019;33(1–2):91–115.

[4] Ashengo YA, Aga RT, Abebe SL. Context based machine translation with recurrent neural network for English–Amharic translation. Mach Transl. 2021;35:19–36.

[5] Lee J, Cho K, Hofmann T. Fully character-level neural machine translation without explicit segmentation. Trans Assoc Comput Linguist. 2017;5:365–78.

[6] Koul N, Manvi SS. A proposed model for neural machine translation of Sanskrit into English. Int J Inform Technol. 2021;13(1):375–81.

[7] Chatzikoumi E. How to evaluate machine translation: a review of automated and human metrics. Nat Lang Eng. 2019;26(2):1–25.

[8] Soto X, Perez-de-Viñaspre O, Labaka G, Oronoz M. Neural machine translation of clinical texts between long distance languages. J Am Med Inf Assoc. 2019;26(12):1478–87.

[9] Niyazbek M, Talp K, Sun J. The development and construction of bilingual machine translation auxiliary tool between Chinese and Kazakh languages. IOP Conf Ser Earth Environ Sci. 2021;687(1):012205 (5pp).

[10] Bywood L, Georgakopoulou P, Etchegoyhen T. Embracing the threat: machine translation as a solution for subtitling. Persp Stud Transl. 2017;25(3):1–17.

[11] Xiao Q, Chang X, Zhang X, Liu X. Multi-information spatial-temporal LSTM fusion continuous sign language neural machine translation. IEEE Access. 2020;8:216718–28.

[12] Rozovskaya A, Dan R. Grammatical error correction: machine translation and classifiers. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics; 2016. p. 2205–15.

[13] Plaza-Lara C. How does machine translation and post-editing affect project management? An interdisciplinary approach. Hikma. 2020;19:163–82.

[14] Gritsay I, Vodyanitskaya L. Pedagogical technologies of machine translation skills forming on the example of bachelor students specializing in mechatronics and robotics at Don State Technical University. E3S Web Conf. 2021;273:12140.

[15] Li S. Research on the external communication of Chinese excellent traditional culture from the perspective of machine translation. J Phys Conf Ser. 2021;1744:032019.