

Research Article

Artificial Intelligence Machine Translation Based on Fuzzy Algorithm

Zhimin Li 

Luohe Medical College, Luohe 462000, Henan, China

Correspondence should be addressed to Zhimin Li; 201701015410@stu.zjsru.edu.cn

Received 2 September 2021; Revised 15 October 2021; Accepted 27 October 2021; Published 19 November 2021

Academic Editor: Sang-Bing Tsai

Copyright © 2021 Zhimin Li. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In order to study machine translation more in-depth, it is particularly important for the research of artificial intelligence with fuzzy algorithms to convert an unfamiliar language into a mature language. The neural network translation model has been developed in recent years and has achieved rich research results. Aiming at the current lack of accuracy of neural machine translation (NMT), which may cause ambiguity, this paper takes English machine translation as an example and proposes an artificial intelligence machine translation optimization model based on fuzzy theory. On the basis of NMT model translation, first the semantics of English machine translation is classified, a semantic selection model is built, then the analytic hierarchy process is used to determine the semantic order of English machine translation, and the corresponding fault-tolerant operation is carried out to the error-prone errors, weight the semantics, and introduce the fuzzy theory to arrange the English semantics of English machine translation. Finally, the performance of the model is analyzed through specific application experiments. The results show that the accuracy of the machine translation selection permutation model is improved by nearly 4.5% and can reach more than 90% compared with other models, and the timeliness is better than other models, which is improved by nearly 15%, which has obvious advantages.

1. Introduction

Machine translation is one of the commonly used services on the Internet, and it is also an important research field in language processing and artificial intelligence. Such as Google Translator, Baidu Translator, Microsoft Bing Translator, etc., all provide online translation services between multiple languages. Although there is still a large gap between the quality of machine translations and professional translators, in some scenarios where the quality of translations is not high, or in the task of translation in a specific field, machine translation has a clear advantage in translation speed and still it is widely used.

At present, there are some urgent problems in NMT. NMT is facing a more severe problem of translation of unregistered words and long sentences. Therefore, achieving the unlimited dictionary size or being able to efficiently deal with the problem of unregistered words and also being able to effectively translate longer sentences are the basic

problems that NMT needs to solve. Aiming at the problem of attention mechanism, from the perspective of semantic ordering, a semantic ordering model based on fuzzy theory is proposed to address the problems of poor accuracy and ambiguity of current English machine translation.

Early perceptrons, due to their simple structure, were unable to handle linear inseparable problems, which led to a long period of low tide in this study. After the 1980s, backpropagation (BP) was introduced to multilayer perceptron (MLP), also called feed forward neural network (FNN). Since then, with the promotion of some related researchers, neural networks have regained attention. In 2006, Hinton et al. solved the problem of neural network training through the layer-by-layer pretraining method. Later, due to the increase in computing power, neural networks have received widespread attention from academia and industry. Results: Sennrich et al.'s team [1] studied the use of monolingual data in NMT in order to improve the fluency of phrase-based statistical machine translation. Hill

et al.'s team [2] made the first comprehensive analysis of word embedding attributes learned by a bilingual text-trained NMT model. The word notation is better than established algorithms such as skip-gram and CBOW. Su et al.'s team [3] believes that although NMT has made great progress, learning the best model parameters for long parallel sentences and making full use of different contexts can make NMT more accurate. Shaimaa Marzouk's and Hansen-Schirra's team [4] studied the application of controlled language (CL) not only as an effective pre-editing technique for improving the output of machine translation (MT) but also for NMT.

In this article, from the perspective of semantic ordering, the accuracy of NMT is poor, which may cause ambiguity and other issues. This article proposes an English semantic ordering model based on fuzzy theory. For different English semantics, the ordering is based on its characteristics. The proposed model is validated. Data are often organized through patterns. Access is also obtained through the action mode. The accurate ordering of semantics helps to better understand complex concepts which is of great significance.

2. Proposed Method

2.1. NMT Model. During the modeling process, Calci Brunner and Brunsham repropose a neural network-based translation method, showing great application potential [5, 6]. Then, the researchers constructed a machine translation model based on the neural network [7, 8]. In many rule-based machine translation systems, linguists assist in writing a series of grammatical rules about the source language and target language, as well as the conversion rules for converting source language data into target language data. However, it is very expensive, time-consuming, and error-prone to produce these rules using all manual labor. One solution is to use past historical translation results as a resource library, where the source language text and its corresponding target language translation are used as examples and try to extract appropriate rules from it. One of the methods is to manually mark the source text and the target language translation to show the association. Sato language and Nagao developed a system that uses a "plane dependency tree" to represent source and target language characters. This relational tree data structure is a form of efficient computer recognition. Two levels are usually used to represent the relationship between the source language and the target language: the first-level depends on the surface form of the text (such as the order of characters and words), which is used for the analysis of the source language and the generation of the target language; the second level depends on the semantic association between words which is used for the conversion from the source language to the target language. This kind of machine translation system takes advantage of the case base on the basis of rule-based machine translation. They can be used for machine translation and other natural language processing tasks [9, 10]. The Google translation model is shown in Figure 1. The model inputs $A, B,$ and C sequentially generate outputs $W, X, Y,$ and Z under the input conditions, where $\langle \text{EOS} \rangle$ is the artificial end of sentence.

The other is called the encoder-decoder model, as shown in Figure 2.

The encoder-decoder [11] model consists of three parts: input x , hidden state h , and output y . The encoder reads the input $x = (x_1, x_2, \dots, x_I)$ and encodes it as a hidden state $h = (h_1, h_2, \dots, h_I)$. When using a recurrent neural network (RNN),

$$\begin{aligned} h_i &= f(x_i, h_{i-1}), \\ c &= q(\{h_1, \dots, h_I\}), \end{aligned} \quad (1)$$

where c is the source language sentence representation, and f and q are nonlinear functions.

$$p(y) = \prod_{t=1}^T p(y_t | \{y_1, \dots, y_{t-1}\}, c), \quad (2)$$

where $y = (y_1, y_2, \dots, y_T)$. When using a recurrent neural network,

$$p(y_t | \{y_1, \dots, y_{t-1}\}, c) = g(y_{t-1}, s_t, c),$$

$$L(\theta) = \max_{\theta} \frac{1}{N} \sum_{n=1}^N \log p_{\theta}(y_n | x_n), \quad (3)$$

where θ is the parameter of the model and (x_n, y_n) is a two-sentence pair.

2.2. Neural Network Overview

2.2.1. Neural Network. A neural network (MPL) is composed of many neurons [12, 13]. One neuron passes the inner product of the vector and weight vector input to the next neuron through a nonlinear transfer function [14, 15]. Neurons here can also be called perceptrons, which are the basic building blocks of neural networks. Figure 3 shows the basic structure of a neuron in a neural network [16, 17].

$a_1, a_2,$ and a_3 represent the inputs, and f is a nonlinear activation function that passes the resulting z to the next neuron [18, 19]. The perceptron can implement simple binary classification problems and can also fit linear functions. Any linear classification or linear regression problem can be used to solve it. The neural network can not only implement the above functions but also achieve more complex problems such as multiclassification and clustering. As the number of network layers increases, the number of hidden layers increases. The number of nodes (neurons) in the hidden layer does not have a fixed number, but as the number increases, the nonlinearity of the neural network will become more significant, and the robustness of the neural network will be stronger. When designing a network, the number of nodes is usually controlled to be 1.2 to 1.5 times the number of input nodes. A neural network with a hidden layer greater than one layer is called a deep neural network. A deep neural network can extract deeper features of an image, which is the main reason why the classification effect is higher than other methods [20].

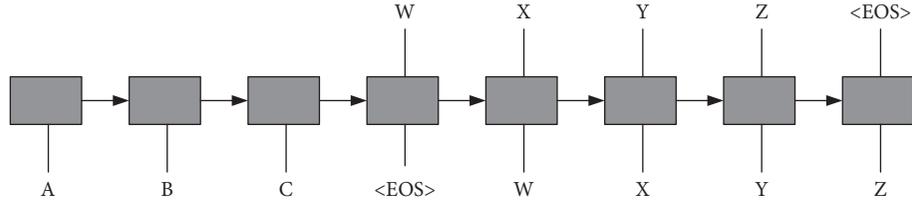


FIGURE 1: End-to-end model.

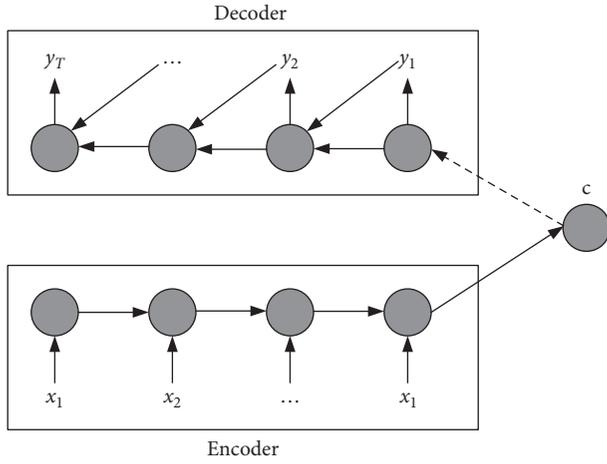


FIGURE 2: Encoder-decoder model.

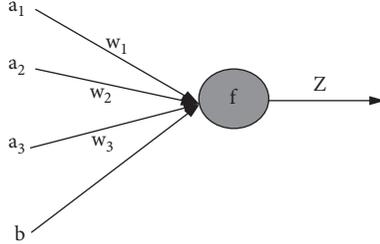


FIGURE 3: The basic structure of neurons in neural networks.

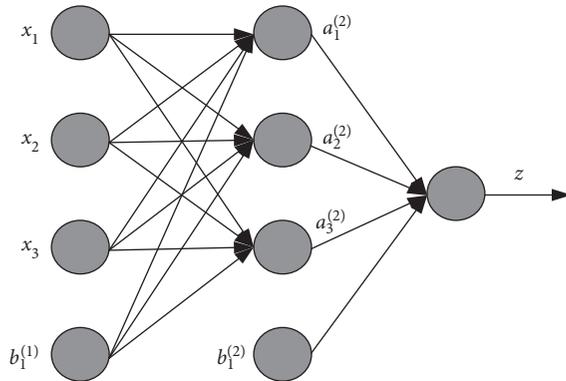


FIGURE 4: The basic structure of a two-layer neural network.

Figure 4 shows the structure of a two-layer neural network.

The input of the neural network is represented by x_1 , x_2 , and x_3 , the output z , the weight w , and the deviation of each

layer is b , and the expression of z is related to the hidden layer of the intermediate layer, that is,

$$z = f(a_1 * w_1 + a_2 * w_2 + a_3 * w_3 + b). \quad (4)$$

The value of the hidden layer is related to the input value of the previous layer. Assume that the bias of the input layer is $b_1^{(1)}$ and the bias of the hidden layer is $b_1^{(2)}$, and each neuron $a_i^{(2)}$ can be expressed as

$$a_i^{(2)} = f(w_{i1}^1 x_1 + w_{i2}^1 x_2 + w_{i3}^1 x_3 + b_1^{(1)}). \quad (5)$$

Among them, $i = 1, 2, 3$, so the output z of this neural network can be expressed as

$$z = f(w_{11}^{(2)} a_1^{(2)} + w_{12}^{(2)} a_2^{(2)} + w_{13}^{(2)} a_3^{(2)} + b_1^{(2)}), \quad (6)$$

which is

$$z_{ik} = f\left(\sum w_{ij}^{(k-1)} a_j^{(k-1)} + b_j^{(k-1)}\right). \quad (7)$$

2.2.2. Recurrent Neural Network. NMT is mostly implemented by recurrent neural networks as shown in Figure 5.

$$h_t = f(h_{t-1}, x_t). \quad (8)$$

The input x is mapped to the output o through a recurrent neural network. Target sequence y , loss function L , weight matrix U , W , and V . The range of the time series T is $[1, T]$.

$$a_t = Wh_{t-1} + Ux_t + b,$$

$$h_t = \tanh(a_t),$$

$$o_t = Vh_t + c, \quad (9)$$

$$\hat{y}_t = \text{soft max}(o_t).$$

The recurrent neural network makes the input sequences of different lengths have the same input vector dimension.

2.3. Semantic Ordering Model Based on Fuzzy Theory

2.3.1. Overview of Semantic Classification. The model can use semantics according to its performance and can be accurately divided into hierarchical semantics and interleaved semantics [21].

$$f(A_i, B_i) = \begin{cases} A_{i-1}, & i = 1, 2, 3, \dots, \\ B_{i-1}, & i = 1, 2, 3, \dots \end{cases} \quad (10)$$

When $B_{i-1} = 1 + A_i$, the semantics of the English to be reordered are homogeneous, and the front-end data of A_i is

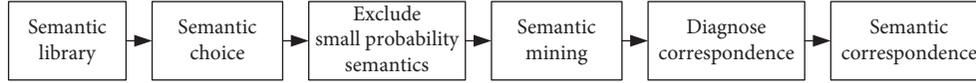


FIGURE 5: Hierarchical semantic ordering process.

replaced by the symbol A_{i-1} ; when $A_{i-1} = 1 + B_i$, the semantics of the reordering are progressive; to-be-sequenced semantics are neither homogeneous nor progressive, so they are treated as interval semantics [22, 23].

2.3.2. Calculation of Semantic Optimal Similarity. When performing English semantic sorting, it is necessary to consider the sorting structure problems in two different directions [24–26]. Hierarchical English semantics uses differences in two different directions to perform semantic sorting.

Figure 5 shows the hierarchical semantic sorting model process. Figure 6 shows the interleaved semantic sorting model.

The sequence structure is interrupted to prevent similar semantic sequence errors during the sorting process. Then perform sequencing, perform a second diagnosis on the sequencing results, and then output the results.

Assume that L_1 is any semantic in the ordering semantics and L_2 is the fuzzy correspondence result of L_1 [27, 28]. It can be obtained that the similarity between L_2 and L_1 is

$$\text{sim}(L_1, L_2) = \frac{\eta}{\eta + d} \quad (11)$$

Use the weighted analytic method to describe formula (11), and its expression is as follows:

$$\text{sim}(s_1, s_2) = \sum_{i=1}^4 \delta \text{sim}(L_1, L_2), \quad (12)$$

where δ represents the weight, s_1 and s_2 represent different data in the running process, L_1 and L_2 represents the semantics in the running process, and $\sum_{i=1}^4 \delta = 1$.

The weighted analytic hierarchy process is carried out 4 times, which is the description of the independent relationship between L_1 and L_2 .

$$S_{\max} = \frac{\sum_{i=1}^n [\varphi_1 \text{sim}(s_1, s_2) + \varphi_2 \text{sim}(L_1, L_2)]}{n} \quad (13)$$

The number of child nodes is n , $\varphi_2 = 1 - \varphi_1$.

3. Experiments

3.1. Experimental Data Set. The research in this subject uses Chinese-English bilingual corpus pairs as the data set, but since there is no large-scale Chinese-English bilingual subtitle public data set, the existing scattered data set is only tens of thousands, which is far from satisfying deep learning. For data set size requirements, experimental data needs to be obtained by yourself. The resources of individual subtitle groups are relatively authoritative in the same type of website, the resources are updated in a timely manner, and the data size is large. Therefore, this topic uses the subtitle

website as the source of the data set. On the subtitles website, you can browse the subtitles uploaded and posted every day at the subtitle downloading site. These subtitles include subtitles in different languages, different formats, and different sources.

Generally, the links of the subtitle compression package are on the movie subtitle web page. Therefore, to obtain the subtitle compression package through a crawler, you must first obtain the link of each movie subtitle web page on the subtitle download page and then find the download link of the subtitle compression package. In order to extract these two types of positioning information, it is necessary to review web page elements, locate the location of links, and then gradually filter based on the hierarchical structure of the links. If there is a higher demand, deeper filtering of the content is required through pattern matching and other methods. After getting the download link, get the link content through the crawler and finally get a total of 36,000 effective subtitle compression packages.

Since the uploaded subtitle compression package not only includes the subtitle group itself but also some other sources, such as reprinted from other subtitle groups, these subtitles are not uniformly regulated and strictly restricted, and the structure of the compression package is also different. Some subtitle files are under first-level compression packages, but some subtitle files exist under deeper subtitle compression packages. The general extraction processing method cannot extract the subtitle files in the deep compression package. Therefore, in order to fully obtain the data resources, we first extract the compressed package file recursive structure under the compression package based on the subtitle compression package we have obtained. Here, there are 4,000 new compression packages extracted through iteration, and a total of 40,000 compression packages as shown in Table 1.

3.2. Experimental Environment. This project is based on Theano, a deep learning framework for experiments. Theano has a large number of convenient optimization libraries. It is based on an efficient multidimensional array and makes full use of the GPU to perform operations, which can greatly improve the operation efficiency. The specific configuration of the experimental platform is shown in Table 2.

3.3. Parameter Optimization. When the sample set size is small, the gradient can be obtained by calculating the error of all samples in each iteration. The gradient value obtained in this way is better globally. However, if the size of the sample set exceeds a certain range, due to memory limitations, etc., it is unrealistic to calculate the error of all samples at a time. However, each time only the error of a single sample is calculated for further gradient correction, due to the

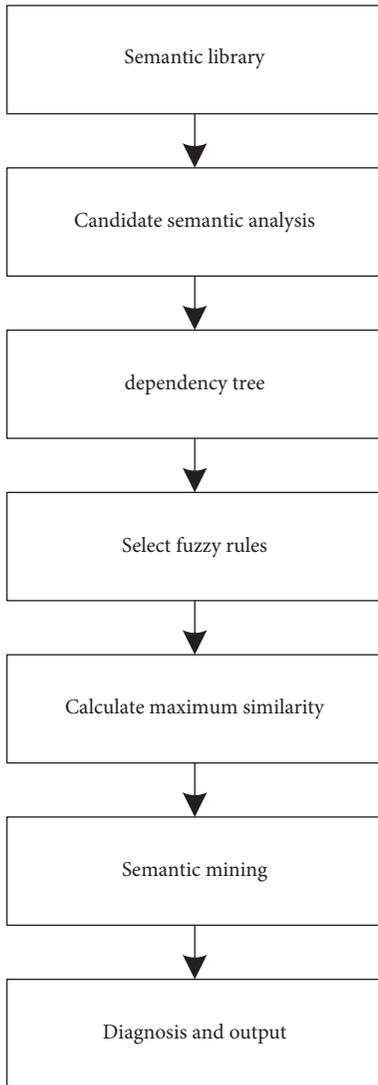


FIGURE 6: Staggered semantic ordering process.

TABLE 1: Comparison of the average correctness of the three models.

Methods	Accuracy
Neural network method	0.655
Machine learning methods	0.786
Method of this article	0.854

TABLE 2: Experimental platform configuration.

OS	CPU	Graphics card	RAM (GB)	Hard disk (TB)
Ubuntu 14.04 64-bit	I7-5930K CPU	NVIDIA GeForce GTX TITAN X	128	2.5

difference of each sample, a mutually canceling effect may be formed, which results in the phenomenon of oscillating the results back and forth and cannot converge. Therefore, using the batch method to select a suitable size can not only

effectively improve the memory utilization through the parallelization of matrix multiplication but also speed up the processing speed. Within a certain limit, as the batch size increases, the globality of gradient optimization becomes better and better, and the possibility of gradient non-convergence also becomes lower and lower.

4. Discussion

4.1. Semantic Reordering Time Analysis. In order to verify the superiority of the improved model proposed in this paper in semantic ordering, this paper uses the neural network method, machine learning method, and comparative verification analysis with the improved model as shown in Figure 7.

From Figure 7, the model in this paper has the highest ranking performance, followed by machine learning methods. With the passage of time, when using this model for semantic sorting, the time-consuming semantic sorting is the least.

The comparison of the average timeliness of the three models is shown in Table 3. The results show that the machine translation selection permutation model is better than other models in terms of timeliness, which is nearly 15% higher. Figure 8 shows the average timeliness of the three models.

According to the data in Table 4 and Figure 8, we can look at the comparison of the translation time of the three models. Among them, the neural network model takes 13.55 s, the cluster learning model takes 8.56 s, and the time required in this article is 5.58 s. According to the comparison of time, the neural network takes time The longest, the learning model is the second, and the time spent in this research is the shortest, indicating that the content of this research is practical.

Although we have conducted many experiments to test the translation model system, the experimental results are contingent and sudden and cannot well represent the search engines' ability to retrieve legal provisions. Therefore, the experimental conclusions obtained are contingent and cannot think it is an authoritative conclusion.

4.2. Accuracy Analysis of Semantic Ordering. Figure 9 shows the experimental results of semantic sorting. It can be seen from Figure 8 that, except for the model in this paper, none of the other models can effectively optimize the semantics of English.

Table 5 shows the comparison of the average correctness of the three models. The results show that the machine translation selection permutation model is better than other models in terms of timeliness, an increase of nearly 4%, and can reach over 90%, which has obvious advantages. Figure 10 shows the average sequencing correctness rate of the three models.

According to the data from Figure 10, you can clearly see the comparison of the accuracy of the three translation models. The accuracy of the neural network model is 0.655, and the accuracy of the machine learning model is 0.786.

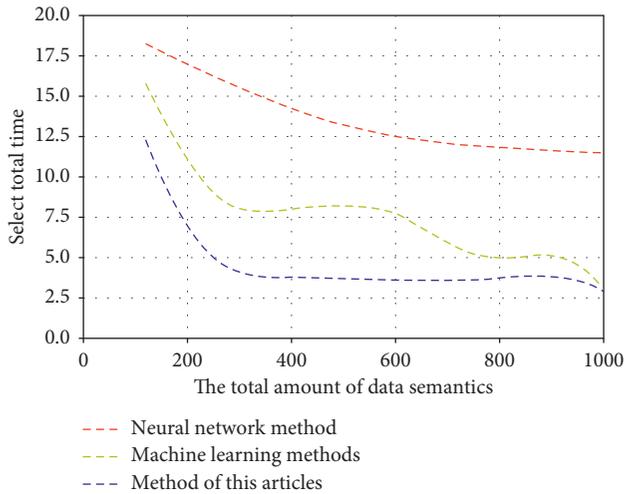


FIGURE 7: Semantic ordering time-consuming curve.

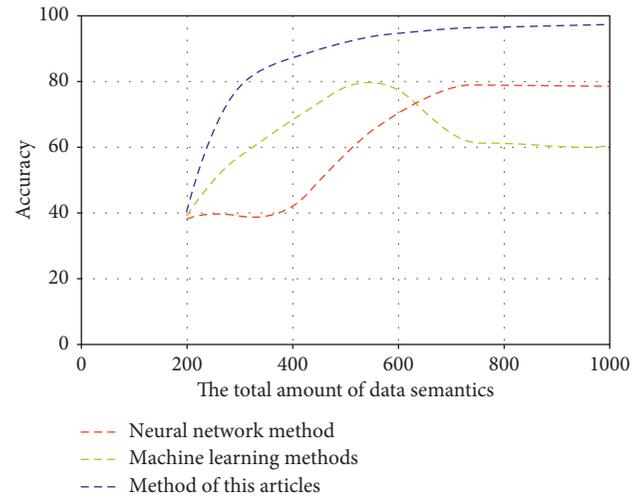


FIGURE 9: Analysis of semantic ordering accuracy.

TABLE 3: Comparison of average timeliness of three models (first experiment).

Method	Shuffle unit time
Neural network method	13.55
Machine learning methods	8.65
Method of this article	5.58

TABLE 5: Comparison of average timeliness of three models (third experiment).

Method	Shuffle unit time
Neural network method	14
Machine learning methods	9.21
Method of this article	5.74

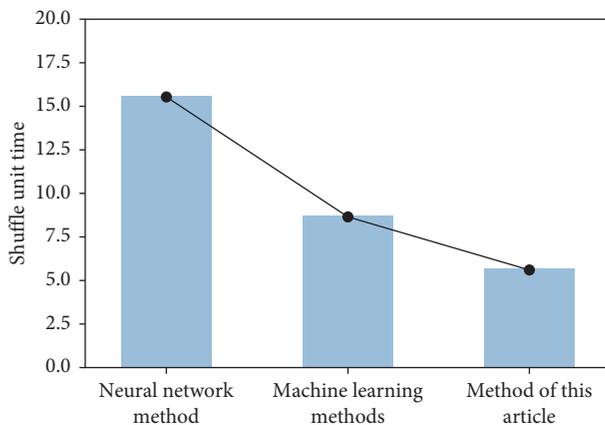


FIGURE 8: Average timeliness of three models.

TABLE 4: Comparison of average timeliness of three models (second experiment).

Method	Shuffle unit time
Neural network method	13.75
Machine learning methods	9.13
Method of this article	5.82

This article explores that the accuracy of the translation model is as high as 0.854. The accuracy rate of the translation model studied in this article is far ahead, and the accuracy rate of the neural network model is the lowest. According to data comparison, the translation model studied in this article is desirable in terms of time-consumption and accuracy.

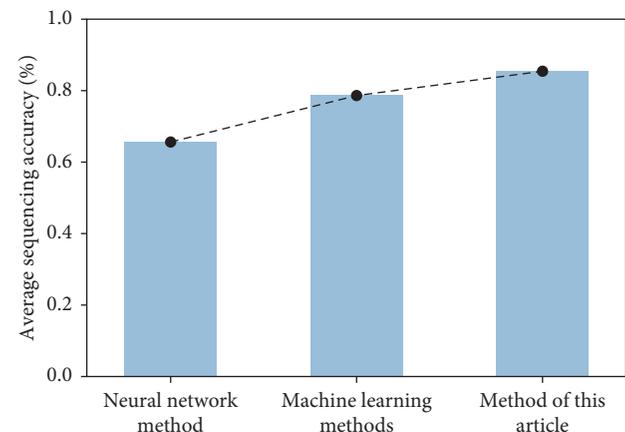


FIGURE 10: Average correctness of the three models.

5. Conclusion

As an efficient language conversion tool, machine translation can perform equivalent conversion of different languages. It has great practical significance. The development of deep learning-related technologies has also improved machine translation methods and performance. Machine translation has undergone a development process from rule-based to statistics-based to neural network-based. Statistical machine translation replaces artificially formulated rules with a data-driven model to solve the problem of bottlenecks in acquiring translation knowledge. The end-to-end model structure of NMT and the use of neural networks have simplified the entire process of translation, solving problems such as the

difficulty of designing original features and the use of nonlocal contextual information.

The main content of this paper is to address the current problems of English machine translation, and propose a fuzzy theory of English machine translation semantic ordering model. NMT is to build an end-to-end neural network model framework to cover the entire translation process, to achieve end-to-end machine translation. Neural networks are used to construct the encoder and decoder parts, respectively. Source and target language sequences: this paper adds the function of semantic ordering on the basis of NMT, which can further improve the accuracy of translation.

Finally, through comparative experiments with machine learning neural networks, it is found that using this model for English semantic ordering has better accuracy and time-consumption than the traditional English semantic ordering model, which has certain advantages. However, the relevant theories and techniques of fuzzy translation have not formed a systematic and complete research theory among experts and scholars. It appears to be too fragmented. It is still in the initial stage and requires in-depth exploration to solve the cost and time in practical applications and technology and other related issues.

Data Availability

No data were used to support this study.

Conflicts of Interest

The author declares no conflicts of interest.

Acknowledgments

This work was supported by the Research on the Construction of Medical University Portal Website Based on User Experience (no. 2019-LYZZHYB025).

References

- [1] R. Sennrich, B. Haddow, and A. Birch, "Improving NMT models with monolingual data," *Computer Science*, vol. 20, no. 5, pp. 16–25, 2015.
- [2] F. Hill, K. Cho, and S. Jean, "The representational geometry of word meanings acquired by NMT models," *Machine Translation*, vol. 31, no. 2, pp. 3–18, 2017.
- [3] J. Su, J. Zeng, and D. Xiong, "A hierarchy-to-sequence attentional NMT model," *IEEE/ACM Transactions on Audio Speech & Language Processing*, vol. 99, no. 1, pp. 15–23, 2018.
- [4] S. Marzouk and S. Hansen-Schirra, "Evaluation of the impact of controlled language on NMT compared to other MT architectures," *Machine Translation*, vol. 33, no. 3, pp. 156–167, 2019.
- [5] J. Zhang and C. Zong, "Deep neural networks in machine translation: an overview," *IEEE Intelligent Systems*, vol. 30, no. 5, pp. 16–25, 2015.
- [6] A. Cerasa and A. Quattrone, "Linking essential tremor to the cerebellum-neuroimaging evidence," *The Cerebellum*, vol. 15, no. 3, pp. 263–275, 2015.
- [7] M. Popovic, "Language-related issues for NMT and PBMT for English–German and English–Serbian," *Machine Translation*, vol. 16, no. 2, pp. 1–17, 2018.
- [8] O. Caglayan, M. García-Martínez, A. Bardet, W. Aransa, F. Bougares, and L. Barrault, "NMT-PY: a flexible toolkit for advanced neural machine translation systems," *The Prague Bulletin of Mathematical Linguistics*, vol. 109, no. 1, pp. 15–28, 2017.
- [9] D. Ataman, M. Negri, M. Turchi, and M. Federico, "Linguistically motivated vocabulary reduction for neural machine translation from Turkish to English," *The Prague Bulletin of Mathematical Linguistics*, vol. 108, no. 1, pp. 331–342, 2017.
- [10] K. Filip, A. Toral, M. Victor, and S. Cartagena, "Fine-grained human evaluation of neural versus phrase-based machine translation," *The Prague Bulletin of Mathematical Linguistics*, vol. 108, no. 1, pp. 121–132, 2017.
- [11] E. Hasler, A. De Gispert, and F. Stahlberg, "Source sentence simplification for statistical machine translation," *Computer Speech & Language*, vol. 45, no. 3, pp. 221–235, 2016.
- [12] P. Williams, R. Sennrich, M. Post, and P. Koehn, "Syntax-based statistical machine translation," *Synthesis Lectures on Human Language Technologies*, vol. 9, no. 4, pp. 1–208, 2016.
- [13] A. Mihael, T. Marco, T. Sara, and B. Paul, "Leveraging bilingual terminology to improve machine translation in a CAT environment," *Natural Language Engineering*, vol. 12, no. 15, pp. 1–26, 2017.
- [14] T. H. K. Beh and D. J. Canty, "English and Mandarin translation using Google Translate software for pre-anaesthetic consultation," *Anaesthesia & Intensive Care*, vol. 43, no. 6, pp. 792–793, 2015.
- [15] A. She and F. L. Department, "A brief analysis of the new translation-comparison model—from the view of language transfer," *Campus English*, no. 43, pp. 234–244, 2017.
- [16] F. Guzmán, S. R. Joty, and L. Márquez, "Machine translation evaluation with neural networks," *Computer Speech & Language*, vol. 45, pp. 180–200, 2016.
- [17] Y. Cui, S. Zhao, and H. Wang, "Identifying brain networks at multiple time scales via deep recurrent neural network," *IEEE Journal of Biomedical and Health Informatics*, vol. 46, no. 13, pp. 10–18, 2018.
- [18] X. Jiang, J. Sun, and H. Ding, "Video image de-fogging recognition algorithm based on recurrent neural network," *IEEE Transactions on Industrial Informatics*, vol. 46, no. 16, pp. 13–19, 2018.
- [19] Y. Chen, Q. Cheng, and X. Fang, "Principal component analysis and long short-term memory neural network for predicting dissolved oxygen in water for aquaculture," *Nongye Gongcheng Xuebao/Transactions of the Chinese Society of Agricultural Engineering*, vol. 34, no. 17, pp. 183–191, 2018.
- [20] E. Mohamed, "The relation of artificial intelligence with Internet of things: a survey," *Journal of Cybersecurity and Information Management*, vol. 1, no. 1, pp. 24–30, 2020.
- [21] K. Gregor, I. Danihelka, and A. Graves, "DRAW: a recurrent neural network for image generation," *Computer Science*, vol. 12, no. 10, pp. 1462–1471, 2015.
- [22] Z. Lv, Y. Han, A. K. Singh, G. Manogaran, and H. Lv, "Trustworthiness in industrial IoT systems based on artificial intelligence," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 2, 2020.
- [23] A. Zl, A. Dc, A. Rl, and B. Aa, "Artificial intelligence for securing industrial-based cyber-physical systems," *Future Generation Computer Systems*, vol. 117, pp. 291–298, 2021.

- [24] J. Suo and Y. Liu, "Semantic similarity algorithm based on agricultural ontology and its application on crop ontology," *Transactions of the Chinese Society of Agricultural Engineering*, vol. 32, no. 16, pp. 175–182, 2016.
- [25] W. Lu, Y. Cai, X. Che, and K. Shi, "Semantic similarity assessment using differential evolution algorithm in continuous vector space," *Journal of Visual Languages & Computing*, vol. 31, pp. 246–251, 2015.
- [26] P. Li, H. Wang, Q. Kenny, Z. Wang, X. Hu, and X. Wu, "A large probabilistic semantic network based approach to compute term similarity," *Knowledge & Data Engineering IEEE Transactions on*, vol. 27, no. 10, pp. 10–17, 2015.
- [27] R. Delmonte, "Dependency treebank annotation and null elements: an experiment with VIT," *Studies in Computational Intelligence*, vol. 589, no. 1, pp. 25–50, 2015.
- [28] J. Jiang and H. Liu, "The effects of sentence length on dependency distance, dependency direction and the implications-Based on a parallel English-Chinese dependency treebank," *Language Sciences*, vol. 50, no. 2, pp. 93–104, 2015.