**Research Article**

**Usability of Computer-Aided Translation Software Based on Deep Learning Algorithms**

Junjun Liu

School of Foreign Languages, Jinzhong University, Jinzhong 030619, Shanxi, China

Correspondence should be addressed to Junjun Liu; ljjljun@163.com

Received 24 December 2021; Revised 16 February 2022; Accepted 18 March 2022; Published 14 April 2022

Academic Editor: Qiangyi Li

Copyright © 2022 Junjun Liu. 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 recent years, due to the development of computer technology and information technology, web technology has changed the mode of translation at an alarming rate. The rapid development of information technology and globalization has increased the demand for translation, especially technical translation, and the use of computer-assisted translation software can greatly improve the quality and efficiency of translation work. The purpose of this article is that under the premise of continuous advancement in computer technology, computer-assisted translation can effectively improve the translation efficiency of translators and the quality of translated text. This article references the practicality of computer translation software as the benchmark and uses computer-aided translation software based on deep learning as the core. At the same time, it introduces the current popular microservice concept to build an electronic computer-assisted translation software based on microservice architecture. Based on the performance of the system, the high availability and scalability of the system are enhanced, so that the entire system can provide stable and efficient computer-assisted translation services for users. At the same time, the usability test method is used to compare and evaluate two common computer-aided translation software, Trados and Wordfast. By observing, recording, and analyzing user behavior and related data, the five attributes of usability can be learned and memorable. The experiments show that the effect of this study on computer-aided software with the help of deep learning knowledge can produce good results, and the robustness and scalability of the software have been enhanced, increasing the competitiveness of the software itself in translation software.

1. Introduction

Translation is a bridge connecting people in different languages. The history of this language can be traced back to 3000 BC. Translators used to work with pens, papers, and words. Translation is the most direct and effective way to resolve communication barriers between natural languages [1]. However, today, with the advent of the Internet, translators are becoming more dependent on computers. On the one hand, a lot of translation is needed. The number of source texts may be millions of words, involving multiple fields, but translators may be required to complete translations within a week [2]. With the shortening and increasing of translation time, the computer-assisted translation is regarded as a huge help for translators to meet the increasing demand for translation. It can not only save costs and time but also improve quality and efficiency [3]. Translators have achieved many successes in applying computer-assisted translation in translation operations. On the other hand, translation companies now value translation technology [4]. Therefore, in these cases, it is generally agreed that computer-aided translation has now become a necessity of the translation community [5]. An intelligent-assisted computer is used to increase translation speed and fill knowledge gaps. Machine technology, natural language processing technology, and artificial intelligence technology are developing rapidly. Many machine translation tools, such as Lingoes, Google Translate, Youdao, are developed. These translation tools show us another point of view that is completely different from traditional translation methods and has brought us a revolution in the field of translation. They can help people with translation a lot: to find
vocabulary online and translate automatically [6]. They can translate a paragraph of a sentence in seconds, which is almost impossible for humans. But their shortcomings are also obvious: the translation quality is low. Another revolution is the concept of computer-aided translation [7]. Computer-aided translation (CAT) tools are software designed to help translators do their jobs faster and more efficiently. More and more translation companies are using and benefiting from CAT tools that are based on powerful features such as terminology lists, translation memories, project management, and alignment features, which are quickly gaining popularity, winning in a highly competitive market, and gaining their share [8]. CAT tools can greatly improve the efficiency and quality of translation and make it easier for teams to collaborate and complete large-scale translation tasks [9]. Not only that, but CAT tools can also meet most of the translation work in daily life, avoiding the need to combine multiple translation tools for translation, which solves this pain point very well.

In terms of the availability of CAT tools, Momenzadeh et al. integrated a decision analysis method from a software engineering perspective and proposed an evaluation framework for translation assistance systems [10]. The research gives the evaluation methods that provide process guidance for each method, and they are applied to the evaluation of translation assistance systems. Miholka et al. constructed a CAT software evaluation framework and design scenarios suitable for different stakeholders, which can accurately evaluate the weight of each indicator in the evaluation system and the performance of the system in each scenario, but the practicability is not strong [11]. In addition to the above research on the evaluation framework of CAT software, there are scholars who have made specific evaluations of CAT software, such as Jonathon N and other experimental methods through translation task experience, comparing open-source software (Pootle) and commercial software (Trados). He believes that compared to open source software, paid commercial software can help translators complete translation projects more efficiently and conveniently [12]. Domestic research on computer-aided translation can be divided into three categories: CAT theoretical research, CAT technology and tool research, and CAT teaching research [13]. In the research of CAT tools, some scholars have compared the advantages and disadvantages of different CAT software and given suggestions for selection. The main benefit is obviously a huge gain in time and efficiency; however, when translating specialized texts and literary texts, the change in the meaning behind each word and sentence is significant. This type of research is more and the content is similar. For example, Baker et al. based on the quality concept, compared the functionality and ease of use of four common CAT software [14]. Lu et al. mainly based on the experience of previous scholars and did not use an evaluation framework or model. They evaluated four specific CAT software and gave recommendations for choosing a CAT software [15]. However, its conclusion is too specific. Although it has a certain reference value, it cannot be applied to other CAT software, and there are certain limitations in terms of timeliness. Another group of scholars proposed a method for systematic evaluation of CAT tools: Huang et al. proposed a four-step evaluation framework for CAT tools by analyzing existing research and combining it with the national standards of Software Engineering Product Evaluation. Its design philosophy mainly includes four points, two of which are from the perspective of the user, that is, the quality is determined by the user and the user has different needs; the other two are from the perspective of the quality of the software. The design concept of its CAT tool evaluation framework is of great inspiration to the creation of this article [16]. Bangyal et al. start with data preprocessing (replacement of missing values, denoising, tokenization, and stemming). Bangyal et al. apply a semantic model with term frequency and inverse document frequency weighting to represent the data [17]. Bhattacharjee proposed a class of cache prefetchers triggered by page table walk (PTW) activity [18]. Wang proposes a real-time flash translation layer (RFTL) scheme to evenly distribute the garbage collection time cost, thereby guaranteeing a near-optimal worst-case response time [19].

With the development of computer-aided translation technology, computer-aided translation software has become more widely used in translation practice. Computer-assisted translation technology combines the super computational power of computer, memory, and the creation of translators, so that translation practitioners can reduce repetitive labor and greatly improve efficiency in translation (nonliterary translation) practice. In this process of human-computer interaction, the computer provides assistance, and the translator grasps the final translation quality, without the burden of rigidity and obvious errors in machine translation. Therefore, CAT software plays a very important role in the daily work of translators. Not only professional translators use CAT software in their work, but also groups such as part-time translators, translation students, and translation enthusiasts have begun to pay attention to computer-assisted translation technology. Computer-assisted translation courses are offered in the translation major, and even universities have added computer-assisted translation majors to train talents in this area. Behind the good momentum of the continuous development of computer-aided translation, we should also see the mixed market of CAT software. There is a wide variety of software at home and abroad. Some products are expensive but the availability is not high. Translators have many problems in software selection and use. What exactly is the problem, what factors are affecting the user experience of CAT software, where is the contradiction between demand and supply, etc. are all urgent problems to be solved. Throughout the study of computer-assisted translation technology at home and abroad, most of them focus on one-way research and technical analysis. Among them, there is less research on the usability of CAT software and less from the perspective of user needs. No matter how professional the computer-aided translation is, CAT software is always a product for translator users, so we must consider the market demand. This article will focus on this point, cut in from the perspective of user experience, analyze the problems raised above more intuitively, try to
find out countermeasures, and make suggestions based on the research results [11].

This article interviews translators based on the user experience of using CAT software in the process of English-Chinese and Chinese-English translation, to understand their subjective experience of using CAT software in translation practice, and to let users talk about the advantages and disadvantages of the software used. The author explains why it is not used. According to the results of the interview, the author designed a questionnaire around the usability evaluation indicators and distributed the questionnaire through the network. The target population covered was professional or nonprofessional translators. Finally, the results of the statistical questionnaire were sorted out and data analyzed. Combining interviews with actual experience to make hypotheses, the four dimensions of learnability, efficiency, error rate, and subjective satisfaction in the five dimensions of usability have been tested on whether users choose to use CAT software. Based on the combination of survey results and theory, the author constructed a CAT software usability evaluation index system and selected a scenario to determine the weights of various indicators in the system using the principal component analysis method.

2. Proposed Method

2.1. Usability Model of Translation Software Based on Deep Learning. This section will introduce the deep network structure model of deep learning: the basic principles and training steps of deep belief network (DBN) and introduce the detailed framework of the user complaint prediction model based on the DBN model in this study [20, 21].

2.1.1. DBN Model. Restricted Boltzmann machine (RBM) is the first model to train each layer of a shallow network with an unsupervised learning algorithm. RBM is a special topology of Boltzmann machine (BM). The principle of BM originates from statistical physics, and it is a modeling method based on energy function, which can describe the higher-order interaction between variables. BM is a symmetrically coupled random feedback binary neural network. The top layer represents the unit feature vector of the hidden layer, the bottom layer is the unit vector of the visible layer, and the link weight w is the model parameter [22, 23].

The visual layer and the hidden layer of BM, the hidden layer and the hidden layer, and the visual layer and the visual layer are all connected to each other, and the correlation between the units is expressed by the weights [11, 24]. RBM restricts the connection between the same layer of BM [25, 26]. It is a kind of double-layer, undirected random neural network model with symmetry and no self-feedback. When the state of the v-unit in the visible layer is given, the conditions between the h-units in each hidden layer are independent [27]. Similarly, when the state of the h-unit in the hidden layer is given, the conditions between the v-units in each visible layer are independent. For a given set of states \((v, h)\), the energy formula for RBM is as follows:

\[
E(v, t) = -\sum_{i=1}^{n} a_i v_i - \sum_{j=1}^{n} b_j h_j - \sum_{i=1}^{n} \sum_{j=1}^{n} v_i w_{ij} h_j.
\]

Among them, \(w_{ij}\) represents the connection weight between visible layer \(i\) and hidden layer \(j\); \(a_i\) represents the offset of visible layer \(i\); \(b_j\) represents the offset of hidden layer \(j\). Let the parameters of the RBM be \(\theta\):

\[
\theta = \{w_{ij}, a_i, b_j\}.
\]

When the parameter \(\theta\) is fixed, the joint probability distribution of \((v, h)\) can be obtained based on the energy function:

\[
P(v, h) = \frac{e^{-E(v,h)}}{Z(\theta)},
\]

\[
Z(\theta) = \sum_v \sum_h e^{-E(v,h)},
\]

where \(Z(\theta)\) is the normalization factor. For practical problems, the edge distribution of the joint probability distribution \(P(v, h)\) needs to be solved:

\[
P(v) = \frac{1}{Z(\theta)} \sum_h e^{-E(v,h)}.
\]

Due to the conditional independence and symmetry between the RBM hidden layer and the visible layer, when \(v\) is given, the \(h\) activation probability is as follows:

\[
P(h_j = 1) = \sigma \left( b_j + \sum_i v_i w_{ij} \right).
\]

When \(h\) is given, the activation probability of \(v_i\) is as follows:

\[
P(v_j = 1) = \sigma \left( a_j + \sum_i v_i w_{ij} \right).
\]

\[
\sigma(z) = \frac{1}{(1 + \exp(-z))}.
\]

From this model parameter \(\theta\), the optimal parameters of the model can be obtained by maximizing the log-likelihood function of the RBM on the training set.

The deep confidence network (DBN) is a deep learning model built in tandem with RBM as a unit. It is trained using an unsupervised greedy layer-by-layer algorithm. It is also a probabilistic generative model that captures higher-order data in hidden layers. The joint distribution of the unit feature vector \(x\) of the visible layer \(v\) and the hidden unit vector \(h\) (where the hidden layer is the \(l\) layer) is as follows:

\[
P(x, h^1, h^2, \ldots, h^l) = \left( \prod_{k=0}^{l-2} P(h^{k+1} | h^k) \right) P(h^{l-1}, h^l).
\]
Among them, the conditional probability distribution of the visible layer of the \( k + 1 \) hidden BM layer is as follows:

\[
X = h^*, \ p(h^*|h^{k+1}).
\]  

(8)

### 2.1.2. Usability Research Framework of User-Assisted Translation Software based on DBN Model

In recent years, deep learning has been widely used in many fields such as image processing, speech recognition, and search engines. Deep learning has two basic features when learning from data: first, increasingly complex representations are formed by a progressive, layer-by-layer approach; second, these progressive representations in between are learned together, with changes at each layer requiring consideration of the needs of both the upper and lower layers. It can learn high-level, abstract feature representations required for classification from large amounts of data. In order to obtain more complex and nonlinear combination features, this study will innovatively apply deep learning methods to the problem of assisted translation software usability research. Deep learning has the advantage of realizing better performance on translation problems and simplifies the problem-solving steps as it fully automates feature engineering. With deep learning, all features can be learned at once. In order to use the deep learning DBN model in this study to study the process framework of the auxiliary translation software usability problem, the feature selection of the software-related business knowledge from the original database of the translation software is used to construct the original features of the model (BSS, OSS, and CSR features). Then, the original features are inputted into the DBN feature learning model, multiple model trainings are performed, the optimal initialization parameter settings are selected, the highest-level feature expression is obtained under the model training of the optimal parameters, and they are inputted into random forest and logistic regression classification for the evaluation of the model. The learning features are mixed with high-level nonlinear combined features based on DBN model automatic learning for software availability prediction and compared with the other two models, confirming the effectiveness of DBN automatic learning features and the deep learning model in the field of auxiliary translation software usability research and availability.

### 2.1.3. Deep Neural Network

Neural networks actually have a very long history. As early as the late 1950s, Rosenblatt designed and produced a "perceptron." This concept is very close to the actual mechanism of neurons. By connecting multiple layers of perceptrons back and forth, a decision network can be formed, and thus laid the cornerstone for the research of neural network. The landing of "perceptron" means that for the first time, humans have implemented the theoretical discussion of artificial neural networks in the form of engineering. Both neural networks and artificial intelligence are mathematical models as well as algorithms that mimic human behavior. Research on neural networks can facilitate or accelerate the development of artificial intelligence. As one of the most important branches of artificial intelligence, neural networks are different from traditional statistical learning methods. It can use a relatively complex structure to characterize the internal structural characteristics of data. This feature makes it useful for processing high-dimensional data. There is a strong ability to resist noise. Neural networks have generally gone through three periods so far. The idea of the first generation of neural networks originated from the MCP artificial neuron model in 1943. At that time, it was hoped that computers could be used to simulate the process of human neurons. The model simplifies the neuron into three processes: linear weighting of input signals, summing, and nonlinear activation. The first use of MCP for machine learning was the perceptron algorithm invented by Rosenblatt to deal with linear binary classification problems. The sigmoid function is an S-shaped function commonly found in biology, also known as an S-shaped growth curve. The second-generation neural network invented the BP algorithm for multilayer perceptron (MLP) in 1986 and used sigmoid for nonlinear mapping, so that the second-generation neural network has the ability to solve nonlinear classification and learning. The sigmoid function has the advantages of being smooth and easy to derive, but its activation function is computationally intensive, and when back propagating to find the error gradient, the derivation involves division; when back propagating, the gradient will easily disappear, thus making it impossible to complete the training of the deep network. The third-generation neural network is also the neural network used so far. It is a deep neural network represented by a deep belief network stacked by multilayer-restricted Boltzmann machines. It not only inherits the learning mechanism of the second-generation neural network, but also proposes a solution to the gradient vanishing problem in deep network training, that is, unsupervised pre-training combines weight initialization and supervised training fine-tuning.

### 2.2. Computer-Aided Translation Software

The concept of computer-assisted translation technology is divided into broad and narrow senses. Therefore, computer-aided translation software also has a difference between broad and narrow senses. Under the broad concept, the computer-aided translation software can include word processing software, speech recognition software, instant messaging software, online translation software, and any software that helps translators handle translation tasks during the translation process. This article discusses computer-assisted translation software in the narrow sense, that is, a computer tool that uses a database and translation memory technology to assist translators to complete translation tasks and improve translation efficiency.

#### 2.2.1. Usability Index

Nielsen conducted a comprehensive analysis of the usability of the software. He believes that usability generally includes 5 attributes. The author makes slight adjustments to it according to the actual situation of this article:
2.2.2. User Experience. According to Garrett’s theory, user experience refers to how the product contacts and functions with the outside world, that is, how the user contacts and uses the product. The user experience is often reflected in the nuances of the product and is very important. A good user experience can increase customer loyalty and bring greater benefits. The elements of user experience can be interpreted from five levels. From abstract to specific, they can be divided into the strategic layer, scope layer, structure layer, framework layer, and presentation layer. Because Garrett’s theory is mainly applied to commercial websites, this article makes minor adjustments to the above-mentioned elements based on the actual situation of the CAT software. The content is as follows:

The strategic layer focuses on two aspects. One is the external demand from the enterprise, that is, what the user wants; the other is the internal enterprise goal, that is, the expectations of the CAT software vendor for its products. In the scope layer, the software elements are transformed into the functional combination of exploring products, that is, the combination of commonly used and unused functions; the information aspect becomes content requirements. Coming to the structural layer, the elements of the software evolve into interaction design, that is, how the system responds to user requests; the information aspect is the distribution of content elements. In the framework layer, the software side emphasizes interface design, that is, how to arrange the interface elements that allow users to interact with system functions; the information side focuses on information expression methods that promote understanding. The ultimate goal of interactive software is to allow users to operate interactively. Consumers are the direct users of software products, which requires us to put consumers first in all stages of software development. Finally, in the presentation layer, both aspects focus on visual design, that is, the appearance of the product. Each level is determined by the levels below it. Decisions at different levels are not made without intersection in time and space. Decisions at higher levels sometimes lead to reevaluation at lower levels. Therefore, the ideal way for decision-makers is to complete one level of work before the next level ends.

3. Experiments

3.1. Data Acquisition. The data can be divided into three parts: time spent, error rate, and success score. First, you need to calculate the total time required to complete each task, which is extracted from the computer screen record. Second, the number of different errors made by each participant was listed from the computer screen record. In the end, each task was successfully checked by all participants. The experimenter evaluated participants’ subjective satisfaction with the software on a 5-point scale. The questionnaire covered the system’s learnability, feedback, and system capabilities. The higher the overall score, the higher the satisfaction and the higher the usability of the software translation software. It includes modules of “performance-satisfaction,” “design-efficiency,” and “information-ease-of-use.” Based on the evaluation index system as a prototype, an attempt is made to build a possible evaluation model and determine the weights for each of the indicators.

3.2. Experimental Environment. Because this experiment requires high-performance computing, the latest Microsoft operating system and high-performance CPU are used in the computer selection to handle large amounts of data, and high-performance graphics cards dedicated to deep learning are used as support. At the same time, we use the latest version of translation software for comparison.

3.3. Experimental Steps. According to the obtained results, the author verified through regression analysis whether the factors such as satisfaction, error rate, learning cycle, and work efficiency will affect the use of CAT software to a certain extent. After the original data are collected and processed, it is brought into the DBN to train and fit the model.

Step 1. Train the first RBM model with $x = h^0$ as the input of the visual layer.
Step 2. The lower-level RBM is trained using observation samples, and the model parameters of the RBM are fixed.
Step 3. Train the output of the low-level RBM as the input of the high-level RBM. The fourth step repeats the second and third steps to repeatedly train all the RBMs to realize the initialization of the model parameters.
Step 4. After determining the model parameters, use the labeled data in the DBN auto-associative memory module to fine-tune the discriminative performance, so that a bottom-up feedback learning method can be used to adjust all the model parameters of the RBM.

When all the model parameters of the DBN are fixed, the original features (BSS, OSS, and CSR features) are inputted
to the visible layer. After automatic learning of multiple hidden layers, high-level features that better represent the data characteristics can be obtained. Deep learning multi-layer network automatically learns effective high-level features.

4. Discussion

4.1. Experimental Results and Analysis of the Impact of Usability Indicators on User Selection.

(1) For users of CAT software, under the framework of usability theory, there are many factors that will affect their choice of whether to use CAT software to complete the translation project, such as subjective satisfaction, error rate, and other factors. During the translation project, the smaller the number of large errors in CAT software, the higher the user’s preference for the software, and they may tend to continue to use CAT software. First, for the above assumptions, establish regression equations using the CAT software and subjective satisfaction, error rate, learning cycle, and work efficiency. Table 1 lists the univariate analysis results. It can be seen from the P value that all variables except the categorical variable of learning cycle and efficiency are significant at the test level of 5%. The total chi-square test P value of all independent variables is 0, indicating that the global test of the model has statistical significance, and all variables can enter the model. The experimental results are shown in Figure 1.

(2) Table 2 gives a comprehensive test of the modulus coefficient. The P value is 0, which indicates that the independent variables included in the model have statistical significance and have a certain effect on the results. Observing the P value, it can be found that all variables except the efficiency of the variables are significant at the 10% test level. Among them, the error rate and satisfaction are statistically significant at the 5% test level. The error rate and learning cycle have a great impact on whether users use CAT software or not as shown in Figure 2. The estimated value of satisfaction is equal to 3.18, which means that if other independent variables are unchanged, the satisfaction is increased by 1 unit, and the advantage of using CAT software is changed by 3.18 times. In the same way, the error rate and the learning cycle are increased by 1 unit, and the advantages of using CAT software are changed by 0.45 times and 0.87 times, respectively, indicating that with the increase of the error rate and the learning cycle, the opportunity to use CAT software has a tendency to decrease. Although the independent variable efficiency was not statistically significant at the 10% test level, its P value was also very close to 0.1. Therefore, the above hypothesis has been verified, that is, higher satisfaction, lower error rate, shorter learning cycle, and higher efficiency will make users more inclined.

4.2. Model Parameter Settings.

(1) In order to simplify the model in this study, the DBN model trained in this study uses the same number of nodes in each layer. Since the two parameters of the number of hidden nodes and the number of hidden layers are mutually affected, these two parameters are adjusted simultaneously in the experiments in this section. For the number of hidden layers, we choose six discrete values to set, they are 1, 3, 5, 7, 9, and 15, respectively. For the number of hidden nodes in each layer, this study also chooses seven discrete values to set, which are 20, 50, 100, and 150. In the evaluation and selection of these two parameters, the size of batch_size is set to 150, the learning rate is 0.005, and the number of training iterations is set to 20. In the fine-tuning phase, this study uses gradient descent to fine-tune the network, and the learning rate is changed to 0.1. As shown in Figure 3, the DBN model is trained by adjusting the number of different hidden layers and the number of hidden nodes in each layer at the same time. The obtained feature vector is then used to obtain the AUC evaluation index.

Table 1: Univariate analysis results.

<table>
<thead>
<tr>
<th>Effectiveness</th>
<th>Score</th>
<th>Df</th>
<th>Sig</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error rate</td>
<td>26.342</td>
<td>1</td>
<td>0.009</td>
</tr>
<tr>
<td>Learning cycle</td>
<td>617</td>
<td>2</td>
<td>0.279</td>
</tr>
<tr>
<td>Satisfaction</td>
<td>23.342</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 2: Global test results of model coefficients.

<table>
<thead>
<tr>
<th>Model</th>
<th>Chi-square test</th>
<th>Df</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step</td>
<td>41.019</td>
<td>5</td>
<td>0.000</td>
</tr>
<tr>
<td>Block</td>
<td>41.019</td>
<td>5</td>
<td>0.000</td>
</tr>
<tr>
<td>Model</td>
<td>41.019</td>
<td>5</td>
<td>0.000</td>
</tr>
</tbody>
</table>
In order to balance RMSE and time consumption, this study also chooses six discrete iterations, which are 1, 5, 10, 20, 30, and 50. At this time, the number of layers is set to 5 with 150 nodes in each layer, the size of batch_size is set to 1000, and the learning rate is 0.005. During the fine-tuning phase, the gradient descent method is used to fine-tune the network, and the learning rate is changed to 0.1. Use RMSE and consumption time to evaluate the number of iterations. As shown in Figure 4, as the number of iterations increases, RMSE will decrease sharply, and then it will become a smooth state, and the time consumed also increases exponentially as the number of iterations increases. According to the above experimental results, this article finally set the number of iterations to 15, the RMSE at this time is 0.1638, and the time for model training is 27 min.

5. Conclusions

First of all, it can be found that models based on DBN feature learning and models with learning features are significantly better than models without learning features, and their AUC indicators are 5.2% and 7.1% higher than models without learning features, respectively. The prediction results of the various classifiers are roughly the same, without too many obvious advantages; in the end, it can be found that the prediction model based on DBN is also 1.7% higher than the model with learning features added; the overall prediction result based on DBN is significantly better than the result of traditional artificial feature engineering. In addition to the analysis results, the following conditions were observed during the experiment. First of all, some software installation takes several hours, but because it is an online version, there is no need to install it, which is more friendly to users. Second, the average user complains about the time it takes to load. Third, novice users suggest that the translation software simplifies the interface. To make it clearer, different audiences have different requirements for CAT software. For example, compared to individual users, institutional users have higher requirements for the comprehensiveness of CAT software. If there are multiple people translating, collaborative translation is required, so that the project management function requirements are lower, and the coverage of professional vocabulary is also higher. In order to test the evaluation index system constructed in this article, the author’s above evaluation system is used as a prototype, and an evaluation model for the part-time translator is built. At present, the system functions and interface still need to be improved. Although it can ensure the normal and stable operation of the system, there is still a long way to go before the productization.

At present, the general development trend of software is that the functions of various software are gradually converging. Therefore, to meet the growing translation needs of users, the most unsolved problem is the translation memory and termbase problems that come with the software. It is a time-consuming and labor-intensive process. If the software’s translation memory and termbase do not have rich examples related to the materials to be translated, the software will not be able to give full play to the function of improving translation efficiency. In response to this situation, many machine-assisted translation software developers also sell their own translation memories and termbases when...
they sell their software. There are still many shortcomings in this research. Due to limited time and energy, the survey sample is small and not sufficiently representative; in the four dimensions of usability covered in this article, the “use efficiency” dimension should be executed when the user reaches a proficiency level. However, the measurement of the time required for a task cannot be counted in the form of the author’s research. In the end, I chose a more subjective method that allows the respondents to judge whether the CAT software affects the translation efficiency. In addition, the development of CAT software is changing rapidly. The performance and characteristics of CAT software outlined by the author before the survey may not be comprehensive now or in the future. With the development of technology, if there is a similar research in the future, it needs to be supplemented.

With the development of computer networking, mainstream translation software can be implemented a month ago. The development of translation software in the future should be based on the user’s service goals and user experience and try to continuously improve in the following aspects. The main version of each translation software is Windon 5, while the Mac version for Apple Computer is relatively rough. Some translation software does not even have a Mac version. Because Apple Computer has gained increasing popularity in domestic and foreign markets, more users have begun using Apple computers, and various translation software developers should increase the development of the Mac version in order to seize this important future market. At present, each translation software can achieve fast query and translation of words, but due to the characteristics of machine translation, sentences or paragraphs are not translated well. Optimization can be tried by adding manual evaluation. This also requires linguistics, mathematics, and computer science.

**Data Availability**

The data that support the findings of this study are available from the corresponding author upon reasonable request.

**Conflicts of Interest**

The authors declare that they have no conflicts of interest.

**Acknowledgments**

The authors received no financial support for the research, authorship, and/or publication of this article.

**References**


