Hindawi
Scientific Programming
Volume 2022, Article ID 4254932, 12 pages
https://doi.org/10.1155/2022/4254932

Research Article

Development and Research of Computer-Aided English-Chinese Translation System Based on Particle Swarm Optimization Programming

Jianyong Li

International School, Xi'an Siyuan University, Xi'an, Shanxi 710038, China

Correspondence should be addressed to Jianyong Li; liyun@xjtu.edu.cn

Received 28 March 2022; Revised 13 April 2022; Accepted 19 April 2022; Published 9 May 2022

Academic Editor: Sheng Bin

This study combines particle swarm optimization programming and computer-aided translation to model English-Chinese translation and designs a computer-aided English-Chinese translation system with a particle swarm optimization algorithm as a collaborative evolutionary strategy. The system is refined into a user environment model, userspace adjustment, network optimization judgment scheme, multi-intelligent body position adjustment scheme, and particle swarm optimization algorithm; the fitness function, particle structure, population structure, and particle update strategy particle swarm algorithm are designed. In this study, we propose to improve the system of the Seq2seq model and offer the conversion layer. In the existing Seq2seq model, the source language sequence generates a representation vector through the encoder. Then the representation vector is directly used as the initial state of the decoder to generate the target language sequence. The application of computer-aided translation software to study abroad text translation is a brand-new attempt worth exploring. To improve translation speed and efficiency of study abroad texts, this study explores the practical process of using CAT software-assisted text translation, summarizes the translation practice, and puts forward relevant suggestions by analyzing the operation steps and problems that arise in the three stages of retranslation, translation, and posttranslation. It is found that computer-aided software is applicable to improve the translation efficiency of texts to a certain extent.

1. Introduction

With the development of computer-aided translation technology, computer-aided translation software (referred to as CAT software) is increasingly widely used. Computer-aided translation technology combines the computer's superior computing ability and memory with the translator's creativity, enabling translation practitioners to reduce repetitive work and significantly improve translation (nonliterary translation) practice efficiency [1]. In this human–computer interaction process, the computer assists, and the translator grasps the final translation quality without the fear of rigidity and apparent machine translation errors. Therefore, CAT software plays a vital role in the daily work of translators. Professional translators use CAT software in their work, and part-time translators, translation students, translation enthusiasts, and other groups have started to pay attention to computer-aided translation technology [2]. The translation majors of significant universities have opened courses on computer-aided translation, and even universities have added computer-aided translation majors specializing in training talents in this field. In this regard, even universities and colleges have established computer-assisted translation majors specializing in training talents in this field [3]. Even though various CAT software are continuously optimized, the current CAT software can handle file formats that are still limited, such as noneditable PDF format and JPEG image format, and there is also the use of other specialized software for conversion of some small-scale translation projects because of the increase in translation preparation work to extend the entire translation process. Behind the excellent momentum of the continuous development of computer-assisted translation, we should also see that the CAT software market is mixed, with a wide variety of software, some of which are expensive but not highly useable, and translators have many problems in the selection and use of the software.

Population intelligence algorithms are a generic term for a class of heuristic search algorithms that originated either from genetics or from studying the group behavior of social animals such as ants and bees. A particle represents an individual in a population intelligence algorithm [4]. The particles interact with each other and follow a specific pattern of motion, trying to find the optimal solution to the problem in the process. Commonly used population intelligence algorithms include the Genetic Algorithm (GA), Differential Evolution (DE) algorithm, particle swarm optimization (PSO) algorithm, etc. The optimization mechanism of these algorithms does not depend too much on the organization of the algorithm [5]. The structural information of these algorithms does not rely too much on the organization of the algorithm, which can be widely applied to the combinatorial optimization and computation of functions, especially in solving some optimization problems with nonlinear, nonconvex, or nondifferentiable objective functions, which show good search performance.

Among these population intelligence algorithms, the PSO algorithm is a simple form easy to use. Faster convergence compared with other population intelligence algorithms and other optimization algorithms (e.g., artificial neural network, simulated annealing), which can usually obtain good search results with low computational cost has been widely used in various scientific research work and industrial production applications related to artificial intelligence [6]. However, in solving some complex optimization problems such as biological sequence comparison, power scheduling, fault diagnosis, engineering optimization, the PSO algorithm, like other population intelligence algorithms, suffers from the defects that it is difficult to get rid of the local optimum and the difficulty in balancing the global search capability and the local search capability, which significantly affects the solution quality of the PSO algorithm in solving such complex optimization problems [7]. In addition, some complex optimization problems have high requirements on the algorithm's solution efficiency. It is still tough to find reliable solutions quickly in a limited time, even though the PSO algorithm has a fast convergence rate.

2. Related Works

The particle swarm optimization algorithm belongs to the swarm intelligent computing method, which was initially proposed by Kennedy and other scholars based on the study of bird behavior. Once the technique was introduced, it immediately attracted the attention of many scholars, so a variety of particle swarm optimization (PSO) application research results were presented, which further promoted the development of related research [8]. During the tedious operation of the software, there is always stagnation due to some detailed mistakes, adding to the burden of translation. After many attempts, familiarity with the software process will naturally improve the dilemma. However, to solve this lack of computer technology, translators should enhance their awareness of interdisciplinary learning, get more familiar with and understand the basic principles of computers, and improve their ability to solve computer technology problems independently. Many researchers have researched and improved the particle swarm algorithm in terms of convergence, parameter settings, topology, and algorithm fusion to promote the overall performance of the algorithm; based on the research of the primary particle swarm algorithm, Ibrahim et al. proposed a standard particle swarm algorithm with the introduction of standard particle swarm algorithm with inertia weights, which was improved by adding a weighting factor to the speed update, which was shown to guarantee convergence and exploration relatively well at the same time, and later evolved into the current standard PSO algorithm [9]. Many practical strategies have been derived from the study of inertia weights, such as the LDIW strategy, the FIW strategy, and the RIW strategy. In 2009, Shadiev et al. proposed an adaptive inertia weighting PSO algorithm [10]. Xue et al. proposed a variable step-size adaptive particle swarm optimization algorithm [11]. The improvement of the above particle swarm optimization algorithm based on inertia weights speeds up the algorithm's search speed, and the global search capability of the algorithm is greatly improved.

The theories related to machine translation can be traced back to the 1930s. With the development of technology, it was finally realized for the first time in 1954 to translate short texts through a computerized machine translation system [12]. Nowadays, machine translation technology has been developed considerably [13]. Machine translation can meet readers' demands with fixed forms, clearer semantics, and less demanding translation quality. However, there are still many bottlenecks in the semantic aspects of machine translation that is difficult to break through [14]. Although processing the bullet text is fast, the quality of the translation of different texts is not satisfactory.

In contrast, the computer-aided translation technology of human–computer interaction combines machine and human, which improves the speed of translation and ensures the quality of translation [15]. In the research and development of the CAT tool, H6ge proposed the evaluation framework of the translation assistance system from the perspective of software engineering by integrating the method of decision analysis, and its rating process is also cyclical [16]. Chong et al.'s user-oriented construction of CAT software evaluation framework: design scenario tests applicable to different stakeholders determine the weight of each index in each scenario in the evaluation system, execute the evaluation, and evaluate the system's performance [17]. For example, Liao et al. compared the differences between open-source software (Pootle) and commercial software (Trados) using experiments on translation task experience and concluded that paid commercial software could help translators complete translation projects more efficiently and conveniently than open-source software [18]. He argues that paid commercial software can help translators complete translation projects more efficiently and conveniently than open-source software.

Research on computer-assisted translation can be divided into three main categories: CAT theory research, CAT technology, and tools research, CAT teaching research, etc., and a small amount of research on the CAT industry. In the study of CAT tools, some scholars compare the advantages and disadvantages of different CAT software and give suggestions for selection. There are more studies in this category, and all of them are similar in content. For example, Su and Li users view quality as a basis for comparing the functionality and ease of use of four standard CAT software [19]. The study is mainly based on the scholars' experience of using the software and does not use a framework or model of measurement. It is an evaluation of four specific CAT software and finally gives recommendations for CAT software selection. However, its conclusions are too precise, and although they have some reference value, they cannot be applied to other CAT software and have limitations in terms of timeliness. The design concept consists of four main points: from the user perspective, that is, quality is user-determined, and users have different needs; the other two points are from the software quality perspective, that is, quality has characteristics, and quality characteristics are W-measurable. The design concept of its CAT tool measurement framework is very inspiring for the creation of this review.

3. Design of Computer-Aided English-Chinese Translation System Construction Based on Particle Swarm Optimization Programming

3.1. Particle Swarm Optimization Programming Model Construction. The particle swarm optimization algorithm, that is, PSO, refers to a new evolutionary algorithm based on swarm intelligence that has emerged in recent years and was developed based on the theory of evolutionary computation related to artificial life. It was initially proposed based on the study of bird behavior and was mainly used to simulate bird foraging behavior. If there is a group of birds looking for the same food in a particular area, and they do not know where the food is, but only the distance between themselves and the food, the easiest way is to search the vicinity of the bird closest to the food. In this algorithm, the solution to the optimization problem is equivalent to the bird's position in the search space, and these birds are like individual "particles." These particles have position and velocity, which is used to determine the distance and direction of flight, respectively, and the optimization function determines the adaptation value. Each particle tracks the current optimal particle and searches in the solution space. Each iteration in the iterative process is not entirely random. When a better solution is found in the iteration, it goes to the next solution based on that better solution. The optimization of real problems is based on the initial random solution and then iterates continuously until the optimal solution is found. During the iterative process, the particle evolves using the tracking of two extremes [20]. One of the extremes is the individual extremum, whose position in space can be represented by the best and optimal solution the particle is looking for; the particle swarm optimization algorithm consists of two different versions, local and global. In the international version, the other extreme value refers to the optimal solution found by the particle population, which is called the global powerful value point and is generally denoted by the best in the solution space.

In contrast, it is the best solution among the particle neighbors in the local version, which is generally denoted by best. The algorithm evaluates the quality of the solution through the fitness function. This review studies the global version of the particle swarm algorithm, so all subsequent studies are directed to the international version. Once the two extremes are found, the particle can update its velocity and position according to Equation (1). Population intelligence algorithms are a generic term for a class of heuristic search algorithms that arise either from genetics or from the study of group behavior in social animals such as ants and bees:

$$v_t = wv(t-1) - \frac{\sqrt{c_1 r_1 (p+t)}}{c_2 r_2 (g-t)}. \tag{1}$$

Among them is the inertia weight, which can regulate the strength of global and local searchability, reflects the acceleration factors of own cognitive and social-cognitive consequences, respectively, and are random numbers in the interval [0, 1], respectively, to increase the randomness in the search process. The local optimum of each particle is the global optimum of the whole population. To prevent particles from flying out of the search space, the velocity range of each particle is set. The particle search process by the fitness function f determines the merit of a particle, which influences the choice of the local optimum and the global optimum. If the text is to be translated and the content in the memory has the same expression or similar expression, then the translator can directly adopt or refer to it, thus improving the translation efficiency. The translated content can be re-imported into the memory to expand the capacity of the memory, which can provide more convenience for future translations:

$$\text{Best}_i = \sum \frac{x_i + t}{\sqrt{p-i}}. \tag{2}$$

The specific steps of the standard PSO algorithm are as follows:

Step 1: Initialize the whole population. Set the size N of the entire population, set the initial number of iterations $t = 0$, the maximum number of iterations T, set the basic parameters inertia weight w, acceleration factor, random values.

Step 2: Evaluate the fitness value of each particle according to the test function.

Step 3: Update each particle and the entire population.

Step 4: Update the velocity of each particle according to the formula and the position.

Step 5: Determine whether the termination condition of the algorithm is satisfied. If it is confident that T is reached, the algorithm stops and outputs the global optimum; otherwise, the number of iterations is added 1 ($t = t + 1$) and continues to return to step 2 iteratively until the maximum number of iterations is satisfied. The particle swarm optimization process is shown in Figure 1. Compared with other population intelligence algorithms and other optimization algorithms (e.g., artificial neural networks, simulated annealing), the PSO algorithm has the advantages of simple form, ease of use, faster convergence, etc. It can usually obtain good search results with low computational cost and has been widely used in various scientific research work and industrial production applications related to artificial intelligence.

In addition to particle swarm algorithms, Genetic Algorithm (GA), Evolutionary Programming (EP), Evolutionary Strategies (ES), and Ant Colony Optimization (ACO) are some of the most widely used iterative algorithms (Ant colony optimization or ACO) are often used to perform optimization operations. These iterative optimization algorithms are inspired by some biological phenomenon in nature and will be simulated using mathematical models. First, the set of solutions in an n-dimensional vector is defined as a "population," and the population is used as the executor of the problem. Each iteration can be considered as a selection mechanism. In evolutionary strategies and optimization algorithms in evolutionary planning, the iterations are performed in a competitive manner between the children and the parents, and the fitness of the children is the only criterion for evaluation, as long as the fitness of the children is higher than that of the parents, the children replace the parents, and vice versa. The fitness comparison compares the current value with the last iteration and the best position in history, that is, only when the particle's current position is better than the best position in history will postreplace the last iteration value. Second, the iteration direction can be set autonomously, that is, the optimization direction. Speed iteration of PSO is similar to the cross-genetic of genetic algorithm. The cross-genetic algorithm produces a new generation of particles (children) from a linear combination of two particles (parents) from the previous iteration, interspersed with variation-induced random quantities [21]. In the PSO optimization algorithm, the iteration's direction can be considered as inherited from the parent of the previous iteration as well, where the vid term is equivalent to the variation factor in the genetic algorithm in terms of its role. The degree (size) of the variation depends on the distance between the best position of the population of particles (parents) of the previous iteration and the current position of the particles. Unlike other evolutionary algorithms, the solutions of both PSO and genetic algorithms are parallel, and the solution process starts from a collection of solutions rather than from one single individual. The significant differences and advantages between PSO and other evolutionary, iterative algorithms are the following: (1) the existence of an information-sharing mechanism between particles, sharing the best position of the population Guest position, which can maximize the advantages of the population. (2) The PSO optimization algorithm can prevent the invalid and harmful changes of particles to the maximum extent. (3) PSO optimization algorithm retains and uses both position and velocity information in the iterative process and can self-adapt according to the velocity vector vid, the iterative process; other evolutionary, iterative type optimization algorithm only retains position information. The study is mainly based on scholars' experience using CAT software without using a framework or model for measurement. It is an evaluation of four specific CAT software and finally gives recommendations for CAT software selection. However, its conclusions are too precise, and although they have some reference value, they cannot be applied to other CAT software, and there are limitations in terms of timeliness:

$$k = \sum_{i=1} \frac{\sqrt{c+1}}{\sqrt{c^2+c}}. \tag{3}$$

3.2. Design of Computer-Aided English-Chinese Translation System. Having a large translation memory and terminology database allows the translator to do two things at once when translating. First, suppose the text to be translated has the same or similar expressions as those in the memory. In that case, the translator can directly adopt or refer to them, thus improving translation efficiency. In the algorithm, the optimization problem's solution corresponds to the bird's position in the search space, and these birds are compared to the individual "particles." These particles have position and velocity, which is used to determine the distance and direction of flight, respectively, and the optimization function determines the adaptation value. Each particle tracks the current optimal particle and searches in the solution space. Second, the translated contents can be re-imported into the memory, which can expand the capacity of the memory and provide more convenience for future translations. In the translation process of the translation project of this review, the text translated later draws on the translation memory of the text already completed earlier, and the retranslation matching degree is higher. Hence, the translation efficiency is relatively higher. In other words, the more translation memory is accumulated at a later stage, the more content there is. Then, the translation speed will be faster for the same type and size of documents. The retranslation function is an integral part of the calculation aid software to perform the memory and terminology library function. Generally speaking, the more extensive the inventory of memory and terminology libraries, the higher the matching degree of retranslation and the higher the percentage of the translation progress bar. It both reduces the burden of repeated translation and dramatically improves translation efficiency.

In the translation process of the translation project of this paper, with the help of the built memory library, the translation progress bar of the retranslation reaches up to 60%, and the lowest is 20%, which vastly improves the translation speed and efficiency and speeds up the process of this translation project. The software generally requires that the project framework to be established before importing the

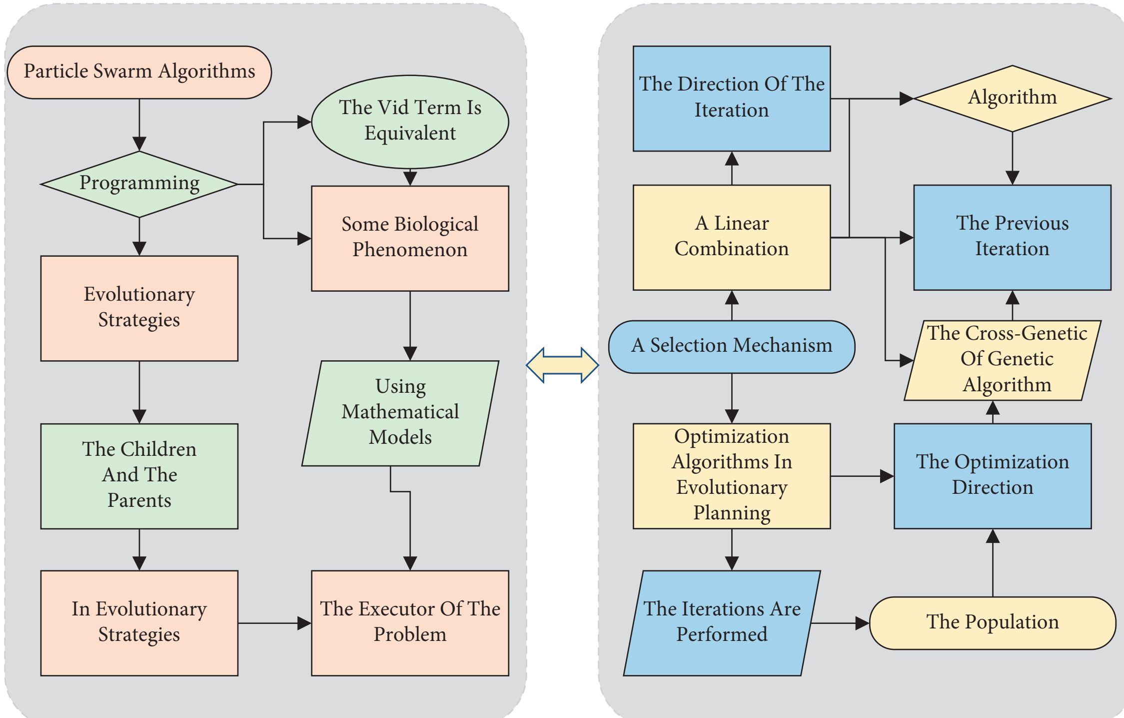


Figure 1: Particle swarm optimization process.

files to be translated. Translators can import multiple files and translate them simultaneously or translate them one by one. This helps to organize and clarify the translation project and helps to raise the translators' awareness of translation project management, and promotes teamwork. The translator establishes the translation project at the beginning of this translation project. The translation progress of each translation file is displayed on the software interface in real-time so that the translator can grasp the current translation process more clearly [22]. With more tasks and complicated steps in the retranslation stage, the translation process requires translators to invest time and energy. In particular, translators who are not familiar with the operation of the software may make different mistakes and delay the translation process due to various operational details at the early stage of using CAT software, thus lacking patience and becoming afraid of the use of computer-aided software or fearing trouble, resulting in giving up the use of computer-aided software. However, once the translators overcome these initial problems and become familiar with the operation process of the software, they will speed up the translation process and improve the translation efficiency. Therefore, the translator needs to set up a long-term vision and patiently cope with the tedious preparation work in the early stage. The experience process model is shown in Figure 2. The main parameters of the standard particle swarm optimization algorithm are the number of population sizes, the maximum number of iterations, the cynical weights, the maximum velocity of the particles, and the acceleration factor. Since there is no general and systematic theory of particle swarm algorithm, selecting particle swarm algorithm parameters mainly relies on experience.

For users of CAT software, many factors influence their choice of whether to use CAT software for translation projects within the framework of usability theory, such as subjective satisfaction, error rate. The fewer significant errors in CAT software during a translation project, the more favorable the user will feel about the software and may be inclined to continue using CAT software. Before using CAT software for translation, there must be a learning process. Although it may not be necessary to be proficient in various software functions before putting it into use, it is essential to learn at least the basic functions of each link. The absence of any association may make the translator's efforts naught. The users are often not proficient in computer technology, so if CAT software is too difficult to learn, it may affect the translator's motivation to use it. Therefore, if CAT software is too difficult to understand, it may affect the translators' motivation to use it. In addition, efficiency is also a major influencing factor. Although CAT software should help translators improve their work efficiency, due to the reality that translators have different abilities and different environments, not every translator can significantly improve their work efficiency even if they use the same software to complete the same translation project, so a higher efficiency for translators may make them more inclined to choose to use CAT software. The performance testing process of the CAT system is shown in Figure 3. Research on computer-assisted translation can be divided into three main categories: CAT theory research, CAT technology and tools research, CAT teaching research, and a small amount of research on the CAT industry.

Usually, the above usability dimensions should be measured by performing specific tasks for specific users.

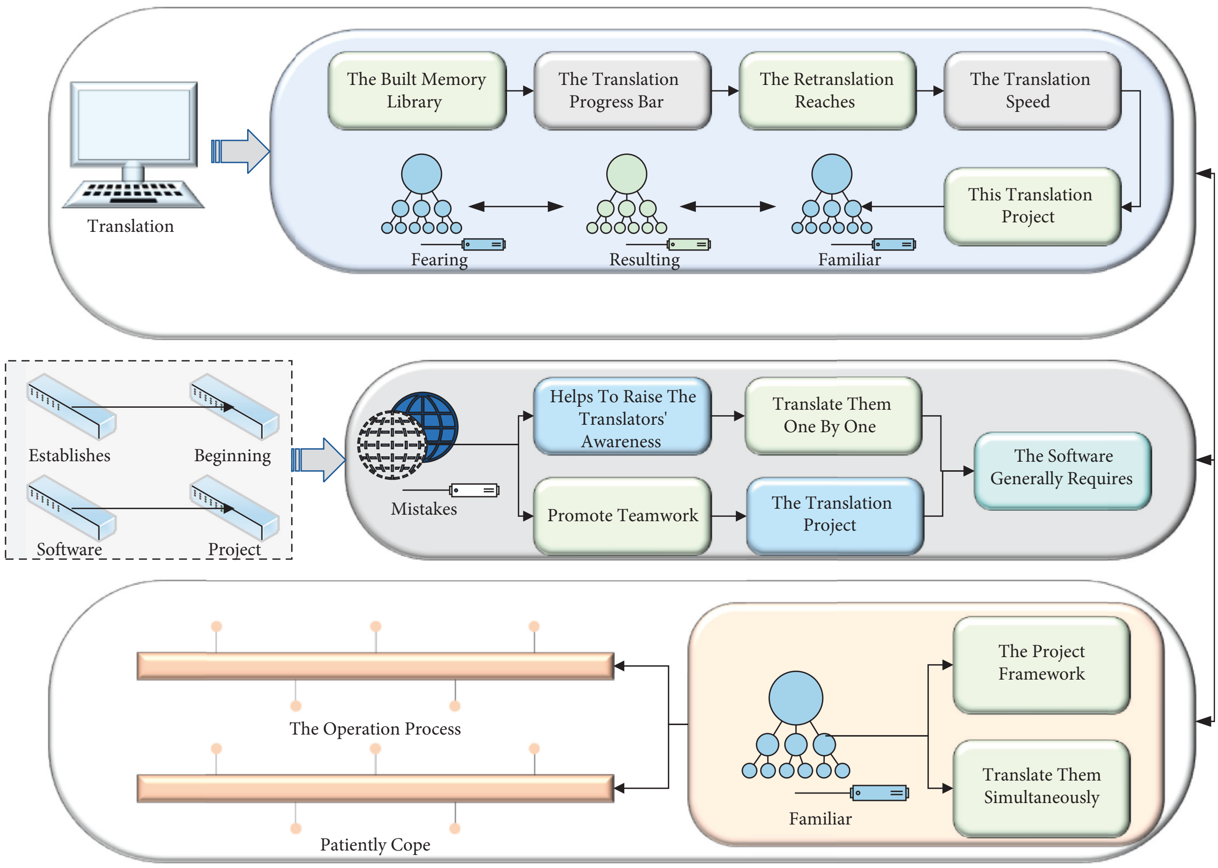


FIGURE 2: Experience process model.

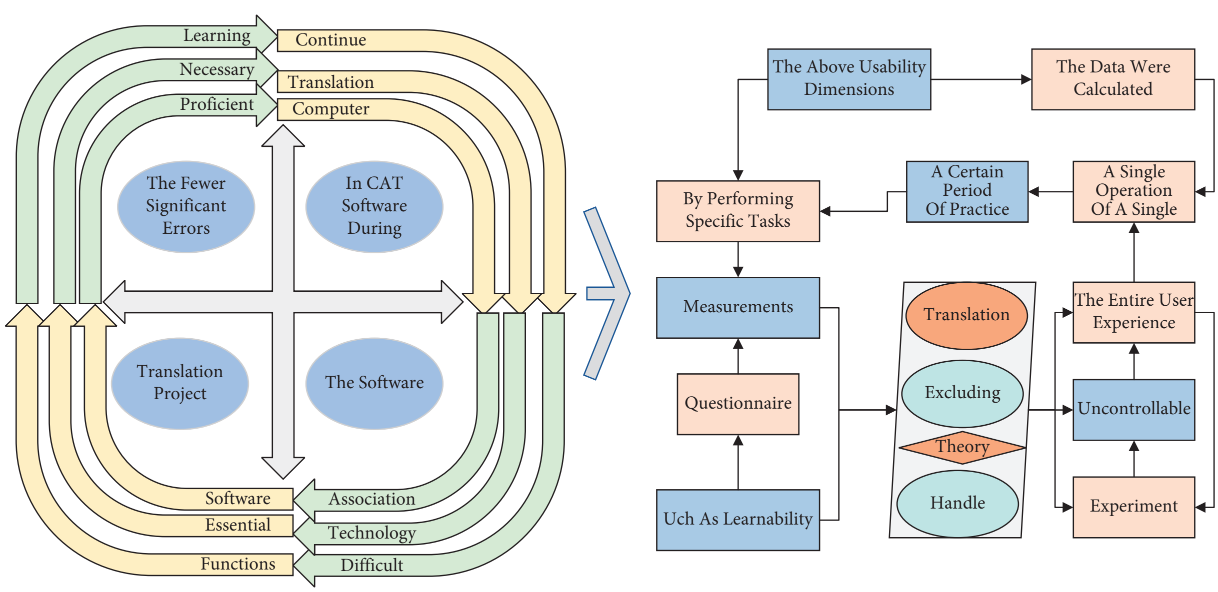


FIGURE 3: System performance testing process.

However, since measurements such as learnability and error rate can only be reflected after a certain period of practice, and a single operation of a single user cannot represent the entire user experience, there are too many uncontrollable factors in the experiment, so the author changed the format of the questionnaire to a survey. Although the questionnaire cannot obtain intuitive data like the experiment, the answers are all based on the respondents' long-term translation practice, excluding the possible investigation chance. In addition to the research on the evaluation framework of CAT software, another scholar has conducted a specific evaluation of CAT software, comparing the differences between open-source software (Pootle) and commercial software (Trados) using experiments on translation task experience. He believes that paid commercial software can help translators complete translation projects more efficiently and conveniently than open-source software. In particular, according to Nielsen's theory, learnability was measured by the time it took for users to become proficient from the time they first encountered CAT software to the time they were able to handle a translation project, and the data were calculated in months; error rate was measured by the number of significant errors in 10 translation projects, and the data were calculated in counts; efficiency was a categorical variable with five levels, that is, significantly improved, improved but not significant, uncertain, reduced but not The subjective satisfaction is measured by the Liker scale, which allows respondents to rate the CAT software they use, with higher scores representing greater satisfaction.

4. Analysis of Results

4.1. Particle Swarm Optimization Programming Model Analysis. The main parameters of the standard particle swarm optimization algorithm are the number of population sizes, the maximum number of iterations, the cynical weights, the maximum velocity of the particles, and the acceleration factor. Since there is no general and systematic theory of particle swarm algorithm, selecting particle swarm algorithm parameters mainly relies on experience. Of course, when solving complex problems with high-dimensional space search, this parameter needs to be increased appropriately, generally between 200 and 300, with the number populations. As the number of populations increases, the algorithm can find a better solution to some extent. Still, the space and time it consumes also increase accordingly. The maximum number of iterations: this parameter is one of the conditions for the algorithm to stop iterating. The common point between parallel texts is more about style than language, which is different from the principle of the computerized translation memory. In addition, the length of such texts is usually short. If the computer-assisted translation is applied, tedious retranslation preparation work is required, which will lengthen the translation process and reduce work efficiency. It is also an important parameter that affects the algorithm's ability to find the optimal solution, but the value cannot be increased indefinitely. The resource overhead will increase accordingly. Inertia weight factor: the linear inertia weight is the most mature research on this parameter. Practice shows that it is generally taken from 0.9 to 0.4 linearly decreasing, achieving good results. In the early stage of the algorithm, the value is more significant to ensure that the particle's velocity is divided into three parts, respectively, inertial acceleration, own experience, and population experience.

(1) The particle's velocity part is the particle's inertial velocity.

(2) Individual cognitive part: this is the experience of the particle itself, and this part gives the particle the ability to search locally.

(3) The social cognitive part: $r\text{and}(g\text{best} - x_i)$ the information sharing among the particles within the race prevents the algorithm from falling into the local optimum, giving the particles a robust global search capability. Based on the inertial speed of the particle, the particle slowly converges to the global optimum by continuously adjusting its position during the iterative process, combining its individual experience with the information sharing and collaboration mechanism in the race. The acceleration factors c_1c_2 are two constants that control the acceleration of all particles toward the global and individual extremes. They have smaller values to enable the particles to have the ability to explore outward as they converge to the target region, and larger values may cause them to fly past the target region.

$c_1 = c_2 = 0$ In the case of a particle flight, the particle will maintain a uniform speed at the current rate until it reaches the end of the search space because the social and individual cognitive abilities of the particle are lost at this time unless the optimal solution coincides with the particle's search path, the PSO algorithm can't find the optimal global solution:

$$v_i = wv_i^k + \sqrt{k-1}. \tag{4}$$

When, the particle has no empirical part of itself, it loses the ability to fly toward its optimal historical position. At this time, the particle has only the current velocity and the social cognitive part, which is called the global PSO algorithm, that is, machine translation technology has been developed to a considerable extent, and for texts with more fixed forms, clearer semantics, and less-demanding translation quality, machine translation can indeed meet the demand. However, there are still many bottlenecks in the semantic aspects of machine translation that are difficult to breakthrough. Although processing bullet texts is fast, the quality of the translation of different texts is still not satisfactory, and the processing of posttranslation editing is still needed:

$$v_i^{k+1} = \frac{wv_i^k + c_1}{\sqrt{g\text{best}^{k-1} - x_i^k}}. \tag{5}$$

The ability to explore new spaces through particle interactions is characterized by fast convergence and better results in some problems. However, it is challenging to jump out of the local optimum in complex issues because it has no local search possibility. When the particle has no group experience, the particle can only fly according to its own search experience and cannot pass to the optimal group position. At this time, the particle only has the current velocity and individual cognitive part, which is called the local PSO algorithm, namely: commonly used population intelligence algorithms include the Genetic Algorithm (GA), Differential Evolution (DE) algorithm, Particle swarm optimization (PSO) algorithm, etc. The optimization mechanism of these algorithms does not depend too much on the organization of the algorithm. The structural information of these algorithms does not rely too much on the organization of the algorithm, which can be widely applied to the combinatorial optimization and computation of functions, especially in solving some optimization problems with nonlinear, nonconvex, or nondifferentiable objective functions, which show good search performance:

$$v_{k+1} = \frac{wv_i^k + c_2 \text{rand}_1}{g\text{best}_i^k - x_i^k}. \tag{6}$$

The absence of inter-particle interactions and the blockage of information sharing makes the population search equivalent to a blind search by a random number of individuals at scale, with slow convergence and a minimal probability of getting the optimal global solution. Later, studies found that in order to get a better solution, it is not necessary to be equal to 2, but taken the interval [0, 4]. The study of the effect and the PSO algorithm over the time-varying process was obtained by transforming them:

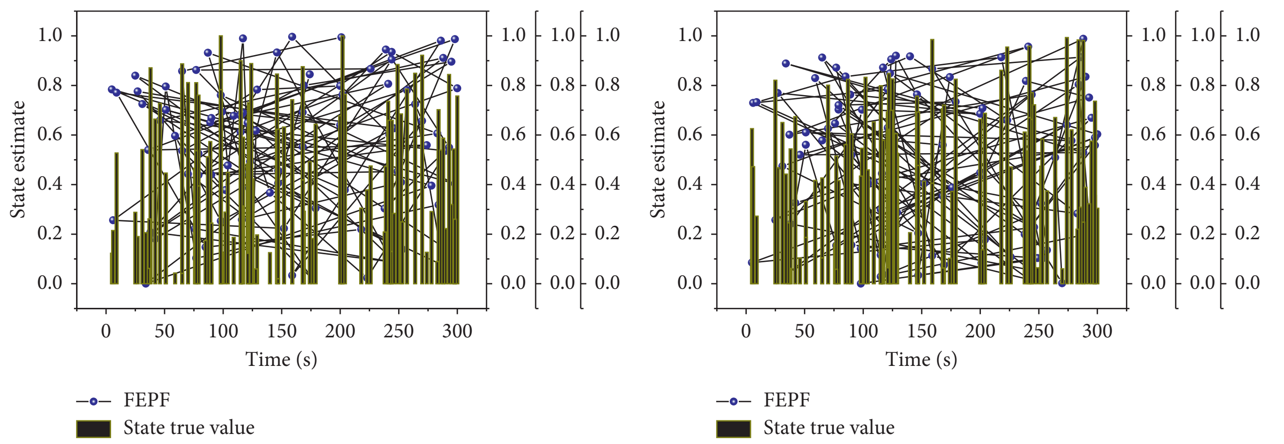
$$c_1 = \frac{(c_1 f - c_{1i})}{c_{li}(n_{\text{maxiter}} - n_{\text{iter}})}. \tag{7}$$

This linearly varying acceleration factor model enhances the algorithm's global let-go capability and can converge to the optimal global solution. In summary, in the above three parts, the first part is the inertial velocity of the particle to enable it to have an inertial motion with a tendency to expand in the search space, thus having the global exploration ability; the second part is the representative thinking about its behavior and learning its empirical knowledge, which is the self-cognitive part [23]; and the third part is inter-particle cooperation and information-sharing between groups, which is social cognition. The above three interact and constrain each other to determine the superiority-seeking ability of PSO. The estimated value of particle movement state and the estimation error can be obtained as shown in Figure 4.

4.2. Computer-Aided English-Chinese Translation System Analysis. The suitability of a text for computer-assisted translation is not determined by the field or the category of the text or the function or features of CAT software. Usually, we need to combine the two and make a comprehensive analysis, which means that a translation project analysis is required for the retranslation stage. In the case of study-abroad texts, their multiple types and formats determine both their applicability and limitations in terms of computer-assisted translation. First, as far as the classes are concerned, self-statements and letters of recommendation belong to more individual and typical texts, which require a high level of literary style and infectiousness of the language, not just grammatical and meaningful precision. Second, what such parallel texts have in common is more about style than language, which is different from the principle of the computerized translation memory. If the computer-aided translation is applied, tedious pretranslation preparation work is required, which will, on the contrary, prolong the translation process and reduce the working efficiency. CAT software plays a rather important role in the daily work of translators. Professional translators use CAT software in their work, and part-time translators, translation students, translation enthusiasts, and other groups have started to pay attention to computer-aided translation technology. The translation majors of major universities have offered courses on computer-aided translation. Even universities have set up additional majors in computer-aided translation, specializing in training this aspect of talent. Therefore, after analysis, the author believes that the advantages of using computer-assisted translation for texts such as self-statements and letters of recommendation are not obvious. However, another category of texts, such as curriculum, transcripts, certificates of registration, bank accounts, honors, school training programs, course descriptions, are repetitive, have a lot of terminologies, and have reference templates. Such texts can usually be accumulated and translated by referring to parallel texts and applying templates. Combined with the features of computer-aided translation memory and terminology database, the translation efficiency of translators can be improved to a certain extent. As far as the format is concerned, although various CAT software are being continuously optimized, the file formats that CAT software can handle are still limited, such as the noneditable PDF format and JPEG image format, which can be converted with the help of other specialized software [24]. Still, the whole translation process will be prolonged for some small-scale translation projects because of the increased preparation work before translation. Therefore, the author believes that the format and length of the text need to be considered to choose the means of translation. If the distance is long, it is best to convert text format using computer-assisted translation. If the form is unmanageable and the length is short, then a manual translation can be performed. A comparison of the three data sets is shown in Figure 5.

In using CAT software to translate study-abroad texts, I encountered many computers technical problems, tried various means to solve them by myself, looked them up on the Internet, or sought help from technicians in related fields. Later, it was found that many of the problems appeared to be operational errors, but in essence, the author lacked an understanding of the software's operating principles. Like the author, many translators are language learners and perhaps not good at computers. During the tedious operation of the software, it always stalls due to specific details of mistakes, adding to the burden of translation. After many attempts, familiarity with the software process will naturally improve the dilemma. However, to solve this lack of computer technology, translators should enhance their awareness of interdisciplinary learning, get more familiar with and understand the basic principles of computers, and improve their ability to solve computer technology problems independently.

4.3. Test Results of Computer-Aided English-Chinese Translation System. The BLEU of the model that preprocesses Chinese by characters + named entities reaches a maximum of 15.93, which is 0.40 higher than that of the model that



(a)

(b)

Figure 4: State estimates and estimation errors.

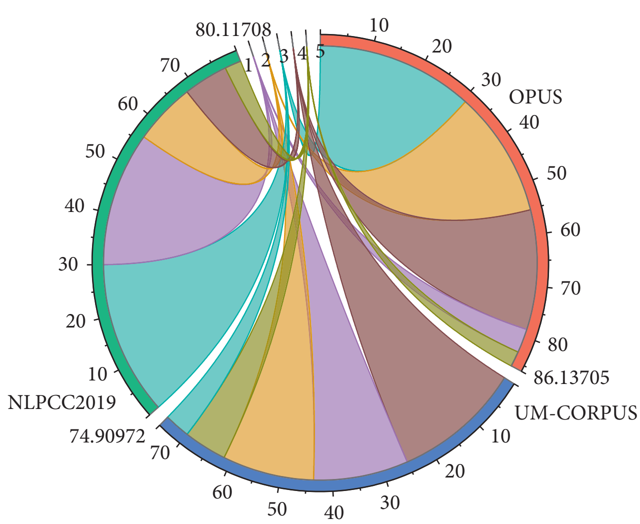


Figure 5: Comparison of the three data sets.

preprocesses Chinese by words. Considering that the BLEU is calculated according to the word sequence output method, the actual translation effect of the model may be improved more. The preprocessing of Chinese by characters + named entities reduces the magnitude of the Chinese vocabulary, reducing the number of parameters of the model. Taking the data set TED2013 as an example, when processing Chinese by word sequence, the size of Chinese word vocabulary is $V_1 = 33607$, and the total number of parameters of the Chinese embedding layer is $V_1 * d$. The total number of parameters of the SoftMax layer is $V_1 * d$, the total number of parameters is $2V_1 * d$, and the total number of parameters is 17.2 M with $d = 256$. When processing Chinese by character + named entity, the size of Chinese word vocabulary is $r2 = 5\ 647$. The total parameters of the Chinese embedding layer and SoftMax layer are $2F2 * d = 2.89$ M, which reduces the parameters by 83.2%. The software generally requires that the project framework be created before the files to be translated can be imported. Translators can import multiple files and translate them simultaneously or translate them one by one. This helps to organize and clarify the translation project and helps to raise the translators' awareness of translation project management, and promotes teamwork. The translator establishes the translation project at the beginning of this translation project. The translation progress of each translation file is displayed on the software interface in real-time so that the translator can grasp the current translation process more clearly. Considering the number of parameters in the English coding embedding, LSTM, and attention layers, the model parameters are reduced by 30–40%. At the same time, the Softmax layer is 83.2% less computationally intensive when solving the probability distribution in the forward direction and the derivation in the reverse order. The model processes 3.96 K words per second, 126% more words per second than processing Chinese with word sequences. Still, considering the increase in text length due to processing text with character sequences, the actual time spent for each batch training is 1.83 seconds, and the total time spent is about 15.2 hours. Compared with processing Chinese with word sequences, the time was reduced by 21.2%.

The first three models mentioned in this paper used initialization method one. In the data set TED2013, the experiments were conducted using the initialization method 2. The model reached a maximum BLEU of 15.61 in the test set within 30 K steps, slightly smaller than method 1. The smaller amount of data and the more extensive word list at the English end resulted in insufficient data to train a better word vector than the unsupervised Glove method. However, experiments using way 3 (model 4) showed loss function curves introduced on TED2013 and in the test set BLEU as shown in Figure 6.

This study mentions that when pretrained word vectors are used in the model, the expressiveness (compared to other values in the surrounding space) is already excellent because the word vectors are trained and converged on a large-scale corpus. Therefore, the model may quickly fall into a local optimum solution when using pretrained word vectors. This conjecture can be verified by comparing the decreasing trend of the loss curve during model training. The natural curve is

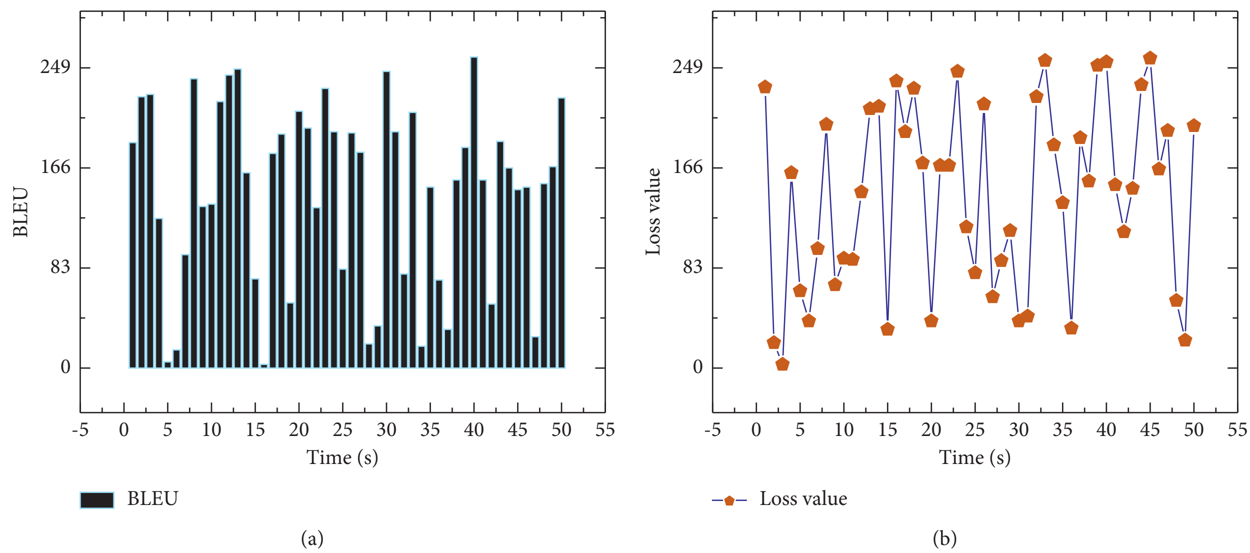


Figure 6: Loss function curve and test set BLEU.

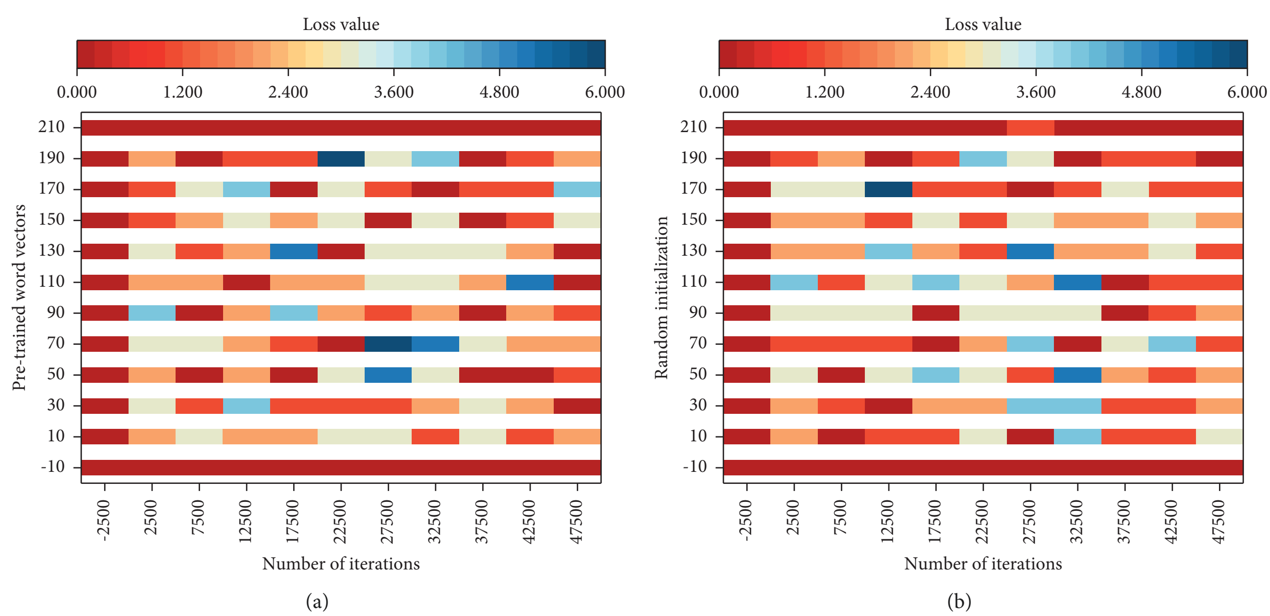


Figure 7: Comparison of the loss when trained with pretrained word vectors and random initialization in two ways.

the training error curve of the model when using the pretrained word vector. The error drops rapidly at the beginning of the movement, reaching about 55 at about 10 K steps, and then falls more slowly and stabilizes at about 50. As analyzed earlier, the loss of the model with pretrained word vectors is smaller than that with random initialization in the early training period. Still, the model with random initialization converges to better results in the later training period. The loss comparison between the pretrained word vector and the random initialization is shown in Figure 7.

This review proposes a new model structure with a conversion layer added between the encoder and decoder. The conversion layer used in this study consists of three components: a forward neural network layer, a residual connection, and a normalization layer. Using the model with the conversion layer, the maximum BLEU on the test set reaches 17.61 in 30K steps when trained on the TED2013 dataset, improving 1.03 BLEU. After adding the transformation layer, the new parameters added to the model are mainly in the forward neural network layer in the transformation layer. When using the intermediate layer with width $h = 4$ d, the RNN consists of two bidirectional LSTMs with $n = 4$ layers, and the coating contains a total of about 64d2 parameters. Considering the embedding layer (about 10 M parameters), the LSTM layer, and the attention layer (within 1M parameters) on the English side, the model increases the parameters by about 30%. The increase in computation during model training and inference is also around 30%. Model 5 processed 3.07 K words per second at TED2013, and each batch training took 2.36 seconds for about 19.6 hours. This increases by 28.9 hours compared to when processing Chinese with word sequences.

5. Conclusion

With an ever-expanding translation market and an ever-increasing translation workload, it is impossible to complete such a heavy and urgent translation task efficiently and with high quality by relying solely on translators working with their heads in the sand. Machine translation is amazingly fast, which saves time and cost significantly. The whole project is made to flow with computer-aided translation tools to reasonably calculate and allocate the translation volume and strictly control the translation quality. This paper proposes developing and researching a computer-aided English–Chinese translation system based on particle swarm optimization programming. From the perspective of translation, under the situation that translation accuracy is not required, more and more users begin to use the combination of machine translation and manual modification to replace computer-aided translation, even though the availability of CAT software has reached a reasonably high level, the process is still more complicated and tedious compared with machine translation. With the continuous development of machine translation technology, machine translation is sufficient to meet users' needs in situations that do not require high-translation accuracy. Machine translation may gradually replace computer-aided translation. Perhaps in the future, CAT software can further explore its professionalism-related functions to improve the added value of products and distinguish the application with machine translation in different fields. No matter how the technology in the translation industry develops in the future and what challenges the CAT industry faces, the development of CAT software must face up to the needs of users and improve the usability level of the software itself.

Data Availability

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

Conflicts of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by International School, Xi'an Siyuan University.

References

[1] R. Nai, "Problems and research on interpretation education in Chinese universities under the background of foreign affairs of large countries," *Converter*, vol. 2021, no. 7, pp. 387–393, 2021.

[2] N. Dhanachandra and Y. J. Chanu, "An image segmentation approach based on fuzzy c-means and dynamic particle swarm optimization algorithm," *Multimedia Tools and Applications*, vol. 79, no. 25-26, pp. 18839–18858, 2020.

[3] N. R. Zhou, A. W. Luo, and W. P. Zou, "Secure and robust watermark scheme based on multiple transforms and particle swarm optimization algorithm," *Multimedia Tools and Applications*, vol. 78, no. 2, pp. 2507–2523, 2019.

[4] G. Che, L. Liu, and Z. Yu, "An improved ant colony optimization algorithm based on particle swarm optimization algorithm for path planning of autonomous underwater vehicle," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 8, pp. 3349–3354, 2020.

[5] Y. Zhang, H.-G. Li, Q. Wang, and C. Peng, "A filter-based bare-bone particle swarm optimization algorithm for unsupervised feature selection," *Applied Intelligence*, vol. 49, no. 8, pp. 2889–2898, 2019.

[6] A. Naseri and N. Jafari Navimipour, "A new agent-based method for QoS-aware cloud service composition using particle swarm optimization algorithm," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 5, pp. 1851–1864, 2019.

[7] S. Tabibi and A. Ghaffari, "Energy-efficient routing mechanism for mobile sink in wireless sensor networks using particle swarm optimization algorithm," *Wireless Personal Communications*, vol. 104, no. 1, pp. 199–216, 2019.

[8] I. U. Rahman, Z. Wang, W. Liu, and B. M. X. Ye, "An N-state markovian jumping particle swarm optimization algorithm," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 11, pp. 6626–6638, 2021.

[9] R. A. Ibrahim, A. A. Ewees, D. Oliva, and M. S. Abd Elaziz, "Improved salp swarm algorithm based on particle swarm optimization for feature selection," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 8, pp. 3155–3169, 2019.

[10] R. Shadiev, A. Sun, and Y. M. Huang, "A study of the facilitation of cross-cultural understanding and intercultural sensitivity using speech-enabled language translation technology," *British Journal of Educational Technology*, vol. 50, no. 3, pp. 1415–1433, 2019.

[11] C. Xue, H. Xu, Y. Guo, and L. Y. H. Z. J. D. Xu, "Accurate bracket placement using a computer-aided design and computer-aided manufacturing-guided bonding device: an in vivo study," *American Journal of Orthodontics and Dentofacial Orthopedics*, vol. 157, no. 2, pp. 269–277, 2020.

[12] R. F. Alfuraih, "The undergraduate learner translator corpus: a new resource for translation studies and computational linguistics," *Language Resources and Evaluation*, vol. 54, no. 3, pp. 801–830, 2020.

[13] M. J. Blas and S. Gonnet, "Computer-aided design for building multipurpose routing processes in discrete event simulation models," *Engineering Science and Technology, an International Journal*, vol. 24, no. 1, pp. 22–34, 2021.

[14] Y. Shu, "Experimental data analysis of college English teaching based on computer multimedia technology," *Computer-Aided Design and Applications*, vol. 17, no. S2, pp. 46–56, 2020.

[15] C. Lum, C. S. Koper, X. Wu, and W. M. Johnson, "Examining the empirical realities of proactive policing through systematic observations and computer-aided dispatch data," *Police Quarterly*, vol. 23, no. 3, pp. 283–310, 2020.

[16] D. L. Guarin, Y. Yunusova, B. Taati, and J. R. S. J. M. M. E. T. A. N. Dusseldorp, "Toward an automatic system for computer-aided assessment in facial palsy," *Facial Plastic Surgery & Aesthetic Medicine*, vol. 22, no. 1, pp. 42–49, 2020.
[17] J. W. Chong, S. Thangalazhy-Gopakumar, K. Muthoosamy, and N. G. Chemmangattuvalappil, "Design of bio-oil additives via molecular signature descriptors using a multi-stage computer-aided molecular design framework," *Frontiers of Chemical Science and Engineering*, vol. 16, no. 2, pp. 168–182, 2022.
[18] C.-Y. Liao, J. B. Ganz, K. J. Vannest, and S. L. M. V. Y.-F. Wattanawongwan, "Caregiver involvement in communication skills for individuals with ASD and IDD: a meta-analytic review of single-case research on the English, Chinese, and Japanese literature," *Review Journal of Autism and Developmental Disorders*, vol. 8, no. 3, pp. 350–365, 2021.
[19] W. Su and D. Li, "Identifying translation problems in English-Chinese sight translation," *Translation and Interpreting Studies*, vol. 14, no. 1, pp. 110–134, 2019.
[20] Y. Jia, M. Carl, and X. Wang, "Post-editing neural machine translation versus phrase-based machine translation for English-Chinese," *Machine Translation*, vol. 33, no. 1-2, pp. 9–29, 2019.
[21] C. Han and Q. Fan, "Using self-assessment as a formative assessment tool in an English-Chinese interpreting course: student views and perceptions of its utility," *Perspectives*, vol. 28, no. 1, pp. 109–125, 2020.
[22] S. Chen, F. Liao, D. Murphy, and S. Joseph, "Measurement invariance of the English, Chinese, and Spanish versions of the barrett-lennard relationship inventory," *Measurement and Evaluation in Counseling and Development*, vol. 55, no. 1, pp. 30–47, 2022.
[23] S. Yu, Y. Zhang, Y. Zheng, and Z. Lin, "Written corrective feedback strategies in English-Chinese translation classrooms," *The Asia-Pacific Education Researcher*, vol. 29, no. 2, pp. 101–111, 2020.
[24] S. Pang and K. Wang, "Language contact through translation," *Target. International Journal of Translation Studies*, vol. 32, no. 3, pp. 420–455, 2020.