Hindawi
Mobile Information Systems
Volume 2022, Article ID 7795836, 10 pages
https://doi.org/10.1155/2022/7795836

Research Article

Development of Automatic English Translation System Based on Fuzzy Matching and Software Simulation

Qian Tan

Southwest Jiaotong University Hope College, Chengdu, Jintang, China

Correspondence should be addressed to Qian Tan; tanqian@swjtu.edu.cn

Received 14 May 2022; Revised 18 June 2022; Accepted 2 July 2022; Published 19 July 2022

Academic Editor: Shadi Aljawarneh

Software requirements are changeable, and the changes in requirements have led to many technical, economic, and management problems in the software development process, which are also considered to be the main source of software development risks. Based on the relationship between software requirements and design, this article analyzes the propagation process of requirement changes and the two dimensions of requirement change risk, opportunity and impact, and discusses software architecture customization that combines requirements, software architecture, and risk. On the basis of QFD and DSM, a composite relationship matrix is created according to requirements and software architecture, and two aspects of software architecture adaptability are analyzed from the perspective of the risk of changes in requirements. With the advent of the 5G era, this issue has gradually been paid attention to in its evolution. In order to further improve the intelligence level of the English translation system and improve the accuracy of English translation, this paper designs a translation algorithm that takes the changes in software requirements as the basis. On this basis, cross-compilation and multithreaded phrase translation loading methods are used to automate the translation system. System test results show that the system has high translation accuracy and good intelligence. Overall, the development of an automatic English translation system based on software change management and 5G networks will further enhance the intelligence and automation of English translation.

1. Introduction

Software requirements are an important part of software development and the basis of software design. As the source of software development, requirements define and constrain the requirements of the final software product. However, compared with conventional products, the abstraction and high complexity of software products make our demand for software products more uncertain, more unstable, more subjective, and more vague [1]. In the software development process, there will be many problems related to software requirements. Statistics about the practice of creating software projects show that most of the reasons for the failure of software projects are not technical problems in software development, but management problems; the most common of which is demand management problems [2]. Studies have shown that more than 50% of software requirements will change before the software product is officially released. In addition, more than 70% of large-scale software systems (for example, the total number of function points exceeds 1,000) will be subject to huge changes in demand during the development process. The widespread use of China's fourth-generation mobile network has further improved the quality of communication network applications and has also provided more convenience for current social development [3]. In this context, a large number of new mobile communication services continue to emerge, and people's demand for using mobile service networks has also been significantly improved [4]. By analyzing the status quo of China Mobile's technology development and its future development prospects, it can be concluded that in the future social development, 5G radio networks will gradually transform into mobile communication systems occupying a certain position [5]. Compared with the mobile networks used in the past, the advantages of 5G are mainly reflected in the transmission speed and transmission quality [6]. In addition, the stability of information transmission is higher than that of other mobile networks. In general, the 5G wireless network has broad prospects for the future, and its unique high-frequency characteristics can enable users to better understand the network bandwidth. For 5G wireless networks in the medium and low frequency range, efficient and extensive network coverage can be achieved through mergers and connections [7]. With the development of information technology, the development of English translation systems based on changes in software requirements and 5G network management has greatly improved the intelligence and accuracy of English translation [8]. The systematic use of automatic translation is the most important software medium for performing English translation. Therefore, the design of automatic English translation system has important practical significance [8]. The automatic English translation system analyzes the characteristics of English vocabulary in detail through semantic analysis and effectively combines the methods of semantic fuzzy matching and automatic phrase analysis to perform large-scale automatic translation of vocabulary, ensuring the accuracy and reliability of translation [9].

2. Related Work

The literature shows that changes in demand have led to various technical, economic, and management problems in software development and are considered to be an important source of risk in software project development. The literature shows that changes to software requirements are not limited to the collection and analysis of requirements. It will affect the entire software development life cycle and most directly affect the design of the software architecture [10]. It is clearly pointed out in the literature that it is generally believed that the key influence of analyzing changing requirements on software development is usually the software architecture. Software architecture describes the components, connections, system properties, and behaviors of a software system, and is the core of software design [11]. Traditional software development models tend to separate requirements and carry out static and mechanical design. The literature shows that with the rapid development of China's network technology and the emergence of 5G networks, the future network will gradually develop in the direction of intelligent development, diversification, and integration [12]. The literature shows that when building a 5G wireless communication system, a suitable location for installing the antenna structure should be selected according to the actual requirements of the network structure to further increase the capacity of the dimensional space [13]. The literature suggests that attention should be paid to the use of culture and language skills in translation work, only in this way can the translation truly convey the cultural connotation of the original language [14]. Through the development of an automatic English translation system based on software demand change management and 5G networks, the literature has improved the intelligence and automation of English translation [15].

3. Relevant Research on Software Requirement Change Management and 5G Network

3.1. Software Requirements Change Management. According to the reasons and possible consequences of demand changes, demand can be divided into the following four categories: (1) variable requirements closely related to the application field of software products, for example, changes in user needs, advances in software technology, adjustments to organizational strategies or guidelines, etc. (2) Sudden demand: certain demands cannot be completely or accurately defined in the demand analysis stage. In the software development life cycle, software developers and users will have a deeper understanding of software requirements and application areas as the development progresses, and feedback information will be integrated into the requirements analysis. Changes in requirements caused by such feedback information are called sudden changes. Make a request (3): indirect demand: this kind of demand is driven by the software development activity itself. When the demand is unstable or suddenly arises, the software development activity must make appropriate adjustments to adapt to these changes, leading to many new demands called indirect demand.

The traditional software development model regards requirements analysis and software architecture design as two independent steps. However, there is no significant difference between software requirements specifications and software. For example, during development and execution, there are no structured software requirements and software architecture. Significant differences: by comparing and summarizing the relevant research on the "demand and design gap" from an empirical point of view, it can be assumed that the traditional software development method that separates software requirements and design will have a serious adverse effect on the entire software development process. In addition, this article points out that the process from requirements to software architecture is still an informal model, which lacks necessary guidelines and good practices, which to a large extent leads to the fact that software products cannot ultimately satisfy users. Requirements: this article analyzes the impact of software architecture on requirements confirmation from two perspectives: (1) how does the software architecture affect the confirmation of claims; (2) what aspects of the software architecture will affect the confirmation of claims? Research shows that nearly 60% of software requirements verification depends on the existing software architecture. Various aspects of the nine software architectures will affect the verification of software requirements, but different aspects have varying degrees of impact, of which the nonfunctional aspects account for 29% of the total exposure. The analysis results provide software developers with reference values for the requirements and processes of software architecture, software architecture design, software development plans, and risk management.

The initiator of software architecture must prove the impact of software requirements changes on software development. To evaluate the impact of requirements changes on software architecture design, it is necessary to start with the relationship between software requirements and software design. Software requirements are based on the knowledge and understanding of the real world by users or other relevant parties. The inconsistency between knowledge and information usually exists in the objective world, and there may be some contradictions in the requirements. These requirements are derived from information that may conflict with each other. Summary: Since requirements need to be expressed in spoken language, users often ignore some implicit requirements, functional requirements, and nonfunctional requirements. Different types of requirements (such as external interface requirements and requirements) and the methods to focus on and capture these requirements are different. Due to the excessively detailed requirements, the development schedule of the software project is delayed, and the cost is exceeded. Requirements depend on the software development process and are related to it. When requirements change, it is difficult to effectively analyze and evaluate the magnitude of its impact.

The software design process is based on the specification of software requirements (the description of the information, function, and behavior of the software system) to complete the software architecture design, data structure design, and software process design or architecture. This clear component dependency is the basis for analyzing change propagation. DSM separation also provides the convenience of disassembling and reorganizing the software architecture. The use of DSM to assess the risk of change propagation can be divided into three main steps:

(1) Create the component DSM of the target software architecture and describe the software design with dependencies between components;
(2) Use relevant data to analyze the influence relationship between components to obtain a risk model for spreading changes;
(3) Based on the propagation risk model, evaluate the propagation risk of changes in software components and architecture.

The risk of change propagation r is the product of the probability of change propagation and the probability of change impact, which is given by:

$$r = cpp \times im. \quad (1)$$

The possibility that the component C_i will be directly changed when the component is changed, and the possibility of the influence of the change means that the component is directly passed to the C_i component. Therefore, the risk propagated from the component to C_i is the product, as in the following formula:

$$r_{ij} = r\left(C_j, C_i\right) = cpp_{ij} \times im_{ij}. \quad (2)$$

Change propagation risk defines the single-layer risk of change propagation (direct risk). For two components C_i and C_j that cannot be directly accessed, the risk of a step change of m from component C_j to component C_i (indirect risk) means that the component changes the influence is passed m times through other m−1, and the component C_i is as in formula (3).

The cumulative risk of change propagation cr is the sum of the risk of direct change propagation and the risk of all indirect change propagation, which is given by:

$$cri_j = \begin{cases} \sum_{m=1}^{\infty} r_{ij}^{(m)}, & i = 1, 2, \ldots, n, j = 1, 2, \ldots, n, i \neq j, \\ 0, & i = 1, 2, \ldots, n, j = 1, 2, \ldots, n, i = j. \end{cases} \quad (3)$$

The degree of influence f is the sum of the direct risk of a specific component propagating changes to all other components and the indirect risk of the change propagating, which is given by:

$$f_j = \sum_{i=1}^{n} cr_{ij}, \quad j = 1, 2, \ldots, n. \quad (4)$$

The degree of influence e is the sum of the direct risk of a specific component C_i propagating changes through all other components and the indirect risk of propagating changes, which is given by:

$$e_i = \sum_{j=1}^{n} cr_{ij}, \quad i = 1, 2, \ldots, n. \quad (5)$$

Among them is formula (6)is the following formula:

$$\mathrm{TR} = \sum_i \sum_j cr_{ij}, \quad i = 1, 2, \ldots, n, j = 1, 2, \ldots, n. \quad (6)$$

According to formulas (2)–(4), the DSM model of the spread risk of software architecture changes, and cumulative changes is obtained.

The DSM diffusion risk model is shown in Figure 1.

Using formulas (4) and (5), calculate the value of the influence f of each component, the influence e and fe+, as shown in Table 1.

Using influence f as the horizontal axis and influence e as the vertical axis, draw a scatter chart to understand the risk of component change propagation in the target software architecture, as shown in Figure 2. The more other components it involves, or the greater the impact of the change activities they involve, the greater the risk of software development.

Take the change trend of the influence degree e of the component C as an example, the change is based on the number of propagation iterations of the change. As shown in Figure 3, in the four iterations of the propagation of the change, the influence degree of the component C increases significantly and then tends to decrease, minimizing the feedback information design module or reducing the occurrence of multiple propagation iterations between components.

Based on the risk assessment algorithm for software development costs and duration, this article imitates the

	A	B	C	D	E	F	G	H	I	J	K	L
A		0.23										
B					0.27	0.23						
C				0.58			0.22	0.11		0.05		
D			0.22				0.02	0.14				
E		0.27				0.18						
F			0.04				0.09		0.50	0.11		
G			0.05					0.07			0.50	0.11
H			0.05				0.08					
I										0.06		
J												
K										0.04		0.38
L												

(a)

	A	B	C	D	E	F	G	H	I	J	K	L
A		0.25	0.00	0.00	0.07	0.08	0.01	0.00	0.04	0.01	0.01	0.00
B			0.02	0.01	0.30	0.31	0.04	0.01	0.17	0.05	0.02	0.01
C				0.71			0.30	0.27		0.07	0.17	0.11
D			0.28				0.11	0.21		0.02	0.06	0.04
E		0.30	0.02	0.01		0.27	0.03	0.01	0.15	0.04	0.02	0.01
F			0.05	0.03			0.10	0.02	0.50	0.14	0.06	0.04
G			0.07	0.05				0.09		0.03	0.51	0.33
H			0.07	0.05			0.10			0.01	0.06	0.04
I										0.05		
J												
K										0.04		0.38
L												

(b)

Figure 1: DSM model of spreading risk and cumulative spreading risk. (a) Risk of change propagation. (b) Accumulated change propagation risk.

Table 1: The *f*, *e* and *fe+* values of each component.

	A	B	C	D	E	F	G	H	I	J	K	L
f	D	0.604	0.568	0.951	0.413	0.728	0.771	0.678	0.967	0.513	1.008	1.059
e	0.53	1.031	1.812	0.797	0.951	1.059	1.201	0.356	0.06	0	0.46	0
fe+	0.53	1.635	2.38	1.748	1.363	1.786	1.971	1.032	1.026	0.513	1.468	1.059

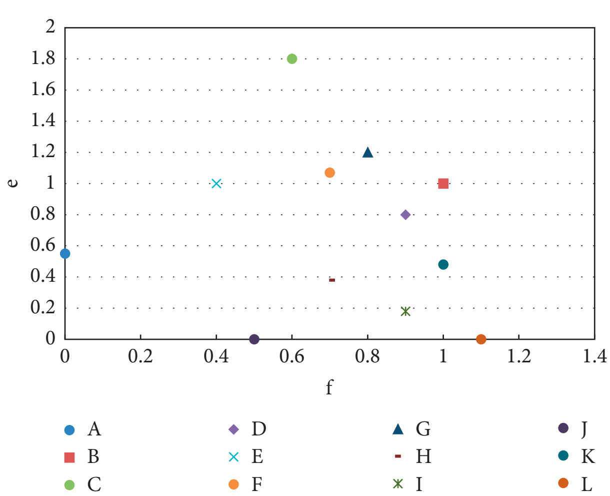


Figure 2: Scatter plot of change propagation risk.

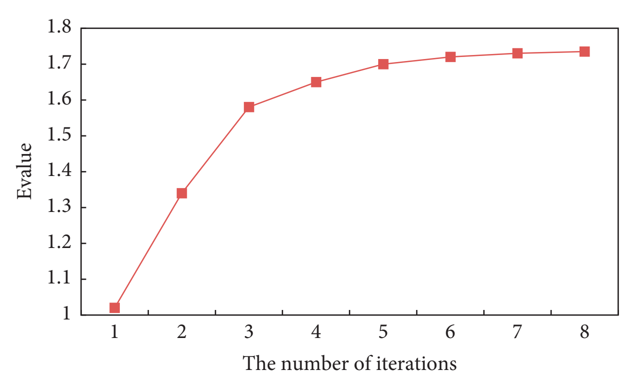


Figure 3: The degree of influence of the C component changes as the number of iterations increases.

DSM risk model to propagate changes. The cost and duration of software development are in thousands of yuan and days. Table 2 shows the input data used for simulation calculations.

The DSM model after modeling split, the probability distribution function and cumulative distribution function of the software development cost, and the simulation result of the construction time are shown in Figure 4.

To illustrate the impact of different design iterations on the cost and time required to develop software, this article also simulates DSM to assess the risk of change propagation. On the basis of the original software architecture, the order of the components was adjusted to obtain the new software architecture order {A, B, E, F, I, K, G, H, C, D, J, L} and the duration simulation in order to compare the impact of software architectures in different orders on the cost and duration of software development. Table 3 shows the simulation results of three groups of software architectures with different component layouts.

Table 2: Analog input data.

Time (days)				Cost (thousand yuan)			
	2HTF minimum	Possible value	Max	Minimum	Possible value	Max	LP
A	5	8	10	7	8	12	0.8
B	9	10	12	8	9	13	0.6
C	2	4	6	4	6	8	0.2
D	7	8	10	10	11	13	0.7
E	12	15	18	41	45	50	1
F	5	8	10	8	9	10	0.9
G	8	10	12	5	8	10	0.5
H	14	16	19	14	20	28	0.4
I	3	5	6	3	5	6	0.6
J	9	10	11	11	14	15	0.7
K	13	15	18	24	28	30	0.5
L	6	8	10	7	9	11	1

Changing the parameter requirements is the source of the risk of changing the software requirements. If the requirements change, this will directly affect the components related to the changed requirements. By mapping the relationship between requirements and software architecture in this article, the QFD requirements model and software architecture can be obtained. Due to the changes in software requirements, the changes in component design in the T period are as follows:

$$re(t) = [xq(t) \times \text{QFD}]^T. \tag{7}$$

When a component changes, in addition to its own changes, it will also cause the risk of changes to other components. Therefore, the parameter A in the dynamic model must contain these two attributes, which is given by:

$$A = \text{CR} + E. \tag{8}$$

In the t-th stage, the correction coefficient of the state Sit of the component C_i is given by:

$$S_i^t = \frac{1}{5}\sum_{m=1}^{5} s_{im}^t. \tag{9}$$

Determine the state adaptation matrix $B(t)$ of the target software architecture at stage t using the following formula:

$$B(t) = \left[b_1^t, b_2^t, \ldots, b_n^t\right] = \left[S_1^t \times a_1, S_2^t \times a_2, \ldots, S_n^t \times a_n\right]. \tag{10}$$

Select l samples $X_i(t)$ multiple times and use Monte Carlo simulation to obtain a set of independent outputs from $Y_i(t)$. Therefore, we can obtain the distribution of this set of output values using the following formula:

$$\begin{aligned} f_{5i}^t(y_i) &= E[Y_i] = E[y_i(X_i(t))] \\ &= \lim_{I\longrightarrow\infty} \frac{y_{i1} + y_{i2} + \cdots + y_{iI}}{I}. \end{aligned} \tag{11}$$

The average value of all items in the set is the expectation that exceeds a certain risk level β, which is consistent with the definition of conditional expectation, which is given by:

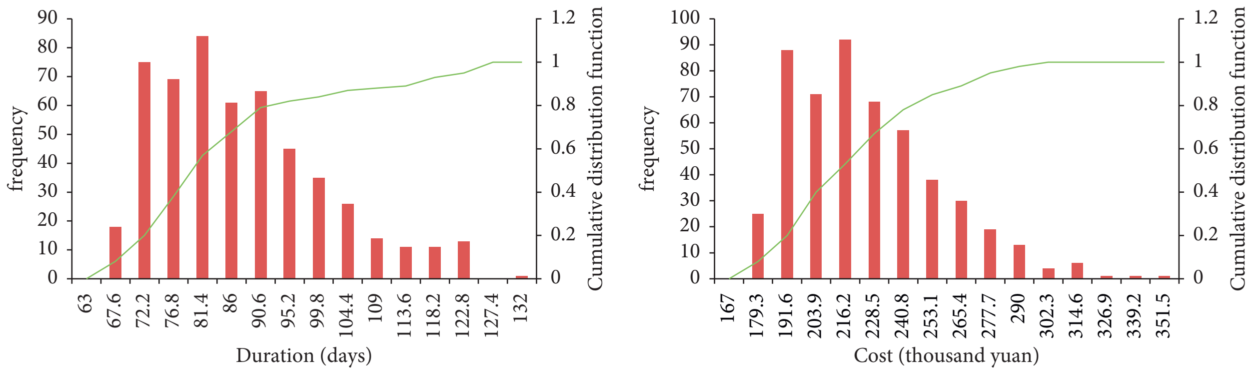
$$\begin{aligned} f_{4i}^t(y_i) &= E[y_i(X_i(t))|y_i(X_i(t)) > \beta] \\ &= E\left[Y_\beta\right] \cong \frac{Y_\beta^{(1)} + Y_\beta^{(2)} + \cdots + Y_\beta^{(k)}}{k}. \end{aligned} \tag{12}$$

In addition, it mainly analyzes the differences, advantages, and disadvantages of different change management strategies compared with the "cost risk" weighing factor "λ." Formula (13) determines the trade-off factors of the strategy implementation cost and the value change risk between strategy a and strategy b in stage t under extreme change conditions. It represents the management cost required to reduce the risk of unit change costs when faced with extreme changes between different strategies.

$$\lambda_{f4|ab}^t = -\frac{\Delta f_1^t}{\Delta y(t)} = -\frac{f_1^t|_{a,f4} - f_1^t|_{b,f4}}{y(t)|_{a,f4} - y(t)|_{b,f4}}. \tag{13}$$

Determine the cost/risk ratio between the various strategies below the unconditional expected value. The cost–risk trade-off can be applied to a single component or a single component, which can also be applied to the entire software architecture. In the fourth step of the decision-making process, Tables 4 and 5 list the trade-off factors of the overall software architecture between the four groups of change management strategies and conditional expectations.

3.2. Network System. When establishing a 5G wireless communication system, it is necessary to select a suitable location for installing the antenna structure according to the actual needs of the network structure to further increase the space capacity. If the technology can be applied to the actual construction of 5G wireless networks in the future, it will help improve the broader functions of 5G networks. The application status of MIMO technology is being gradually introduced nationwide. However, theoretically speaking, the more antenna structures installed, the better the quality of information transmission can be guaranteed, and certain guarantees are also provided in terms of security. At present, manufacturers such as Huawei and ZTE have developed 16 (T/R), 32 (T/R), 64 (T/R), and 128 (T/R) AAU equipment. More than 20 provinces have gradually launched pilot projects, and the project has completed the construction of 5G radio networks using AAU equipment. In this context, the continuous introduction of MIMO technology into 5G construction has also brought many new challenges: First, if 5G construction is carried out based on the currently available location addresses, how to maximize economic benefits and what measures should be taken? The vision to solve existing problems is not satisfactory. Second, compared to the low frequency range currently used in China, it is 3.5 GHz. Frequency resources may not meet the requirements of large-scale operations, and the operational requirements of operators also pose challenges to network connections. In addition, before the development of the 5G



(a) (b)

Figure 4: DSM's development cost and construction time modeling results have the risk of spreading changes.

Table 3: The results of modeling the cost and duration of the software architecture for different layouts of components.

	Cost (thousand yuan)		Duration (days)	
	Average value	Standard deviation	Average value	Standard deviation
Initial architecture	227.2	30	92	18
Partitioned architecture	217.5	30	84	12
New permutation architecture	240.3	45	104	13

Table 4: Factors for the compromise of conditional expectations among the four groups of strategies at level 4.

	Strategy one	Strategy two	Strategy three	Strategy four
Strategy one	—	0.5183	0.0925	0.1796
Strategy two	0.5122	—	−0.2804	−0.0222
Strategy three	0.0944	−0.2803	—	0.5002
Strategy four	0.1797	−0.0282	0.5705	—

network, the 4G network has been in a dominant position in the communication field for a long time, and this situation cannot fully occupy the existing 4G frequency resources. Third, the 5G downlink must be based on multiple antennas, and the transmission power must be increased to achieve wide coverage. However, in this case, it has a corresponding effect on the number of terminal antennas and transmit power of the uplink device. In this case, in order to increase the coverage of the downlink to achieve the construction goal, it is necessary to appropriately manage the coverage area. However, due to the current structure of the 4G website, the distance between the more concentrated regional stations has changed and is mainly controlled within 300 m. How to take effective measures? This measure further enhances the effect of information transmission by reducing the coverage area and reducing the space and at the same time increasing the number of wireless nodes to be established according to the demand of the network, which has certain difficulties.

4. Research on the Design of English Translation System

4.1. Technical Research on English Translation. Domestication translation technology is a method of transforming certain cultural elements into content familiar to the target language audience. The domestication translation method takes the target language as the target and interprets the same information and the same meaning from different perspectives of the source language. This translation method can help readers feel the similarities and differences in cross-cultural language communication, appreciate the charm of different languages and cultures, and make the translation better adapted to readers' reading, expression, and appreciation habits. For example, if there are major differences in translation practice in order to better convey the meaning of the source language while respecting the reading habits of readers in the target language, then the translation and the original text can achieve the same effect, certain components, or product templates After a special conversion, it can also be naturalized into the source language.

Different languages have developed in their cultural environment for a long time, gradually forming unique cultural expressions. If only one word is translated, not only the target effect of the translation cannot be achieved but there may even be contradictory expressions, which affects cultural transmission. Using different languages as carriers can express different meanings of the same psychological content. This way of expression does not mean that they have the same meaning in terms of form, connotation, and context. It is also a way of conveying the same information in the original language. Languages that are not smoothly translated from the front can be translated from the opposite perspective and vice versa. By analyzing the benefits, acceptability, and readability of the target language, and not using reserved forms, effective translation results can be achieved. The voice of foreigners is based on the cultural background of the source language, divided into

Table 5: Total cost, total risk, and total cost of strategy implementation of software architecture changes.

Strategy		Change cost risk > 0 (unit: yuan)				Strategy execution cost person (unit: yuan)			
		$Z = 1$	I = 2	I = 3	I = 4	I = 1	I = 2	I = 3	I = 4
One	F1	1781	1155.3	42068	13669.8	279.4	559.8	839.4	1119.6
	F2	297.0	1766.1	6437.0	20500.1	333.7	667.2	1001.6	1335.1
Two	F3	168.2	984.8	3312.3	9743.7	306.1	788.1	1545.3	3042.6
	F4	2814	1540.7	5254.9	15431.8	366.4	970.1	1988.6	39624
Three	F5	61.8	457.4	1652.1	5516.0	438.8	907.1	1435.5	1849.5
	F6	132.9	849.0	2999.6	9613.6	532.6	1129.5	1820.0	2331.0
Four	F7	62.0	347.7	10492	30665	4384	10023	1714.6	26749
	F8	133.0	690.0	2242.2	7164.0	533.0	1269.4	2262.3	3728.5

approximate forms, and then converted into the target language. This foreign language translation method can effectively convey the meaning and image of the original text, realize direct language and cultural exchanges, and reflect the appearance of the original text. It not only retains the meaning of the source language but also reflects the characteristics of the source text in the target language.

4.2. Design of English Translation System Based on 5G Background. This paper uses semantics to preprocess the English translation function based on semantic information to form an English phrase tree. The specific stages of the technical route are as follows: selection of word attributes, syntactic and semantic functions, training functions for forming decoded sentences, testing of decoded sentences, and output testing. Results: the words and grammar are aligned, the functions of the language part are marked according to the alignment, and the output becomes the attributes of the English phrase tree node. The technical route is shown in Figure 5.

During the operation of the system, the Linu platform was selected as the operating environment, and an English translation system was developed through MoseS. When phrases are used in the *G*. Der-side word corpus, the phrase length is explicitly set to 7.

The English automatic translation system is mainly composed of two parts: algorithm and software. The system software design is based on an embedded environment, which includes vocabulary collection, information processing, vocabulary planning, automatic control, and other modules. In order to extract the information that can reflect the characteristics of the distribution rules of the system, a method of combining information and intelligent planning is selected to realize the intelligent control and planning of the system. Design and develop automated translation system software based on embedded ARM environment, use Ti-ny0S to design the interface of system network components, and use Linux kernel to manage the cross-compilation of information management system software to enhance system management intelligence. The main function of the information management system of the automated translation system is to collect, merge, transmit information, plan integration, and so on. The FIFORAM buffer comprehensively analyzes the nature of fuzzy matching and the information about the translation status. The upper computer module is the carrier for remote transmission of the control information of the control system as shown in Figure 6.

(1) The realization of the semantic ontology modeling of the automatic English translation system is mainly through the organic combination of English translation and semantic feature analysis to develop related algorithms to realize the semantic ontology modeling of the system. Use the semantic feature extraction method to appropriately map and design the concept grid presented in translation and obtain the fuzzy inference related parameters of the system parametersusing the following formula:

$$\Delta((\beta) = \begin{cases} s_k, & K = \text{round}(\beta), \\ a_k = \beta - k, & a_k \in [-0.5, 0.5]. \end{cases} \quad (14)$$

Establish a robust semantic scoring index, use the method of logical fuzzy argumentation to construct the semantic tree of the concept, and obtain the model of the semantic ontology of the system. According to this model, a decision is made on the cross and complex assessment of English intelligence and automatic translation, and the following formulas are obtained:

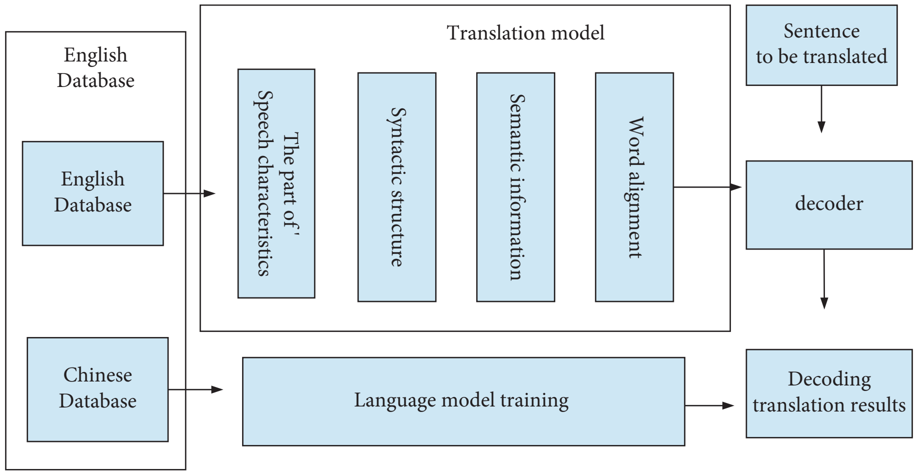


Figure 5: The technical route of the translation system.

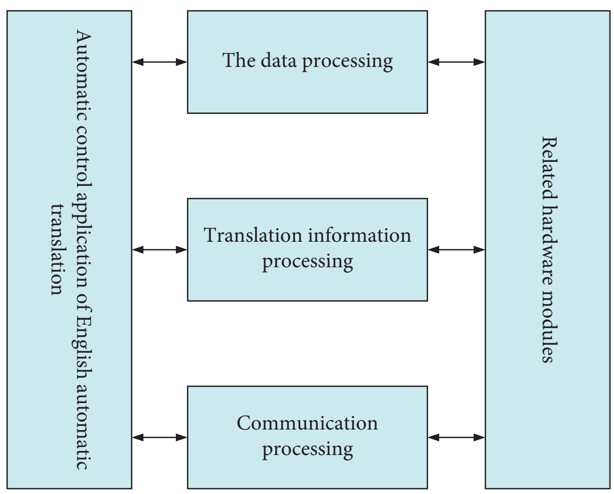


Figure 6: Information transmission model with three-tier architecture.

$$\left(\xi_{ij}^{+}, \eta_{ij}^{+}\right) = \Delta\left(\frac{\min_i \min_j \Delta^{-1} d\left(t, f^{+}\right)}{\Delta^{-1} d\left(t, f^{+}\right) + \rho \max_i \max_j \Delta^{-1} d\left(t, f^{+}\right)} + \frac{\rho \max_i \max_j \Delta^{-1} d\left(t, f^{+}\right)}{\Delta^{-1} d\left(t, f^{+}\right) + \rho \max_i \max_j \Delta^{-1} d\left(t, f^{+}\right)}\right), \tag{15}$$

$$\left(\xi_{ij}^{-}, \eta_{ij}^{-}\right) = \Delta\left(\frac{\min_i \min_j \Delta^{-1} d\left(t, f^{-}\right)}{\Delta^{-1} d\left(t, f^{-}\right) + \rho \max_i \max_j \Delta^{-1} d\left(t, f^{-}\right)} + \frac{\rho \max_i \max_j \Delta^{-1} d\left(t, f^{-}\right)}{\Delta^{-1} d\left(t, f^{-}\right) + \rho \max_i \max_j \Delta^{-1} d\left(t, f^{-}\right)}\right). \tag{16}$$

(2) The design of the translation algorithm optimizes the design of the translation algorithm through the combination of feature semantic analysis and phrase translation and develops and plans the combination of phrase translation. The specific matrix expressions are given by:

$$X_1 = \begin{bmatrix} M.G.P.P \\ P.VP.M.P \\ G.M.G.EP \\ VG.P.P.G \\ EG.EP.VP.M \end{bmatrix}, \quad (17)$$

$$X_2 = \begin{bmatrix} P.M.VP.VP \\ VP.EP.G.G \\ M.G.P.VP \\ EG.VP.VP.M \\ P.VP.M.VP \end{bmatrix}, \quad (18)$$

$$X_3 = \begin{bmatrix} G.P.VP.VG \\ VP.G.P.G \\ VG.VP.G.P \\ G.VG.EG.VP \\ P.VP.M.VP \end{bmatrix}, \quad (19)$$

$$\omega = \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{pmatrix} = \begin{bmatrix} M.G.VP.P \\ VP, VG, P, G \\ G.P.M.G \end{bmatrix}. \quad (20)$$

Realize predictive and automatic translation based on the relevance between phrases and obtain the dual semantic relevance of phrase translation combination as shown in the following formulas:

$$d_i^+ = \sqrt{\sum_{j=i}^{m} \Delta^- \left(\omega_j', \beta'\right) \Delta^{-1} \left(d\left(\left(x_{ij}, a_{ij}'\right), \left(x_i^{\prime +} + a_i^{\prime +}\right)\right)\right) 2}, \quad (21)$$

$$d_i^- = \sqrt{\sum_{j=i}^{m} \Delta^- \left(\omega_j', \beta'\right) \Delta^{-1} \left(d\left(\left(x_{ij}, a_{ij}'\right), \left(x_i^{\prime -} + a_i^{\prime -}\right)\right)\right) 2}. \quad (22)$$

The dimensionless processing of translated phrases is a specific process, which is given by:

$$D_i^+ = \frac{d_i^+}{\max_i d_i^+}, D_i^- = \frac{d_i^-}{\max_i d_i^-}, \quad i = 1, 2, \ldots, n, \quad (23)$$

$$R_i^+ = \frac{g_i^+}{\max_i g_i^+}, R_i^- = \frac{g_i^-}{\max_i g_i^-}, \quad i = 1, 2, \ldots, n. \quad (24)$$

According to the draft algorithm, combined with English translation, the optimization of the English automatic translation algorithm is realized. Cross-compilation and program loading methods are used to improve the algorithms to be loaded into the system information processing module. Based on software development, it further promotes the combination of machine intelligent translation and phrase translation.

This paper conducts a simulation experiment to further test the performance of the automatic English translation system based on the combination of machine intelligent translation and phrase translation. The accuracy of translation output and the memory speed of semantic information in English are selected as the test results. System tests show that the accuracy and recall speed of the system translation are high, and the degree of intelligence and automation is also high.

5. Conclusion

This article is mainly based on the early stage of software project development (including demand analysis and software architecture design). In the context of demand change, starting from the meaning of demand change, it discusses the adaptability of software architecture and the risk of software demand change. As we all know, in the past, network architecture systems based on traditional models could not effectively meet current social development and people's diverse needs for communication network services. We can also see that accelerated promotion and research and development of mobile networks are the inevitable trend of future social development. However, it is believed that with the advancement of current technology, the above-mentioned problems can be well solved through more active research and development. In general, the development of an automatic English translation system based on software change management and 5G networks will further enhance the intelligence and automation of English translation. In addition, due to the differences in professional experience and knowledge structure between users and software developers, it is often difficult to obtain complete and accurate requirements. The system design is mainly composed of two parts, namely the development of the software system and the development of the English translation algorithm. The automatic translation algorithm optimization project is based on the combination of semantic feature analysis and phrase translation, and the software design of the automatic translation system is based on an embedded environment.

Data Availability

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

Conflicts of Interest

The author declares that there are no conflicts of interest.

References

[1] V. Szalvay, "An introduction to agile software development," *Danube Technologies*, vol. 3, 2004.

[2] T. E. Fægri and N. B. Moe, "Re-conceptualizing requirements engineering: findings from a large-scale, agile project," in *Proceedings of the Scientific Workshop Proceedings of the XP2015*, pp. 1–5, Helsinki, Finland, May 2015.

[3] J. G. Andrews, S. Buzzi, W. Choi et al., "What will 5G be?" *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1065–1082, 2014.

[4] A. Gupta and R. K. Jha, "A survey of 5G network: architecture and emerging technologies," *IEEE Access*, vol. 3, pp. 1206–1232, 2015.

[5] M. Agiwal, A. Roy, and N. Saxena, "Next generation 5G wireless networks: a comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 3, pp. 1617–1655, 2016.

[6] N. Shahriar, S. Taeb, S. R. Chowdhury et al., "Reliable slicing of 5G transport networks with bandwidth squeezing and multipath provisioning," *IEEE Transactions on Network and Service Management*, vol. 17, no. 3, pp. 1418–1431, 2020.

[7] P. Wang, Y. Li, L. Song, and B. Vucetic, "Multi-gigabit millimeter wave wireless communications for 5G: from fixed access to cellular networks," *IEEE Communications Magazine*, vol. 53, no. 1, pp. 168–178, 2015.

[8] J. Sangeetha and S. Jothilakshmi, "Speech translation system for English to dravidian languages," *Applied Intelligence*, vol. 46, no. 3, pp. 534–550, 2017.

[9] D. Xiong, F. Meng, and Q. Liu, "Topic-based term translation models for statistical machine translation," *Artificial Intelligence*, vol. 232, pp. 54–75, 2016.

[10] S. McGee and D. Greer, "Towards an understanding of the causes and effects of software requirements change: two case studies," *Requirements Engineering*, vol. 17, no. 2, pp. 133–155, 2012.

[11] E.-M. Schön, D. Winter, M. J. Escalona, and J. Thomaschewski, "Key challenges in agile requirements engineering," in *Lecture Notes in Business Information Processing*, pp. 37–51, Springer, Berlin, Germany, 2017.

[12] Y. Wang, Q. X. Chen, N. Zhang, C. Feng, F. Teng, and M. Y. Sun, "Fusion of the 5G communication and the ubiquitous electric Internet of Things: application analysis and research prospects," *Power System Technology*, vol. 43, no. 5, pp. 1575–1585, 2019.

[13] N. K. Kiem, H. N. B. Phuong, Q. N. Hieu, and D. N. Chien, "A novel metamaterial MIMO antenna with high isolation for WLAN applications," *International Journal of Antennas and Propagation*, vol. 2015, 9 pages, Article ID 851904, 2015.

[14] H. Zhu, L. Jing, and Y. Wang, "Knowledge representation and semantic inference of process based on ontology and semantic web rule language," *Transactions of Nan Jing University of Aeronautics & Astronautics*, vol. 34, no. 1, pp. 72–80, 2017.

[15] S. Ye and W. Guo, "Semi-supervised neural machine translation based on sentence-level BLEU metric data selection," *Pattern Recognition and Artificial Intelligence*, vol. 30, no. 10, pp. 937–942, 2017.