INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 16, Issue: 3, Month: June, Year: 2021
Article Number: 4170, https://doi.org/10.15837/ijccc.2021.3.4170

Construction Method of Tender Document Based on Case-based Reasoning

T.T. Zhou, X.D. Gao, G.Y. Wei

Tingting Zhou, Xuedong Gao*, Guiying Wei
School of Economics and Management
University of Science and Technology Beijing, China
30 Xueyuan Road, Haidian Distict, Beijing 100083, China
*Corresponding author: gaoxuedong@manage.ustb.edu.cn

Abstract

The core activities of tender documents compilation are to collect similar historical tender documents, select compilation templates of tender documents and revise templates of tender documents partially. However, when the historical tender documents have accumulated to a certain amount, it becomes extremely difficult for compilers to summary, reuse and revise templates artificially in traditional compiling methods. Based on case-based reasoning (CBR), this paper studied the content recommendation method in the process of tender document construction. Firstly, a structured model of tender documents was constructed, and similar tender cases were retrieved from the tender case database according to the characteristics of tender cases; Secondly, the non-interference sequence index was used to measure the similarity of clauses used in similar tender cases, and the recommended sequences of reference template and content module of tender documents were constructed, which realized the recommendation of compiling templates of tender documents and partial revision of templates; Finally, the knowledge of the new tender case was updated. The empirical analysis shows that the construction method of tender documents based on case-based reasoning not only proposes a suitable strategy for compiling tender documents, but also improves the compilation efficiency of tender documents.

Keywords: compilation of tender documents, case-based reasoning, structured tender documents, non-interference sequence index, reference templates of tender documents.

1 Introduction

The tender document is the basis for the tenderer to make the bidding documents and participate in the tenders, and also the important basis for the evaluation experts to evaluate the bids (bidding represents the act of offering something or to do something for a particular price, while tender represents the act calling for bidders to bid for something, then selecting the best supplier). In addition, the tender document is the basis for signing the contract, and most of the contents of the tender document should be included in the contract [11]. Therefore, the compilation of tender documents is a critical link. The compilation of tender document is the processing of editing clauses and charts in a tender document in accordance with a certain format and logic, in order to describe the tender project. Through interviews with experts and business personnel in the tender field, this paper summarizes the actual process of tender documents compilation and finds that its core activities are collecting similar historical tender documents, selecting tender document compilation templates, and modifying tender document templates partially, these activities are not only the key points but also the difficulties in compiling the tender documents.

Traditional compilation methods of tender documents need professionals to select similar tender documents from historical documents, and modify them according to the actual content of the project artificially. Finally, a target tender document is obtained. This method is reliable to a certain extent, but over relying on people might cause low efficacy and high operating cost. At the same time, it is easy to lead to the lack of compilation content, inappropriate, and even illegal behavior [15, 22].

At present, the research on the compilation of tender documents mainly focuses on two aspects. On the one hand, the research basically stays in emphasizing the standardization of the contents of tender documents. This kind of article mostly appears in the periodical papers in the construction industry. For example, the literature [20] points out the problems easily occurred in the traditional tender document compilation method, and puts forward the matters needing attention in the compilation process to remind the compilers to avoid. On the other hand, the research has begun to introduce machine writing, natural language processing and other computer means to optimize the content compilation process of tender documents, which is still at the stage of structuring the tender documents. For example, in the latest paper [21], aiming at the problem of template solidification of tender documents for power industry, nine standardized tender document templates are divided for different tender project types, which are used for reference in the compilation of tender documents. Besides, the division method does not consider that the same tender project type will also cause inconsistency in the use of templates, so it is difficult to ensure the pertinence of the content recommendation of the template. Therefore, it is significant to propose a complete set of intelligent recommendation algorithm for tender document content.

Therefore, this paper studies the intelligent recommendation of tender documents based on case-based reasoning technology. The main innovations are as follows: First of all, this paper summarized the key and difficult problems in the compilation process, and then introduced case-based reasoning as the theoretical framework to build an intelligent recommendation algorithm of tender document content from similar case retrieval, to case reference template reuse, to case reference template revision, and to new case learning, so as to ensure the rationality of the recommendation system; Secondly, we consumed clustering method to retrieve similar tender case sets, which realized the first step of content recommendation of tender documents based on case-based reasoning, which greatly reduced the recommended scope of tender cases; Thirdly, the non-interference sequence index was consumed to measure the similarity of tender case content, to realize the construction and recommendation of similar case reference templates and content modules, so as to ensure that the recommended content is more scientific and accurate;

Finally, through the construction of tender case learning algorithm, the tender knowledge of new problems was updated into the database, which made the recommendation system more perfect.

2 Related works

2.1 Compilation of tender documents

Literature [4, 14] points out that 'The purpose of the compilation of the tender documents is to provide all relevant information about the proposed contract, rules, conditions, etc. to the contractor, and to provide each contractor with general data in sufficient detail to adapt to the project. Therefore, it should be prepared as detailed and accurate as possible'.

Brook [2] indicated that major problems associated with content quality of tender documents include missing information, late information, wrong information, insufficient detail, impracticable designs, inappropriate information, unclear information, provisional information, poorly arranged information, uncoordinated information and conflicting information.

Chen et al. [3] proposed several opinions in compiling the tender documents: should fully grasp the business information, accurately identify the tender requirements, and strictly abide by the laws and regulations at the same time to invite industry experts, and then ensure the reasonable and lawful content written in the tender documents. On November 1, 2007, the Standard Construction Tender Document of the People's Republic of China was officially released. This document is the only official authorized document in China, and it is also a relatively complete and formal template for compiling tender documents [5].

Wang proposed an electronic tender document compilation system. The template components of the tender document were compilated by business personnel artificially according to the national standard tender document, and then the template combining and editing were carried out for the actual tender project. This method had certain logical innovation [17]. The emergence of machine writing has provided an effective way to replace the manual writing method, and it is mainly used in news writing, poetry creation, automatic summarization, biography generation, and abstract writing etc. [19]. Tang edited and integrated the tender documents by using the extraction natural language generation mechanism, which improved the compilation efficiency of the tender documents to a certain extent. However, the text generation is relatively stiff, which requires human resources to polish the language [16].

In a word, the above researches have mainly completed the structural and standardized study of the tender documents, but there was a lack of appropriate use of the existing tender lessons in compiling tender documents. This paper will propose a case-based reasoning method to optimize the entire process of tender document construction. The tender document template is constructed automatically by referencing historical tender lessons, which greatly reduce human participation, making the tender documents compilation more efficient and scientific.

2.2 Case-based reasoning theory

The existing theories have made a preliminary exploration on the optimization of the compiling tender documents. Aiming at their shortcomings, this paper applies data mining technology and case-based reasoning (CBR) to analyze and design the recommendation algorithm of tender documents content.

The basic process of CBR is: when a new problem is encountered, the system retrieves the original case base according to the key characteristics, finds a candidate case that is most similar to the new problem, and reuses the reference template of the candidate case. If the reference template of this candidate case is not satisfied, it can be modified to adapt to the problem to be asked. Finally, the revised case can be saved in the database as a new case, so as to serve as a reference next time when similar problems are encountered [6]. The above process can be summarized into four steps: case retrieval, case reuse, case revision and case learning. CBR takes case as knowledge element, knowledge acquisition and representation are natural and direct, and can be self-learned. It has the advantages of simplifying knowledge acquisition, improving solving efficiency through direct acquisition, high solving quality, and being suitable for non-computational derivation [10].

With the deepening of the research of CBR, the application scope and field of CBR are expanding. It has been applied in general problem solving, legal cases, medicine and medicine, weather forecasting, machine fault diagnosis, enterprise consulting and decision-making, etc., and has proved the effectiveness and practicability of CBR method [1, 7, 9, 12, 13].

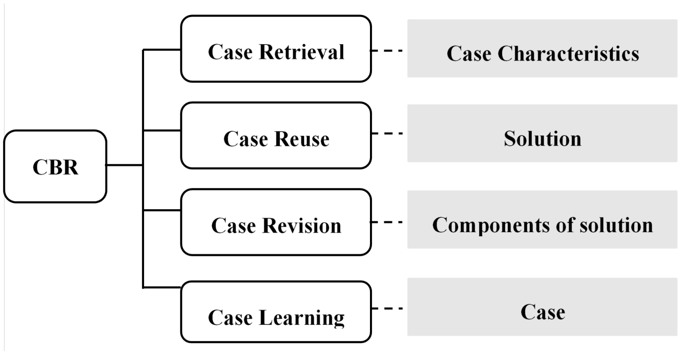

Figure 1: The research object of the core link of CBR

Combined with the steps of CBR, the research objects of each link of compiling tender documents were obtained. As shown in Figure 1, the research object of the case retrieval is case characteristics, the research object of case reuse and revision is the solution (reference template)nad its components, and the research object of the case learning is the case.

This paper follows the thinking mode of CBR to create an intelligent recommendation method for compiling tender documents. The research ideas of this paper are as follows: Firstly, the tender document was structured and divided into several document units. This paper selected modular document units as examples for content recommendation. Secondly, the nearest neighbor algorithm was used to calculate the similarity of tender case characteristics among cases, and similar tender cases were retrieved from historical tender cases. Then, the content recommendation model of tender documents based on non-interference sequence index was established, and the recommendation sequence of reference template can be directly used from the similar case set, i.e. the case reuse. If the reference template cannot be used directly, enter case revision to make partial revision (content module) to a reference template, and the new solution will be learned into the tender knowledge base as a reference template for the next recommendation process finally. This method separated business personnel from the tedious task of compiling tender documents, and ensured the efficient compiling of tender documents by optimizing the compiling process of tender documents.

3 Construction of content recommendation model of tender documents

3.1 Structured processing of tender documents

Definition 1: The Document Unit is the content of the tender document which is constrained by the title at any logical structure level in the tender document.

The analysis of the original tender documents reveals that the tender documents have the following characteristics: a. The tender document has a relatively fixed structure framework; b. According to its own compilation requirements, the tender documents contain many titles, which naturally divide the tender documents into content units with logical independence; c. The content units of tender documents have strict hierarchical relationship logically; d. The content units of the same level have no logical sequence relationship.

The core idea of CBR is to use the historical lessons to help solve new similar problems. Therefore, this paper classified the document unit manually from the perspective of whether it needs to reference historical lessons and the degree. Among all tender document units, the factual document unit is the tender document unit which does not need to refer to the historical tender lessons. It is used to objectively describe a tender activity. The empirical document unit is all document units except the factual document units, which can improve the compilation efficiency by referring to or directly citing the historical tender lessons. And according to reference degree, the empirical document unit could be also divided into three types further: the formatted document unit, the modular document unit and the characteristic document unit, as shown in Figure 2. Structured processing is to divide the tender documents into detachable, mergeable and logically connected document units, which is the basis of recommending the contents of tender documents.

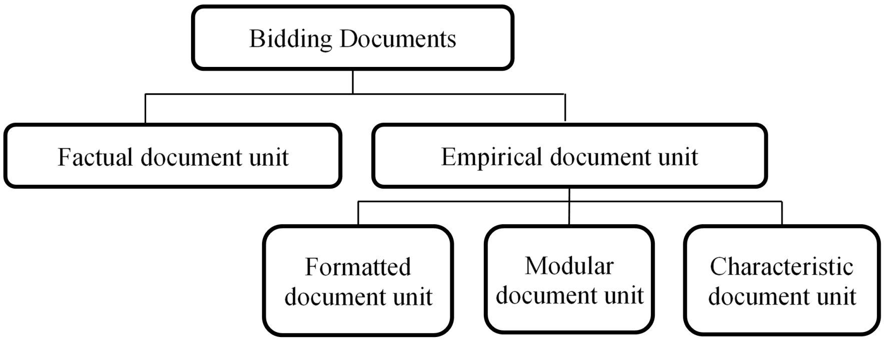

Figure 2: Structured model of tender documents

Table 1: Recommendation strategy of tender document content

Document unit	Content characteristic		Recommendation strategy	
	historical lessons	**reference degree**	**recommended**	**recommended content**
Factual	unneeded	not at all	no	
Formatted	need	complete	yes	templates
Modular	need	a lot	yes	templates, components of templates
Characteristic	need	a little	yes	components of templates

According to the division of document units, the recommended object of the tender document content specified in this paper was the empirical document unit, and the recommended content were the templates and components of the templates. The modular file unit in the empirical document unit covered all kinds of recommended content, therefore, taking it as an example, this paper selected a document unit named “bidder qualification requirements” to study content recommendation method of tender documents.

3.2 Retrieval of similar tender cases

By summarizing the experience of experts, this paper obtained 17 elements (as shown in Figure 3) covering five aspects of project overview, qualification conditions, bid evaluation methods, suppliers and complaints/objections, which can be used for similar tender case retrieval. However, if it was directly applied to case retrieval, it would lead to low matching degree of similar cases and high computational complexity inevitably. Therefore, we formulated the following principles to simplifying elements.

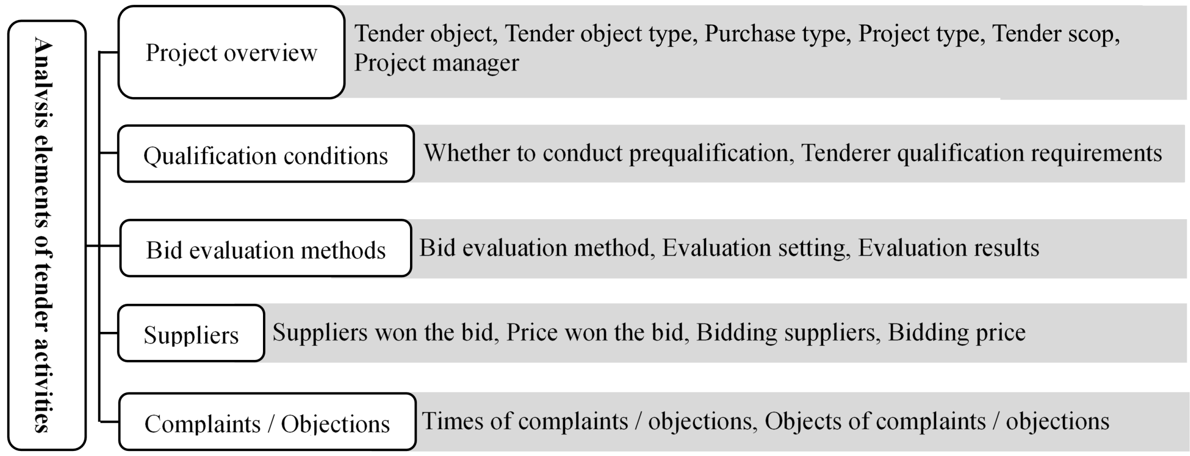

Figure 3: Analysis elements of tender activities

According to the principle of data sparsity, the elements of suppliers were reduced. The main reason was that part of supplier data was missing and cannot be filled due to special circumstances, such as tender failure.

According to the principle of representativeness of data content, firstly, the project manager, tender scope and complaint/objection elements in project overview were reduced, as the values of the above elements in all tender activities were nearly the same. Then, the evaluation setting under the analysis factors of project overview and scoring method was reduced. The values of the above elements in all tender activities were nearly different, mainly because the tender projects were built around specific tender objects.

According to the principle of data processing simplicity, the bidder qualification requirements and scoring methods in condition of qualification elements were reduced. The main reason for the scoring results under the condition of analysis elements was that the data types of the above analysis elements were all textual, which required complex natural language processing methods to analyze the specific content.

In conclusion, the tender case characteristics <project type (A_1), procurement method (A_2), whether to conduct prequalification (A_3), bid evaluation method (A_4), tender object type (A_5)> were constructed .

Table 2: Data Type Transformation of Case Characteristics

	Original Data Type	Transformation Rules	New Data Type
A_1	str	0: Goods, 1: Engineering	num
A_2	str	0: Public tender, 1: Invitational tender	num
A_3	Bool	0: No, 1: Yes	num
A_4	str	0: The Comprehensive Evaluation Method, 1: The Lowest Bid Evaluation Method	num
A_5	str	0: Pump, 1: Groove, 2: Elevator,......	str

By analyzing the data type of these case characteristics, it was found that A_1, A_2 and A_4 were all character variable with two kind of value, so the two kind of value can be directly converted into 0-1 values. Besides, as Boolean variable, A_3 could be converted to 0-1 values as well. The 0-1 values above could be used in the Equation (3) directly.

Due to the variety of types of A_5 , it was impossible to realize the continuity by artificial design of change rules, and we just number its character value. So far, the whole transformation results have been shown in Table 2. In order to achieve A_5 participating in the calculation of Equation (3), it was necessary to deeply study the tender cases of specific A_5 and the number of each type of A_5 in each category, to calculate the distance of different tender object types. The distance of two tender object types is introduced as follows:

$$Dist(c_{kA_5}, c_{lA_5}) = |c_{kA_5} - c_{lA_5}| = 1 - \sum_{\rho=1}^{g} J_\rho(c_{kA_5}, c_{lA_5}) \tag{1}$$

$$J_\rho(c_{kA_5}, c_{lA_5}) = \begin{cases} \frac{|c_{kA_5}|_\rho + |c_{lA_5}|_\rho}{|c_{kA_5}| + |c_{lA_5}|} & (\exists (c_{kA_5} \in CL_\rho) \wedge (c_{lA_5} \in CL_\rho)) \\ 0 & (\neg (c_{kA_5} \in CL_\rho) \wedge (c_{lA_5} \in CL_\rho)) \end{cases} \tag{2}$$

At present, the commonly used case retrieval algorithms include knowledge guide method, neural network method, inductive index method and nearest neighbor method [8]. In this paper, the nearest neighbor method was selected as the method basis for tender case retrieval. The difference degree of case characteristics between tender case C_k and C_l is as follows:

$$Sim_{case}(C_k, C_l) = \sqrt{\frac{\sum_{i=1}^{N} |c_{kA_i} - c_{lA_i}|^2}{N}} \tag{3}$$

N is the total number of tender case characteristics, N=5, c_{kA_i} is the value of tender case C_k under tender case characteristic A_i, The smaller the value of $Sim_{case}(C_k, C_l)$, the higher the similarity of tender cases, otherwise, the lower the similarity.

Based on the nearest neighbor method, this paper proposed a tender case retrieval algorithm. The steps are as follows:

Algorithm 1 $CaseRetrieval(C_{n+1})$: Similar tender cases of new problem C_{n+1}

Input: tender cases $\{C_k|k \in N^+\}$, tender case class CL_ρ ($\rho \in [1, g]$), decision threshold λ_{case}, tender case characteristics $A =< c_{(n+1)A_1}, c_{(n+1)A_2}, c_{(n+1)A_3}, c_{(n+1)A_4}, c_{(n+1)A_5} >$ of new problem C_{n+1}
Output: similar tender cases C_{sim} of new problem C_{n+1}
Step 1: Calculate difference degree $Sim_{case}(C_{n+1}, C_k)$ between the new problem C_{n+1} and all the tender cases C_k.
Step 2: Similar case retrieval. Set $C_{sim} = \phi$, if $Sim_{case}(C_{n+1}, C_k) \leqslant \lambda_{case}$, thus $C_k \in C_{sim}$; else $C_k \notin C_{sim}$
Step 3: Output the results of C_{sim}

3.3 Reuse of reference template of tender document

After the retrieval of historical tender cases similar to the new problems, the compilation of tender documents will enter the stage of compiling tender documents or partial modification referring to similar case templates. The compilation template of tender document is the reference template, and the process of constructing and recommending reference template of similar cases is case reuse. First of all, let us introduce several tender concepts need to be used. Tender case C_k is a set composed of a tender document and the characteristics of tender object described in the tender documents. And tender document is composed of several clauses naturally, thus, clause D_i is the minimum content unit with logical number in the tender document. Besides, the tender cases with the same tender document could be clustered into one tender case class (CL_ρ), and each tender case class only contain one document template (S_i).

Definition 2: Reference Template is solution in which all the content in the tender document is not the same, and denoting it as (S_i).

Taking the document unit of bidder qualification requirements document unit as an example, Table 3 describes 3 tender cases and 3 kind of tender document used by the cases. However, document content of C_1 and C_3 are the same, thus only 2 reference templates are included finally, i.e. $S_1 = \{D_1, D_2, D_3, D_4\}$, $S_2 = \{D_1, D_3, D_5, D_6\}$.

Table 3: The clauses usage in bidder qualification requirements document unit

Clause	**Content**	C_1	C_2	C_3
D_1	The bidder has independent legal personality and the ability to perform the contract;	1	1	1
D_2	Only manufacturer bids are accepted;	1	0	1
D_3	The bidder has ISO quality system certification;	1	1	1
D_4	The bidder has the financial, technical and production capacity or supply performance required for the performance of the contract, and meets the corresponding conditions specified in the tender document;	1	0	1
D_5	The manufacturer is expected to make direct investment in the project, but also accept the agent's bidding; if the agent bids, the manufacturer's sole authorization for the project shall be provided;	0	1	0
D_6	Different units in charge of the same person, or with a controlling or management relationship, may not participate in the same tender or the tender for the same tender project whose bid sections have not yet been divided.	0	1	0

Clause Usage $J_k(D_i)$ is the judgment function of whether clause D_i is used. If clause D_i is used in tender case C_k, thus $J_k(D_i) = d_{ik} = 1$; otherwise, $J_k(D_i) = d_{ik} = 0$, as shown in Table 3 (1 means to be used, 0 means not to be used). Thus, the connection between clauses can be established through their usage of the tender case. When the two clauses appear or disappear at the same time in some tender cases, it indicates that the two clauses are related closely, such as set $\{D_1, D_3\}$, $\{D_2, D_4\}$, $\{D_5, D_6\}$; On the contrary, such as the clauses D_2 and D_5, D_4 and D_6, they never appear at the same time, the similarity between them is low.

Based on the above analysis, the non-interference sequence index can be used to measure the difference between two objects. If the non-interference sequence index is the same, the two objects have strong similarity. In the next part of this paper, the contents of the tender documents will be clustered and recommended according to this method.

Non-interference Sequence is a positive integer sequence, $M = (M_1, M_2, \ldots, M_k, \ldots)$, where, nth clause is greater than the sum of the previous (n-1) clause, i.e., $M_n > \sum_{k=1}^{n-1} M_k$, $n \geqslant 2$ [18].

Suppose a data set C has m objects and n attributes, and these attributes are all binary attributes, which are recorded as $C_1, C_2, \ldots, C_k, \ldots, C_n$, then d_{ik} is the value of object $D_i (i \in [1, m])$ for attribute C_k. The Non-interference Sequence Index of object D_i is: $q(D_i, M) = \sum_{k=1}^{n} d_{ik} * M_k$, where, M is a selected non-interference sequence [18].

Lemma 1 [18]: For two objects D_i and D_j in the binary attribute data set C, whose attribute values are $d_{i1}, d_{i2}, \ldots, d_{in}$ and $d_{j1}, d_{j2}, \ldots, d_{jn} (d_{ik}, d_{jk} \in \{0, 1\}, k \in [1, n])$, respectively, which are recorded as $J_C(D_i) = (d_{i1}, d_{i2}, \ldots, d_{in}) = \{J_k(D_i) | k \in [1, n]\}$, $J_C(D_j) = (d_{j1}, d_{j2}, \ldots, d_{jn}) = \{J_k(D_j) | k \in [1, n]\}$. Non-interference sequence index is $q(D_i, M) = \sum_{k=1}^{n} d_{ik} * M_k$ and $q(D_j, M) = \sum_{k=1}^{n} d_{jk} * M_k$. Thus if $q(D_i, M) = q(D_j, M)$, $J_C(D_i) = J_C(D_j)$, otherwise $J_C(D_i) \neq J_C(D_j)$.

To sum up, the recommended idea for reference template of tender document is divided into two steps, as shown in Figure 4. The first is the construction of reference template, which identifies all the reference templates used from the tender documents of the tender case set similar to the new problem, so as to ensure the relevance of the recommended content; the second is to recommend the reference template according to the frequency of the reference template in the similar tender case set and the sorting result of the reference template, so as to realize the content recommendation of the tender document reference template and ensure the reuse work high efficiency.

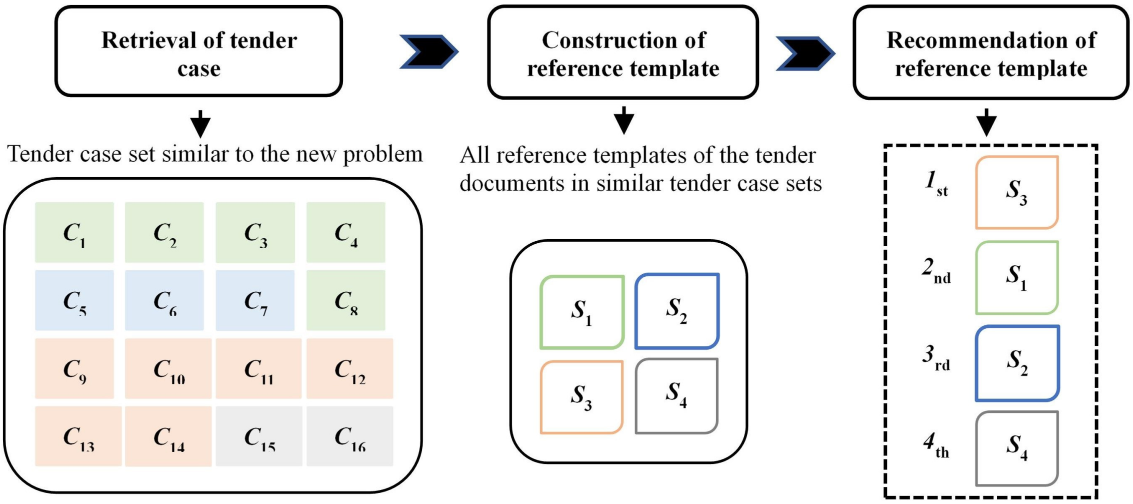

Figure 4: Recommended idea for reference template of tender documents

Theorem 1: In the tender case database, if and only if the non-interference sequence index of all tender cases in the tender case class is the same, there is only one reference sample in the tender case class. This is defined as The Decision Theorem of Reference Template.

For any tender case class CL_ρ, and any non-interference sequence $M = (M_1, M_2, \ldots, M_i, \ldots)$ in the tender case database, assume any tender case $C_k, C_l \subset CL_\rho(\rho \in [1, g], k \neq l)$, the template of C_k is S_k, all clauses in the tender case database is $D = \{D_i | i \in [1, m]\}$,Theorem 1 shows that: (1) if $q(C_k, M) = q(C_l, M)$, thus $S_k = S_l$; (2) if $q(C_k, M) > q(C_l, M)$ or $q(C_k, M) < q(C_l, M)$, $S_k \neq S_l$.

Proof: According to the Lemma 1, (1) if $q(C_k, M) = q(C_l, M)$, thus $J_k(D) = J_l(D)$, and $\forall i, i \in [1, m], J_k(D_i) = J_l(D_i)$; Moreover $S_k = \{D_i | J_k(D_i) = 1\}, S_l = \{D_i | J_l(D_i) = 1\}$, and $J_k(D_i), J_l(D_i) \in \{0, 1\}$, thus $S_k = S_l$; At the same time, $C_k, C_l \subset CL_\rho(\rho \in [1, g], k \neq l)$, so there is only one reference template S_k in tender case class CL_ρ. (2) if $q(C_k, M) > q(C_l, M)$ or $q(C_k, M) < q(C_l, M)$,thus $J_k(D) \neq J_l(D)$; Set $i(i \in [1, m])$ be arbitrary subscript that satisfy $J_k(D) \neq J_l(D)$, thus $J_k(D_i) \neq J_l(D_i)$, and as $J_k(D_i), J_l(D_i) \in \{0, 1\}$, thus $J_k(D_i) = 1, J_l(D_i) = 0$, or $J_k(D_i) = 0, J_l(D_i) = 1$. At this time, if $J_k(D_i) = 1, J_l(D_i) = 0$, and as $S_k = \{D_i | J_k(D_i) = 1\}, S_l = \{D_i | J_l(D_i) = 1\}$, then $D_i \in S_k, D_i \notin S_l, S_k \neq S_l$; Besides, $C_k, C_l \subset CL_\rho(k, l \in [1, g], k \neq l)$, thus there are at less two reference templates in tender case class CL_ρ. Otherwise, it could also be proves as above if $J_k(D_i) = 0, J_l(D_i) = 1$.

To sum up, the Recommendation algorithm of reference templates of tender documents was proposed. The steps are as follows:

In step 3, $|S_\rho|_C$ is the frequency of reference template S_ρ that appearing in the tender case set C, $|C|$ is the total number of tender cases, and reference template value of S_ρ in C is $V_C(S_\rho) = |S_{\rho C}|/|C|$.

In step 4, the reference template sequence SS is a sequence with strict order relationship after sorting the reference template set $S = \{S_\rho | \rho \in [1, g]\}$ according to certain rules.

Algorithm 2 $TemplatesRecommend(C_{n+1})$: Recommended sequence of reference templates of tender documents

Input: clause usage $\{d_{ik}|i \in [1,m], k \in [1,n]\}$, arbitrary non-interference sequence M, tender case set $C_{sim} = \{C_k|k \in [1,n]\}$ that similar to the new problem C_{n+1}
Output: recommended sequence of reference template S_{rec} for new problem C_{n+1}
Step 1: Calculate non-interference sequence index $q(C_k, M)$ of each tender case in C_{sim}.
Step 2: Construct reference templates. The tender cases with the same $q(C_k, M)$ are merged into one tender case class CL_ρ at the same time, define the reference template of this class as S_ρ at this time, all reference template sets $S_{C_{sim}}$ are obtained.
Step 3: Assess reference template value. Calculate the recommended value $V_{C_{sim}}(S_\rho)$ of each reference template for $S_{C_{sim}}$ in C_{sim}.
Step 4: Construct recommended sequence of reference templates of tender documents. The reference sample set $S_{C_{sim}}$ is arranged in descending order according to $V_{C_{sim}}(S_\rho)$, and the reference sample sequence $SS = (S_1, S_2, ...S_m)$ is obtained. Intercept the reference template sequence SS satisfying $V_{C_{sim}}(S_\rho) > \alpha$ to construct the reference template recommended sequence $S_{rec} = (S_1, S_2, ...S_i)$ to the new problem C_{n+1}.
Step 5: Output the results of S_{rec}.

3.4 Revision of reference template of tender document

After analyzing the content recommendation ideas of tender documents, it was found that the precondition for entering the recommendation process of content modules was that the business personnel have determined subjectively that all the recommended reference templates cannot be used directly, and need to modify the content of one of the reference templates. Based on hierarchical clustering method, this paper proposed the idea of constructing content module to modify the reference template of tender documents as shown in Figure 5.

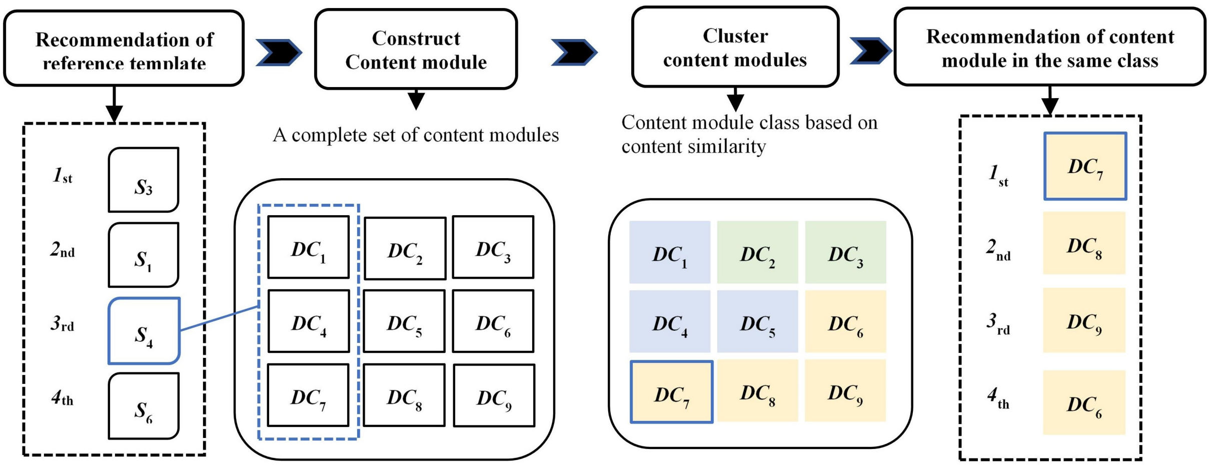

Figure 5: Recommended idea for content module of reference templates

Firstly, constructing a complete set of content modules, and the content modules of each reference template were identified from the reference template recommendation sequence to ensure the relevance of the revised content. Then, the hierarchical clustering of content modules is carried out based on the similarity of content modules. At the same time, according to the frequency of content modules in similar tender case sets, the recommended sequence of content modules within the same category of content modules to be modified is obtained, and the recommended content modules are used to modify it. The accuracy of the correction is ensured.

Definition 3: Content Module DC_p is a collection of clauses. In the same way, reference template is a collection of content modules.

Theorem 2: In the tender case database, if and only if the non-interference sequence index of all clauses in the content modules is the same, the content modules can divide all reference samples. This is defined as the Decision Theorem of Content Modules.

For any content module DC_p, assume any clauses $D_i, D_i \subset DC_p(i, j \in [1, m], i \neq j)$, Theorem 2 shows that: (1) if $q(D_i, M) = q(D_j, M)$ and $D_i \in S_k$, thus $DC_p \in S_k$; (2) if $q(D_i, M) = q(D_j, M)$ and $D_i \notin S_k$, thus $DC_p \notin S_k$; (3) if $q(D_i, M) > q(D_j, M)$ or $q(D_i, M) < q(D_j, M)$, $D_i \in S_k$, thus $D_j \notin S_k$; (4) if $q(D_i, M) > q(D_j, M)$ or $q(D_i, M) < q(D_j, M)$, and $D_i \notin S_k$, thus $D_j \in S_k$.

Proof: According to the Lemma 1, (1) if $q(D_i, M) = q(D_j, M)$, thus $J_C(D_i) = J_C(D_j)$, and $\forall k, k \in [1, n], J_k(D_i) = J_k(D_j)$. If $D_i \in S_k$, thus $J_k(D_i) = 1, J_k(D_j) = 1$ and $D_j \in S_k$, and as $D_i, D_i \subset DC_p(i, j \in [1, m], i \neq j)$, thus $DC_p \in S_k$; (2) While if $D_i \notin S_k$, thus $J_k(D_i) = 0, J_k(D_j) = 0$ and $D_j \notin S_k$, and as $D_i, D_i \subset DC_p(i, j \in [1, m], i \neq j)$, thus $DC_p \in S_k$; (3) if $q(D_i, M) > q(D_j, M)$ or $q(D_i, M) < q(D_j, M)$, thus $J_C(D_i) \neq J_C(D_j)$. Set $i(i \in [1, m])$ be arbitrary subscript that satisfy $J_k(D) \neq J_l(D)$, thus $J_k(D_i) = J_k(D_j)$. If $D_i \in S_k$, thus $J_k(D_i) = 1, J_k(D_j) = 0$ and $D_j \notin S_k$. And as $D_i, D_i \subset DC_p(i, j \in [1, m], i \neq j)$, it may lead to the contradiction between $D_i \in S_k$ and $D_j \notin S_k$, thus DC_p could not divide S_k. (4) While if $D_i \notin S_k$, thus $J_k(D_i) = 0, J_k(D_j) = 1$ and $D_j \in S_k$. And as $D_i, D_i \subset DC_p(i, j \in [1, m], i \neq j)$, it may also lead to the contradiction between $D_i \notin S_k$ and $D_j \in S_k$, thus DC_p could not divide S_k as well.

Besides, several concepts should be introduced as follows, in order to conduct the revision of reference template. $|DC_p|$ is the frequency that the content module DC_p is occurred in the tender case $C = \{C_k | k \in [1, n]\}$, the content module value of DC_p is: $V_C(DC_p) = |DC_p|/n$.

It is known that the usage of clauses D_i and D_j in the tender case database $C = \{C_k | k \in [1, n]\}$ are represented as $J_C(D_i) = \{d_{ik} | k \in [1, n]\}, J_C(D_j) = \{d_{ik} | k \in [1, n]\}$, then the difference degree between clauses D_i and D_j is:

$$Sim_{doc}(D_i, D_j) = \sqrt{\frac{\sum_{k=1}^{n} |d_{ik} - d_{jk}|^2}{n}} \tag{4}$$

According to Theorem 2, all clauses in the same content module are used the same in the tender case database, so a content module only needs to select one clause as a representative to calculate the degree of difference between the content modules. Thus, as for content modules $DC_p = \{D_i | D_i \in DC_p, i \in [1, g]\}, DC_q = \{D_j | D_j \in DC_q, j \in [1, l]\}$, the difference degree between them is:

$$Sim_{doc}(DC_p, DC_q) = Sim_{doc}(D_i, D_j) \tag{5}$$

And, content modules whose difference degree meets a certain threshold are combined into a content module class $DM_x = \{DC_p | DC_p \in DM_x, x \in [1, R]\}$, and content modules within the same category can be recommended to each other. In the same way, as for two content module classes $DM_x = \{DC_p | DC_p \in DM_x, x \in [1, R]\} = \{D_i | D_i \in DM_x\}$ and $DM_y = \{DC_q | DC_q \in DM_y, y \in [1, R]\} = \{D_j | D_j \in DM_y\}$, the difference degree between them is:

$$Sim_{doc}(DM_x, DM_y) = max(Sim_{doc}(DC_p, DC_q)) = max(Sim_{doc}(D_i, D_j)) \tag{6}$$

Recommended sequence of content module $D_{rec}(p)$ is a sequence of content modules starting from DC_p. This sequence is ordered according to certain rules. For example, in Figure 5, we regard DC_7 as the target content module, the recommended sequence of DC_7 is $D_{rec}(7) = (DC_7, DC_8, DC_9, DC_6)$.

According to the content module recommendation idea of the reference template, the revision algorithm of the reference templates of the tender documents was proposed. The steps are as follows:

Algorithm 3 $ModulesRecommend(DC_p)$: content modules for target content module DC_p

Input: clause usage $\{d_{ik}|i \in [1, m], k \in [1, n]\}$, arbitrary non-interference sequence M, recommended sequence of reference template S_{rec} to the new problem C_{n+1}
Output: recommended sequence of content module $D_{rec}(p)$ for content module DC_p
Step 1: Calculate non-interference sequence index $q(D_i, M)$ of each tender case in C_{sim}.
Step 2: Construct content module. The clauses with the same non-interference sequence index are merged into a content module DC_p.
Step 3: Assess content module value. Calculate the recommended value $V_C(DC_p)$ of each content module in C_{sim}.
Step 4: Calculate the difference degree OD_{sim} between content modules.
Step 5: Construct content module class. Regard each content module DC_p as a content module class DM_p.
Step 6: OD_{sim} is assigned to the difference degree of initial content module class ODM_{sim}. By using hierarchical clustering method, the content module classes with the smallest difference between are merged into a new content module class. Repeat the merge until all content modules merge one class. Select threshold β and regard the content module class satisfying $ODM_{sim} < \beta$ as the fianl content module class.
Step 7: Use content module to divide reference template. Identify the recommended sequence of reference templates to be revised, and divide the content module set $S_\rho\{DC_p, DC_i, DC_j...\}$ of each reference template S_ρ. **Step 8:** Construct recommended sequence of content modules. Assume that the target content module is DC_p, and identify the content module class $DM_x = (DC_p, DC_q, DC_s, DC_r)$ in which DC_p is located, then sort DC_p, DC_q, DC_s, DC_r in this class in descending order according to their difference degree with DC_p. If the difference degree is the same, these content modules are further sorted in descending order according to the content module value, and the final sorting result is regarded as the recommended sequence of content modules. For example, if $Sim_{doc}(DC_p, DC_s) < Sim_{doc}(DC_p, DC_r) = Sim_{doc}(DC_p, DC_q)$ and $V_C(DC_q) > V_C(DC_r)$, thus $D_{rec}(p) = (DC_p, DC_s, DC_q, DC_r)$.
Step 9: Output the results of $D_{rec}(p)$.

3.5 Study of tender case knowledge

The tender knowledge involved in this paper includes: tender case class (K_1), distance of tender object types (K_2), reference template (K_3), content module (K_4), content module value (K_5), clause difference degree (K_6), content module difference degree (K_7), recommended sequence of content module (K_8). Only the tender case reuse and tender case revision can generate new tender knowledge. Moreover, the generation of new knowledge is very different between tender case reuse and tender case revision as shown in Table 4, where Y means Yes, N means No, U means Unsure. Therefore, it is necessary to design tender case learning algorithms separately.

Table 4: Generation of new tender knowledge

Tender knowledge	K_1	K_2	K_3	K_4	K_5	K_6	K_7	K_8
Case reuse	Y	Y	N	N	Y	Y	Y	U
Case revision	Y	Y	Y	U	Y	Y	Y	U

3.5.1 Tender knowledge learning of case reuse

According to the new knowledge generation of case reuse in Table 4, the tender case learning algorithm ideas based on case reuse was proposed in this section. The input of the algorithm was: new problem C_{n+1}, clause usage after learning $\{d_{ik}|i \in [1,m], k \in [1,n+1]\}$, original content module DC_p, original content module value $V_C(DC_p)$, original tender case class $CL_\rho(\rho \in [1,g])$, original reference template $S_\rho(\rho \in [1,g])$, original distance of tender object type $Dist(c_{iA_5}, c_{jA_5})(i,j \in [1,n])$, original clause difference degree $Sim_{doc}(D_i, D_j)(i,j \in [1,m])$.

• Tender case class learning method

S_{n+1} as the solution of the new case C_{n+1} constructed by case reuse was a reference template S_ρ existing in the original tender case database, the new case C_{n+1} can be directly added to the tender case class CL_ρ using the reference template S_ρ. Thus $\forall i(i \in [1,m])$, if $d_{ik} - d_{i(n+1)} = 0$, then $C_{n+1} \in CL_\rho, C_k \in CL_\rho$.

• Distance of tender object type learning method

The learning result in the case reuse process was that a new problem is added to an original tender case classes, so only the distance related to the tender object type in the same tender case class with the new problem were affected, it could be recalculated through Equation (1) and Equation (2). The distance results irrelevant to the tender object type of the new problem did not change.

• Content module value learning method

Because only one new problem was learned at a time, frequency of content module might remain unchanged or increase once, and the total number of tender cases increased by one.It was known that, original tender case was $C = \{C_k|k \in [1,n]\}$, original content module was DC_p, original content module value is $V_C(DC_p)$, reference template of new problem is S_{n+1}, then the new content module value $V_C{'}(DC_p)$ was as follows:

$$V_C{'}(DC_p) = \begin{cases} V_C(DC_p) * n/(n+1) & (DC_p \notin S_{n+1}) \\ (V_C(DC_p) * n + 1)/(n+1) & (DC_p \in S_{n+1}) \end{cases} \tag{7}$$

• Clause difference degree learning method

It was known that, clause was $D_i, D_j(i,j \in [1,m])$, original clause difference degree was $Sim_{doc}(D_i, D_j)$, clause usage was $\{d_{i(n+1)|i \in [1,m]}\}$ of new problem C_{n+1}, then new clause difference degree $Sim_{doc}{'}(D_i, D_j)$ could be obtained by adjusting $Sim_{doc}(D_i, D_j)$:

$$Sim_{doc}{'}(D_i, D_j) = \sqrt{\frac{((Sim_{doc}(D_i, D_j))^2 * n + |d_{i(n+1)} - d_{j(n+1)}|^2)}{n+1}} \tag{8}$$

• Content module difference degree learning method

There was no new content module generated in the case reuse, but the new problem changed the clause difference degree, which led to the change of content module difference degree indirectly. Using Equation (7) to update the new content module difference degree $Sim_{doc}(DC_p, DC_q)$ as follows:

$$Sim_{doc}{'}(DC_p, DC_q) = Sim_{doc}{'}(D_i, D_j) \tag{9}$$

where $D_i \in DC_p, i \in [1,g]\}, D_j \in DC_q, j \in [1,l]$.

• Recommended sequence of content module learning method

When content module class $DM_x, x \in [1,R]$ had been obtained by default, and new content module difference degree $Sim_{doc}{'}(DC_p, DC_q)$ and new content module value $V_C{'}(DC_p)$ had been updated, we used step 8 of Algorithm 3 to update the new recommended sequence of content module $D_{rec}{'}(p)$.

3.5.2 Tender knowledge learning of case revision

According to the new knowledge generation of case reuse in Table 4, the tender case learning algorithm ideas based on case revision was proposed in this section. The input of the algorithm was: new problem C_{n+1}, clause usage after learning $\{d_{ik}|i \in [1,m], k \in [1, n+1]\}$, original content module DC_p, original content module value $V_C(DC_p)$, original tender case class $CL_\rho(\rho \in [1, g])$, original reference template $S_\rho(\rho \in [1, g])$, original distance of tender object type $Dist(c_{iA_5}, c_{jA_5})(i, j \in [1, n])$, original clause difference degree $Sim_{doc}(D_i, D_j)(i, j \in [1, m])$.

- Tender case class learning method

Since the premise of entering case revision was that the new problem cannot be solved by case reuse, i.e. the solution of the new problem was different from all the solutions in the original tender case database. Thus, the new problem did not belong to any original tender case class according to Theorem 3 and Algorithm 3.It was known that, new problem was C_{n+1}, original tender case class was $CL_\rho(\rho \in [1, g])$. The tender case database need to add a tender case class which only contained the new tender case C_{n+1}, thus, the new tender case class after learning was $CL_\rho(\rho \in [1, g+1])$, and $CL_{\rho+1}\{C_{n+1}\}$.

- Distance of tender object type learning method

As a new case C_{n+1} was added after learning, it meant its tender case class only contained one tender case type. If the tender case type is a new type, this tender case type will be added in the database, and its distance with other tender object types is 0 according to Equation (1) and Equation (2); Otherwise all the distance of tender object type remains unchanged.

- Reference template learning method

Since the relationship between tender case class and reference template was 1-to-1, the solution to the new problem was a new reference template. Thus, the new reference template after learning was $S_{\rho+1} = S_{n+1}$.

- Content module learning method

There were three ways to construct the reference template of case revision learning: (1) the new combination of the original content module; (2) original content module is divided into several new content modules; and (3) several new clauses are constructed.

As for the new clauses, this paper used the following strategies to construct the content module: if the new reference template $S_{\rho+1}$ contained σ new clauses, each new clause $D_i \in S_{\rho+1}(i \in [m+1, m+\sigma])$ was constructed as a new content module. When the new cases were accumulated for a period of time, Algorithm 2 would be used to determine whether these content modules containing only one clause could be merged into one content module; For each original clause $D_i(i \in [1,m])$, using step 2 of Algorithm 3 to repartition the content module, the result of partition was uncertain, it might split one initial content module into several new content modules, or kept the original content module of this clause unchanged, thus new content module $DC_q^{'}(q \in [1, Q])$ after learning is:

$$DC_q^{'} = \begin{cases} D_i & (i \in [m+1, m+\sigma]) \\ DC_p & (\forall D_i, D_j \in DC_p, d_{i(n+1)} = d_{j(n+1)}, i, j \in [1, m]) \\ D_i, D_j & (\forall D_i, D_j \in DC_p, d_{i(n+1)} = 0, d_{j(n+1)} = 1, i, j \in [1, m]) \end{cases} \tag{10}$$

- Content module value learning method

If no new content module was generated in case revision, the calculation of content module value was the same as that of case reuse; If a new content module constructed by the new clauses was generated, the frequency of that content module was 1; If the initial content module was divided into several new content modules, then the frequency of the new content module was increased by 1 on the basis of the frequency of the original content module, thus new content module value $V_C{}^{'}(DC_q^{'})$ is:

$$V_C{}'(DC_q') = \begin{cases} V_C(DC_p) * n/(n+1) & (DC_q' \in DC_p, DC_q' \notin S_{\rho+1}, q \in [1, Q-\sigma]) \\ (V_C(DC_p) * n + 1)/(n+1) & (DC_q' \in DC_p, DC_q' \in S_{\rho+1}, q \in [1, Q-\sigma]) \\ 1/(n+1) & (q \in [Q-\sigma, Q]) \end{cases} \quad (11)$$

• Clause difference degree learning method

The difference degree between the original clause and the new clause need to be recalculated by Equation (4); The difference degree between the original clauses could be obtained by the difference degree of the original clauses $Sim_{doc}(D_i, D_j)$ and the original clauses usage of new problems; As the frequency of the new clause was 1, the whole difference degree between new clauses was 0. Thus, the new clause difference degree $Sim'_{doc}(D_i, D_j)$ is:

$$Sim'_{doc}(D_i, D_j) = \begin{cases} \sqrt{\frac{((Sim_{doc}(D_i,D_j))^2 * n + |d_{i(n+1)} - d_{j(n+1)}|^2)}{n+1}} & (i, j \in [1, m]) \\ \sqrt{\frac{\sum_{k=1}^{n+1} |d_{ik} - d_{jk}|^2)}{n+1}} & (i \in [1, m], j \in [m+1, m+\sigma]) \\ 0 & (i, j \in [m+1, m+\sigma]) \end{cases} \quad (12)$$

• Content module difference degree learning method (The same as case reuse)

• Recommended sequence of content module learning method (The same as case reuse)

4 Results and discussion

In order to verify the effectiveness of the recommendation algorithm, this paper will carry out experiments on the bidder qualification requirements document unit based on four steps of case-based reasoning. We selected 144 tender documents of an enterprise as the data source for the experiment, and randomly selected 25% of the tender documents as the test set data, namely C, and the remaining 75% of the tender documents as the training set data.

Table 5: Recommended results of case reuse

New problem ID	S_{rec}	**Actual solution**	**If recommend successfully?**	**New problem ID**	S_{rec}	**Actual solution**	**If recommend successfully?**
1	S_3, S_1, S_4, S_6	S_8	0	19	S_3, S_1, S_4, S_6	S_3	1
2	S_1, S_2, S_5, S_3	S_1	1	20	S_3, S_1, S_4, S_6	S_{16}	0
3	S_1, S_2, S_5, S_3	S_1	1	21	S_3, S_1, S_4, S_6	S_{17}	0
4	S_1, S_2, S_5, S_3	S_1	1	22	S_1, S_2, S_5, S_3	S_1	1
5	S_3, S_1, S_4, S_6	S_3	1	23	S_1, S_2, S_5, S_3	S_2	1
6	S_3, S_1, S_4, S_6	S_3	1	24	S_3, S_1, S_4, S_6	S_{18}	0
7	S_1, S_2, S_5, S_3	S_1	1	25	S_3, S_1, S_4, S_6	S_3	1
8	S_3, S_1, S_4, S_6	S_{20}	0	26	S_3, S_1, S_4, S_6	S_{19}	0
9	S_3, S_1, S_4, S_6	S_1	1	27	S_1, S_2, S_5, S_3	S_1	1
10	S_3, S_1, S_4, S_6	S_3	1	28	S_1, S_2, S_5, S_3	S_1	1
11	S_3, S_1, S_4, S_6	S_4	1	29	S_1, S_2, S_5, S_3	S_1	1
12	S_3, S_1, S_4, S_6	S_{10}	0	30	S_1, S_2, S_5, S_3	S_1	1
13	S_3, S_1, S_4, S_6	S_{11}	0	31	S_3, S_1, S_4, S_6	S_{21}	0
14	S_3, S_1, S_4, S_6	S_{12}	0	32	S_3, S_1, S_4, S_6	S_{22}	0
15	S_1, S_2, S_5, S_3	S_{13}	0	33	S_1, S_2, S_5, S_3	S_{14}	0
16	S_3, S_1, S_4, S_6	S_4	1	34	S_3, S_1, S_4, S_6	S_3	1
17	S_3, S_1, S_4, S_6	S_{15}	0	35	S_3, S_1, S_4, S_6	S_3	1
18	S_1, S_2, S_5, S_3	S_1	1	36	S_3, S_1, S_4, S_6	S_9	0

In case retrieval algorithm (Algorithm 1), for each new problem C_0, we finally chose the average difference degree of tender case characteristics $\overline{Sim}_{case}(C_0, C_i)$ as the judgment threshold

λ_{case}, by analyzing the results of difference degree of tender case characteristics, and the cases satisfying $Sim_{case}(C_0, C_i) > \overline{Sim}_{case}(C_0, C_i)$ are merged into the similar case set C_{sim} of C_0.

In the case reuse algorithm (Algorithm 2), seven different tender case classes and reference templates $\{S_1, S_2, S_3, S_4, S_5, S_6, S_7\}$ were divided by non-interference sequence index in the training set. The recommended sequence S_{rec} of reference templates of all problems were obtained by calculation, and the length of them all were exactly 4, as shown in Table 5, where 1 represented success, 0 represented fail. "If recommend successfully?" means the recommendation in case reuse to be successful if "actual solution" is involved in S_{rec}. Thus, compared with the actual reference templates, the success rate of tender case content recommendation in case reuse was 58.33%. In addition, 15 new reference templates were found in the test set case, and defined them as $S_8, S_9, \cdots, S_{21}, S_{22}$. Next, the cases recommended unsuccessful in case reuse will be put into the case revision algorithm for experiment.

In the case revision algorithm (Algorithm 3), firstly, eight content modules (the number of abscissa is the subscript of each content module, such as, 1 means DC_1 were divided by non-interference sequence, and the similarity clustering of content modules was performed by hierarchical clustering method (as shown in Figure 6). In the clustering of content modules, the content modules can be synthesized into a content module class after 5 steps, and the similarity distance of the each step was 0.137, 0.193, 0.236, 0.373 and 0.918, respectively. We found that when distance less than 0.373, the similarity of content module classes is higher, so two content module classes are formed finally: $DM_1 = \{DC_1, DC_2, DC_4, DC_5\}$, $DM_12 = \{DC_3, DC_6, DC_7, DC_8\}$. The content modules within the same class could be recommended to each other. The recommended sequence of content modules could be seen in column $D_{rec}(p)$ of Table 6.

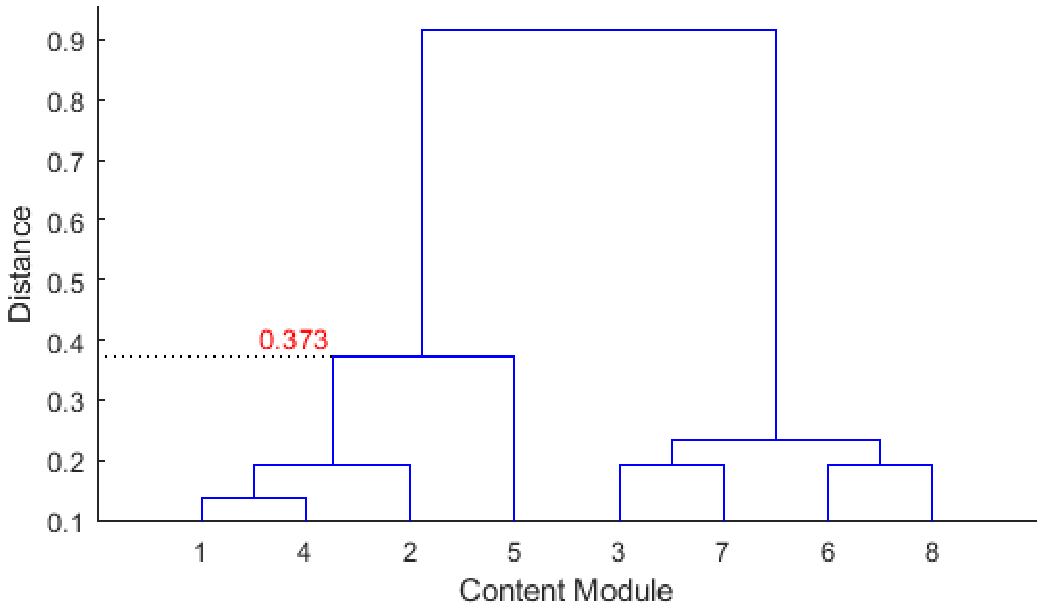

Figure 6: Hierarchical clustering of content modules

Secondly, the remaining 15 test cases were recommended by case revision algorithm, and the results were shown in Table 7 (the number in the second column and the third column is the subscript of each content module, such as, 1 means DC_1).

In this section, the content in second column and the third column could be compared to identify if the recommendation is successful. For example, as for "new problem 13", its recommended sequence of reference templates and content modules are "$S_3\{1, 2, 4\}$, $1\{1, 2, 4, 5\}$, $S_4\{1, 2, 4, 6\}$, $S_6\{1, 2, 3, 6, 8\}$". Therefore, in order to obtain its actual usage of content modules $\{1, 7\}$, from the first reference template S_3, we try to replace (delete/insert) its content module according to the $D_{rec}(p)$ in Table 6. The recommendation reference template $S_4\{1, 2, 4, 6\}$ was selected finally, and the detail operation was: DC_1 was retained, DC_2 and DC_4 were deleted, DC_6 was replaced by DC_7, as DC_6 and DC_7 are in the same content module class that could be recommended to each other. Besides, the position where the font color different corresponds to the recommended position of the recommended reference templates in Table 7. It also should be noted that this part of the operation is artificial processing performed by the personnel based on the recommended results.

Table 6: Content module difference degree matrix and recommended sequence

	$Sim_{doc}(DC_p, DC_r)$								$V_C(DC_p)$	$D_{rec}(p)$
	DC_1	DC_2	DC_4	DC_5	DC_3	DC_6	DC_7	DC_8		
DC_1	0	0.236	0.136	0.373	0.991	0.962	0.991	0.981	1	DC_1, DC_4, DC_2, DC_5
DC_2	0.236	0	0.192	0.441	0.981	0.933	0.981	0.953	0.944	DC_2, DC_4, DC_1, DC_5
DC_4	0.136	0.192	0	0.397	0.981	0.953	1	0.972	0.981	DC_4, DC_1, DC_2, DC_5
DC_5	0.373	0.441	0.397	0	0.918	0.967	0.918	0.948	0.861	DC_5, DC_1, DC_4, DC_2
DC_3	0.991	0.981	0.981	0.918	0	0.304	0.192	0.236	0.019	DC_3, DC_7, DC_8, DC_6
DC_6	0.962	0.933	0.953	0.967	0.304	0	0.304	0.192	0.074	DC_6, DC_8, DC_3, DC_7
DC_7	0.991	0.981	1	0.918	0.192	0.304	0	0.236	0.019	DC_7, DC_3, DC_8, DC_6
DC_8	0.981	0.953	0.972	0.948	0.236	0.192	0.236	0	0.037	DC_8, DC_6, DC_3, DC_7

The reference templates and their target content module could be obtained to be revised by dividing the reference template into content modules, then the content recommendation of the tender case was realized finally by replacing target content module with the content modules in the recommended sequence in Table 7. As for "new" in column "Actual usage of content modules", they were new content modules constructed by new clauses that did not exist in the historical tender documents, and would not be involved in this recommendation. Instead, the "new" could be studied into the tender case database as new tender knowledge that was called case learning. So far, combined with the case reuse, the final success rate of all recommendations has reached 94.44%, the recommendation effect was significant.

Table 7: Recommended results of case revision

ID	**Actual usage of content modules**	**Recommended reference samples and its content modules**	**If recommend successfully?**	**Recommended location**
1	{1,new}	S_3{1,2,4},S_1{1,2,4,5},S_4{1,2,4,6},S_6{1,2,3,6,8}	1	1
8	{1,7,new}	S_3{1,2,4},S_1{1,2,4,5},S_4{1,2,4,6},S_6{1,2,3,6,8}	1	3
12	{1,2,4,6,new}	S_3{1,2,4},S_1{1,2,4,5},S_4{1,2,4,6},S_6{1,2,3,6,8}	1	3
13	{1,7}	S_3{1,2,4},S_1{1,2,4,5},S_4{1,2,4,6},S_6{1,2,3,6,8}	1	3
14	{6,new}	S_3{1,2,4},S_1{1,2,4,5},S_4{1,2,4,6},S_6{1,2,3,6,8}	1	3
15	{1,4,new}	S_1{1,2,4,5},S_2{1,3,4,5},S_5{1,5,7},S_3{1,2,4}	1	1
17	{1,4,6,8,new}	S_3{1,2,4},S_1{1,2,4,5},S_4{1,2,4,6},S_6{1,2,3,6,8}	1	3
20	{1,4,6,8,new}	S_3{1,2,4},S_1{1,2,4,5},S_4{1,2,4,6},S_6{1,2,3,6,8}	1	3
21	{1,2,6,new}	S_3{1,2,4},S_1{1,2,4,5},S_4{1,2,4,6},S_6{1,2,3,6,8}	1	3
24	{1,4,6,8,new}	S_3{1,2,4},S_1{1,2,4,5},S_4{1,2,4,6},S_6{1,2,3,6,8}	1	3
26	{1,2,4,6,new}	S_3{1,2,4},S_1{1,2,4,5},S_4{1,2,4,6},S_6{1,2,3,6,8}	1	3
31	{new}	S_3{1,2,4},S_1{1,2,4,5},S_4{1,2,4,6},S_6{1,2,3,6,8}	0	
32	{1,6,new}	S_3{1,2,4},S_1{1,2,4,5},S_4{1,2,4,6},S_6{1,2,3,6,8}	1	3
33	{1,3,5,new}	S_1{1,2,4,5},S_2{1,3,4,5},S_5{1,5,7},S_3{1,2,4}	1	2
36	{new}	S_3{1,2,4},S_1{1,2,4,5},S_4{1,2,4,6},S_6{1,2,3,6,8}	0	

Case learning is a means to expand and update the case database, and it is also an important condition to ensure that the CBR algorithm can keep effective and reliable over time. After a new problem is solved, this target case and its solution shall form a new case, which can be stored in the case database and used as an existing case to solve the new problem in the future. With the continuous growth of case database, this paper's recommendation method will become more and more reliable.

5 Conclusion

Aiming at the compilation problem of tender documents, this paper found that its core activities were to collect similar historical tender documents, select compilation templates of tender documents and revise templates of tender documents partially. However, when the historical tender documents have accumulated to a certain amount, it becomes extremely difficult for compilers to select and revise templates subjectively.

Based on the theory of case-based reasoning and data mining technology, contributions achieved are as follows: retrieval of similar tender cases of target cases, reuse and recommendation of reference templates of similar tender cases; if they could not be reused, the reference templates are further revised, that is, the reference templates are divided by content modules, and the local content modules are recommended; finally, after all the recommendation processes are completed, the project is implemented example learning process, all tender knowledge of the target case is updated into the database. The results of real data show that the accuracy rate of the content recommendation of the tender document is as high as 94.44%, and the recommendation effect is remarkable, and the compilation process of the tender document is more efficient and standardized.

In the existing literature, there are not many optimization methods for the compilation of tender documents, so this paper can enrich the recommendation methods of the content of tender documents to a certain extent. Next, this paper will focus on the confirmation of the tender documents, and introduce the evaluation system of the content recommendation algorithm of the tender documents, so as to further improve the content construction system of tender documents.

References

[1] Adeyanju, I. (2012). Generating weather forecast texts with case based reasoning, *International Journal of Computer Applications*, 45(10), 35–40, 2012.

[2] Brook, M. (2004). *Estimating and tendering for construction work*, Butterworth-Heinemann, 2004.

[3] Cook, A.E. (1991). *Construction tendering: theory and practice*, BT Batsford, 1991.

[4] Chen, J.; Xue, G. (2006). How to avoid common problems in the compilation of tender documents, *Coal Economy Research*, (08), 42–44,2006.

[5] Compilation group of standard documents (2007). *Standard construction tender documents of the people's Republic of China*, China Planning Press, 2007.

[6] Fang, W.J.; Li, S.W.; Yuan, Y.; Wang, W.W. (2005). The research and application of case—based reasoning technology, *Agricultural Network Information*, (01), 13–17, 2005.

[7] Ho, C.; Chen, J.J. (2006). The Case-Based Reasoning System for Knowledge Management of Engineering Consulting Projects, *2006 International Conference on Computational Inteligence for Modelling Control and Automation and International Conference on Intelligent Agents Web Technologies and International Commerce (CIMCA'06)*, 234–234, 2006.

[8] Jiang, Z.S.; Chen, L.P.; Luo, N.M. (2007). Similarity analysis in nearest-neighbor case retrieval, *Computer Integrated Manufacturing Systems*, 13(06), 1165–1168, 2007.

[9] Leung, Y.T.; Parija, G.R.; Sharma, S. (2010). *Diagnosis of equipment failures using an integrated approach of case based reasoning and reliability analysis*, US, 2010.

[10] Li, F.G. (2007). Research on intelligent decision making technology based on optimized case-based reasoning, PhD thesis, Hefei University of Technology, 2007.

[11] Ling, J.Z. (2019). Compilation of tender documents, *Science and Wealth*, 000(012), 122, 2019.

[12] Padraig, C.; Barry, S. (1994). A comparison of model-based and incremental case-based approaches to electronic fault diagnosis, *Respiratory Medicine*, 94(12), 1184–1191, 1994.

[13] Rissland, E.L.; Skalak, D.B. (1989). Combining case-based and rule-based reasoning: a heuristic approach, *Proceedings of the 11th International Joint Conference on Artificial Intelligence - Volume 1 (IJCAI'89)*, 524–530, 1989.

[14] SAMUEL, L. (2011). Quality of tender documents: case studies from the UK, *Construction Management & Economics*, 29(1-3), 275–286, 2011.

[15] Sun, A.H. (2019). Measures to improve the compilation process of enterprise tender documents, *Tax Paying*, 13(03), 291–292, 2019.

[16] Tang, J. (2019). Ideas and methods of machine writing tender documents, *Tender Procurement Management*, (09), 43–46, 2019.

[17] Wang, R. (2013). Design and development of electronic tender documents compilation system—Exploration of Baohua Tender's Second Generation Electronic Tender Platform, *China Tender*, (31), 22–25, 2013.

[18] Wu, S.; Wang, Q.; Jiang, M.; Wei, Q. (2013). Clustering algorithm of categorical data in consideration of sorting by weight, *Journal of University of Science and Technology Beijing*, 35(08), 1093–1098, 2013.

[19] Xiong, K.L. (2020). Research on machine writing based on deep learning, Master's thesis, Shanghai Normal University, 2020.

[20] Ye, H.Y. (2019). Problems and countermeasures in the compilation of project tender documents, *Architecture, building Materials and Decoration*, 000(008), 141, 109, 2019.

[21] Yu, J.X.; Zhang, C.; Wu, J.Q. (2019). Design and development of tender document compilation system for power industry engineering services, *Enterprise Reform and Management*, 000(002), 206–207, 2019.

[22] Zhao, W.G. (2014). The importance of the consolidation of tender document model in electronic tendering, *China Science and Technology Vertical and Horizontal*, 000(014), 233–234, 2014.

Cite this paper as:

Zhou, T.T.; Gao, X.D.; Wei, G.Y. (2021). Construction Method of Tender Document Based on Case-based Reasoning, *International Journal of Computers Communications & Control*, 16(3), 4170, 2021.
https://doi.org/10.15837/ijccc.2021.3.4170