Front. Struct. Civ. Eng. 2020, 14(6): 1403–1417
https://doi.org/10.1007/s11709-020-0666-8

RESEARCH ARTICLE

Heuristic solution using decision tree model for enhanced XML schema matching of bridge structural calculation documents

Sang I. PARK[a,b], **Sang-Ho LEE**[b*]

[a] *Department of Civil, Environmental and Architectural Engineering, University of Colorado at Boulder, Boulder, CO 80309, USA*
[b] *Department of Civil and Environmental Engineering, Yonsei University, Seoul 03722, Korea*
[*] *Corresponding author. E-mail: lee@yonsei.ac.kr*

© Higher Education Press 2020

ABSTRACT Research on the quality of data in a structural calculation document (SCD) is lacking, although the SCD of a bridge is used as an essential reference during the entire lifecycle of the facility. XML Schema matching enables qualitative improvement of the stored data. This study aimed to enhance the applicability of XML Schema matching, which improves the speed and quality of information stored in bridge SCDs. First, the authors proposed a method of reducing the computing time for the schema matching of bridge SCDs. The computing speed of schema matching was increased by 13 to 1800 times by reducing the checking process of the correlations. Second, the authors developed a heuristic solution for selecting the optimal weight factors used in the matching process to maintain a high accuracy by introducing a decision tree. The decision tree model was built using the content elements stored in the SCD, design companies, bridge types, and weight factors as input variables, and the matching accuracy as the target variable. The inverse-calculation method was applied to extract the weight factors from the decision tree model for high-accuracy schema matching results.

KEYWORDS structural calculation document, bridge structure, XML Schema matching, weight factor, data mining, decision tree model

1 Introduction

The structural calculation document (SCD) of a bridge structure is generated during its design, and the information contained in the document is used as an essential reference during the entire lifecycle of the structure. However, a collaboration between various companies and experts is necessary due to the nature of large-scale civil engineering projects, the generators and users of information are often different. In the engineering field, the users often need the parts of the content information included in the documents rather than the information of several documents having similar contents as pointed out by Liu et al. [1]. It needs a technique of extraction, retrieval, or structuralization of document items in order to cope with these features effectively.

The extraction and retrieval of the necessary information from engineering documents are mainly progressing in the field of legal or regulatory checking. Because this process has to be utilized a lot of document information and complex regulations, and it is still performed manually in most parts, thus requiring time-consuming and repetitive works. Tan et al. [2] proposed a way to reduce time, cost, and errors through Automated Compliance Checking (ACC). Zhong et al. [3] studied for quality compliance of buildings using Web Ontology Language (OWL) and Semantic Web Rule Language (SWRL). Zhang and El-Gohary [4] conducted a study to improve the reasoning of information extraction by applying a rule-based natural language process to construction regulatory documents. Epistemology is also being applied for information retrieval. Refs. [5,6] conducted initial step research on information retrieval and management by applying web retrieval technology to the AEC domain. Zhang and

Article history: Received Feb 5, 2020; Accepted Feb 17, 2020

El-Gohary [7] proposed a context-aware semantic model for sustainable construction practices and applied to the practical field [8]. Automating design reviews using artificial intelligence (AI) have also been attempted [9].

Unlike the case by applying information retrieval or extraction method to the generated document itself, there are also attempts to improve the information utilization by changing the external format of the document. It is because most of the engineering documents in construction projects, including bridge SCDs, are in an unstructured information format, and therefore it is hard to retrieve the required information as indicated in Refs. [1,10]. Accordingly, many researchers realized that the structuralization of information is an essential factor in the improvement of productivity of the construction industry. Since 2000, studies have been actively conducted on the structuralization of engineering documents in the construction industry. Ma et al. [11] investigated the efficient information exchange in construction projects by using the Extensible Markup Language (XML), which is a description language that can express structuralized information. Park et al. [12] presented a general process for structuralization of unstructured engineering documents based on explicit semantic information, and Kim et al. [13] proposed a mathematical foundation for structuralization of bridge SCDs by using an apparent semantic structure, which was used to convert an SCD into an XML document, thereby contributing to its efficient structuralization. Particularly, the structuralized engineering documents can maximize the accuracy and efficiency of information exchange in each of the sub-fields of construction projects by combining semantic information with the Building Information Model (BIM), which is central to the information exchange in the construction field. For example, Lee et al. [14] emphasized the importance of using document information as a strategy for efficient operation, maintenance, and management of bridge structures, and Kim [15] investigated the linkage between the CAD model and SCD information of steel bridges to increase the practical applicability. However, the structuralization of this information is valuable only if the high quality of the collected data can be ensured. This was the essential motive of this study. The SCDs are the important references should be storing the reliable contents as well as the accessibility of the information, which is the critical factors affecting the post-design phases. It is, however, almost impossible to review the quality of data in accumulated large SCDs manually. Therefore, the authors intended to review the quality of the SCDs contents trough standard process automatically. A schema matching is an excellent technique to solve the mentioned problems.

Schema matching is a process that identifies the semantic relationship between two or more schema components. This process enables qualitative improvement of the stored data owing to a high-quality basic schema, as presented in Ref. [16]. Ref. [17] proposed a method that can be applied to the standardization of the information items contained in a bridge SCD by using the method proposed by Yi et al. [18] for applying the schema matching technique into engineering documents in the construction industry. Furthermore, Ref. [19] proposed a simplified matching method to resolve the inefficiency in the matching schema speed for the XML application schema matching technique proposed by Ref. [18], which has a complex hierarchy, such as that in bridge SCDs, and contains many similar elements. However, to obtain a high accuracy of schema matching in the method proposed by Park et al. [19], the element weight factors in the XML Schema matching corresponding to each situation are required. Moreover, two or more weight factors, not one, are present in this case, which makes it difficult to apply to the type parameter optimization process, and a direct correlation cannot be derived between the element weight factors and the matching accuracy because schema matching has the main function of determining the semantic relationship between elements whose relationship information is absent or lost. Thus, the matching techniques can be used the concept of the equality constraint method presented in Ref. [20] to estimate the optimal element weight factors under these conditions, fixing the remaining weight factors, except for one, to a constant value and experimentally performing an optimization process to calculate these element weight factors. However, the excessive time consumption in this experiment decreases its practical applicability.

This study aimed to enhance the practical applicability of XML Schema matching, which improves the speed and quality in storing the information contained in bridge SCDs. The practical applicability of XML Schema matching basically requires computing speed and accuracy. Thus, this study analyzed the limitations in using typical XML Schema matching and proposed improvement methods to apply it to bridge SCDs efficiently. To maintain high matching accuracy, this study proposed a heuristic solution for selecting the optimal weight factor for the matching process by introducing a decision tree model, which is one of the data mining techniques. A pilot study was conducted by using bridge SCDs commonly employed in practice.

2 Extensible Markup Language schema matching for structural calculation documents

2.1 Extensible Markup Language schema matching techniques and simplified method for structural calculation documents

Schema matching is a mapping technique that identifies a semantic relationship between two schema models. Schema matching has been playing a key role in schema integration, data warehouses, E-commerce, and semantic query processing, and this process has been performed manually or automatically since the 1980s. Particularly, Ref. [16] states that the application of the schema matching technology can improve the quality of stored data by converting information according to a standardized structure. Accordingly, various types of automated schema matching techniques have been studied. Matching methods that use the elements that constitute the schema have been proposed by utilizing constraints on the element types, along with linguistic features such as element names and descriptions [21–25]. To use the schema structure for matching, a graph matching is applied to relations such as supertype/subtype, sibling, or neighbor, and representative studies include Refs. [26,27]. When the schema itself is incomplete, the schema lacks detailed information, or the number of instances generated by the schema is sufficiently large, the contents can be used for matching, such as the pattern or range of instances, word frequency, or key terms. Ref. [28] is a representative study on the instance-level approach. The types of schemas used in this case include various types such as text, database, ER model, XML model, and graph model. The target source in this study is a bridge SCD, and XML-based schema matching was applied because XML is the most popular language in information exchange, the XML structure is easy to understand, and the bridge SCD has a deep hierarchy. Among such techniques, the XML application schema matching technique proposed by Ref. [18] is easy to apply to large-scale engineering documents because this technique does not consider the data type of the elements contained in the schema. As shown in Fig. 1, the XML application schema matching technique basically consists of two processes: the semantic similarity measuring process between the elements by using two XML Schemas, target and source, and the relaxation labeling process to consider the relationship between elements.

In this case, the similarity measuring process, which is the first step, is a process of quantifying the similarity between one element of the source schema and the elements of the target schema in the range of 0 to 1 based on the element name. The quantitative similarity value comprehensively considers the similarity of the target element to the parent element (P), sibling element (S), and child element (C), along with the name comparison for one element (NE), which is represented in Eq. (1):

$$SM(a,b) = \sum_{x} \omega_x \cdot Q_x(a,b), \tag{1}$$

where a and b refer to an element of the source schema and an element of the target schema, respectively, and $\{x \in NE,P,S,C\}$. Q refers to the degree of similarity between the target elements, $Q_{\mathrm{NE}}(a,b)$ refers to the degree of similarity between one element a of the source schema and one element b of the target schema, and $Q_{\mathrm{S}}(a,b)$, $Q_{\mathrm{C}}(a,b)$, $Q_{\mathrm{P}}(a,b)$ refer to the similarities of a and b to the sibling, child, and parent elements. In this case, ω is a similarity weight factor corresponding to Q, having a value of 0 to 1, and $\sum_{x} \omega_x = 1$. According to Park et al. [19], the optimal value of ω for obtaining high accuracy according to the target document for XML Schema matching is changed, and direct correlation between ω and matching accuracy cannot be established.

Relaxation labeling, which is the second step of the XML application schema matching, is a process that reflects the connection relationship or structural constraints between elements. The matching reliability matrix can be expressed as Eq. (2):

$$P^{(t+1)}(m,k) = \frac{P^{(t)}(m,k)q^{(t)}(m,k)}{\sum_{u=1}^{v} P^{(t)}(m,u)q^{(t)}(m,u)}, \tag{2}$$

where t refers to a repeated order, m and k refer to elements of the source schema and target schema, respectively, and v refers to the total number of elements in the target schema. The support function q for quantifying the structural constraints and distances is expressed in the following Eq. (3):

$$q(m,k) = \sum_{n=1}^{w}\sum_{u=1}^{v} \gamma_{mn}(k,u)P(n,u), \tag{3}$$

where w refers to the total number of elements in the source schema and γ_{mn} refers to the quantitative value of similarity according to the structural linkage. Yi et al. [18] expressed

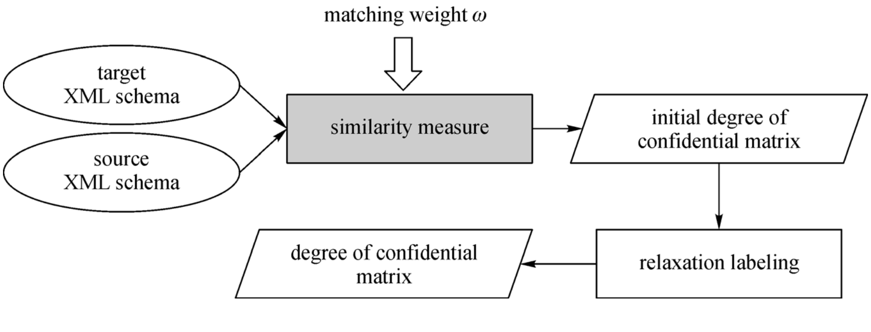


Fig. 1 Basic process of XML application schema matching.

this relationship with Eq. (4):

$$\begin{cases} \gamma_{mn}(k,u) = 1, & \text{if } e(m,n) = 1 \wedge e(k,u) = 1, \\ \gamma_{mn}(k,u) = \dfrac{1}{d_{mn} + d_{ku}}, & \text{otherwise,} \end{cases} \tag{4}$$

where $e(x,y)$ is a symbol representing a structural relationship between two nodes, x and y, having a value of 1 in the case of a direct connection that is a parent-child relationship. When indirectly connected, the values differ according to the distance. Consideration of the relationship between these structural distances can improve the accuracy when the targeted documents are of a relatively small scale. However, when the number of elements in a document exceeds 1000 units or the documents have a deep hierarchy, the efficiency with respect to the computation time drops sharply. Particularly, in the case of a large section with a similar pattern, such as a bridge SCD that is repeated many times, this could be reflected in the improvement of the computing speed. Accordingly [19], proposed Eq. (5) to simplify Eq. (4):

$$\psi_{mn}(k,u) = \begin{cases} 1, & \text{if } m = n \wedge k = u, \\ 0, & \text{others.} \end{cases} \tag{5}$$

The elements on the structural relationship appearing in Eq. (5) are complemented by adjusting the weight factor shown in Eq. (1).

2.2 Selection of weight factors of Extensible Markup Language schema matching for structural calculation documents

The authors of this study experimentally derived the weight factor of a similarity measure suitable for bridge SCDs. The suitable weight factor was based on the accuracy of the final schema matching, as expressed in Eq. (6):

$$accuracy = \frac{n(TP) + n(TN)}{n(TP) + n(TN) + n(FP) + n(FN)}, \tag{6}$$

where $n(X)$ refers to the total number of corresponding elements, and TP, TN, FP, and FN refer to true positive, true negative, false positive, and false negative, respectively.

According to Ref. [19], the four factors ω_{NE}, ω_{S}, ω_{C}, and ω_{P} of the element weight factor in the similarity measure have no correlation, except when the sum is 1, and the change of ω shows no general association with the accuracy of schema matching. Thus, this study used the concept of the equality constraint method presented by Lin [20], as shown in Fig. 2, to fix the remaining weight factors, except for one, and experimentally perform the optimization process to calculate the element weight factors.

In the method in Fig. 2, the order of i does not typically influence the determination of the weight factor. Because the weight factors selected in the preceding experiment are

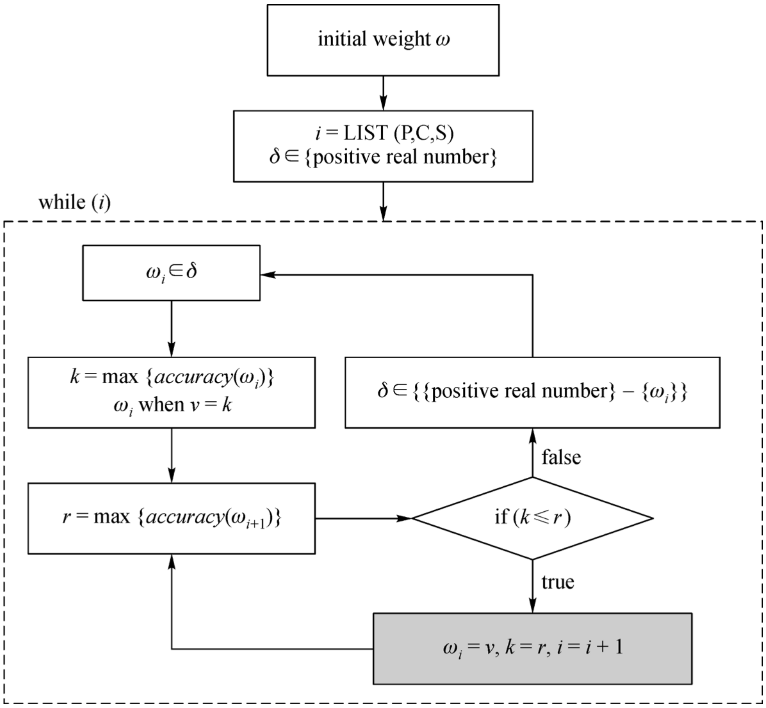

initial weight ω
i = LIST (P,C,S)
δ ∈ {positive real number}
while (i)
ω_i ∈ δ
k = max {accuracy(ω_i)}
ω_i when v = k
δ ∈ {{positive real number} − {ω_i}}
false
r = max {accuracy(ω_i+1)}
if (k ≤ r)
true
ω_i = v, k = r, i = i + 1

Fig. 2 Equality constraint method-based weight selection process.

included in the following experiment, the accuracy of the following experiment result does not become lower than that of the earlier experiment. Accordingly, the determination of the first element weight factors tends to influence the determination of the approximate range of the accuracy measurement. Nevertheless, this finding would be useful in the experiment to select weight factors within an undefined range, because the advantage of this technique is that results with high accuracy can be obtained with relatively few experiments conducted with various possibilities.

In this manner, Fig. 3(a) shows the changes in the average of accuracy and accuracy of matching while changing the ratio of ω_P and maintaining the ratios of ω, other than ω_P, as constant.

Figure 3(a) shows that the matching accuracy is high when the ratio of ω_P is low. In this experiment, ω_{NE}:ω_S:ω_C:ω_P = 1:1:1:1/2 is the most appropriate distribution of the element similarity weight factors. Figure 3(b) shows the change in matching accuracy according to the change in ω_C. The ratio of ω_{NE} to ω_S was fixed at 1, the ratio of ω_P was fixed at 1/2, and ω_C was varied. In this case, the ω_C ratio of 1 indicates the highest matching accuracy. Figure 3(c) shows the average accuracy changes according to the change in the ratio of ω_S by applying the initial ω_{NE} values of 1 along with the ω_C ratio of 1 and ω_P ratio of 1/2 obtained from the previous experiment. In the case of ω_S, MM and SMM show a different behavior from that of the previous results. MM shows the highest matching accuracy when the ω_S ratio is 1/3 and SMM is 1. Under the condition that the weight factor ratios of MM and SMM should be the same, it is reasonable to use the ω_S ratio of 1/2. The ratio value of ω_{NE} is further determined according to the result of Fig. 3(c), which is further confirmed by the results shown in Fig. 4. As a result, the ratios of element similarity measure with the highest accuracy obtained from this experiment are ω_{NE}:ω_S:ω_C:ω_P = 1:1/3:1:1/2 for MM and ω_{NE}:ω_S:ω_C:ω_P = 1:1:1:1/2 for SMM.

2.3 Adaptability validations of the simplified Extensible Markup Language schema matching technique

As previously described, the difference between MM and SMM is the difference in the equation applied during the relaxation labeling process, which is an optimization process of element matching. As shown in Section 2.2, these results could influence the accuracy of matching, leading to a different setting of the element similarity weight factors. However, as can be seen in Figs. 3–4, the accuracy of the two modules can vary, and the maximum value of the accuracy is nearly the same, which indicates that there is no significant difference between the two modules for bridge SCDs in terms of accuracy.

However, significant differences are present in the performance of the two modules in terms of computing time. Figure 5 shows the comparison results on the computing speed of the two modules applied to the same model.

In Fig. 5, the dotted line graph shows the one-time computing time of the MM and SMM modules, and the solid line graph shows the computing time when the final matching reliability matrix is generated. When the number of elements in the model is small, the SMM result was approximately 13 times faster than the MM (MM: 0.2 s.,

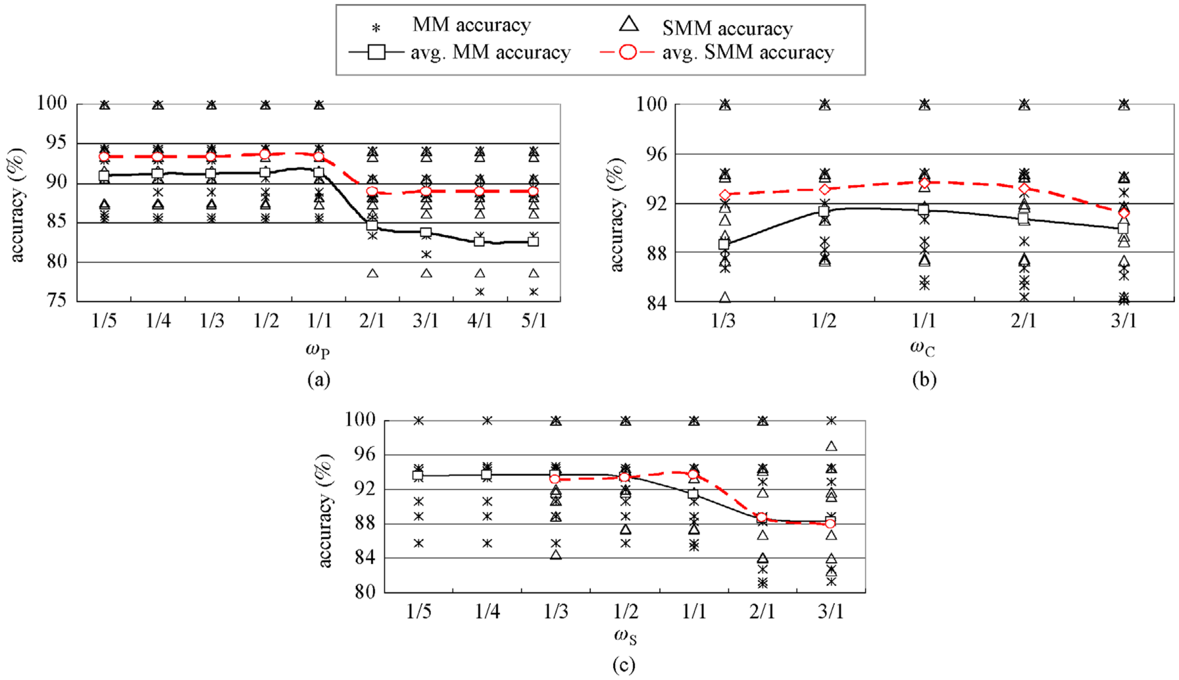


Fig. 3 Changes in the accuracy of XML Schema matching according to change in weight factors: (a) ω_P; (b) ω_C; (c) ω_S.

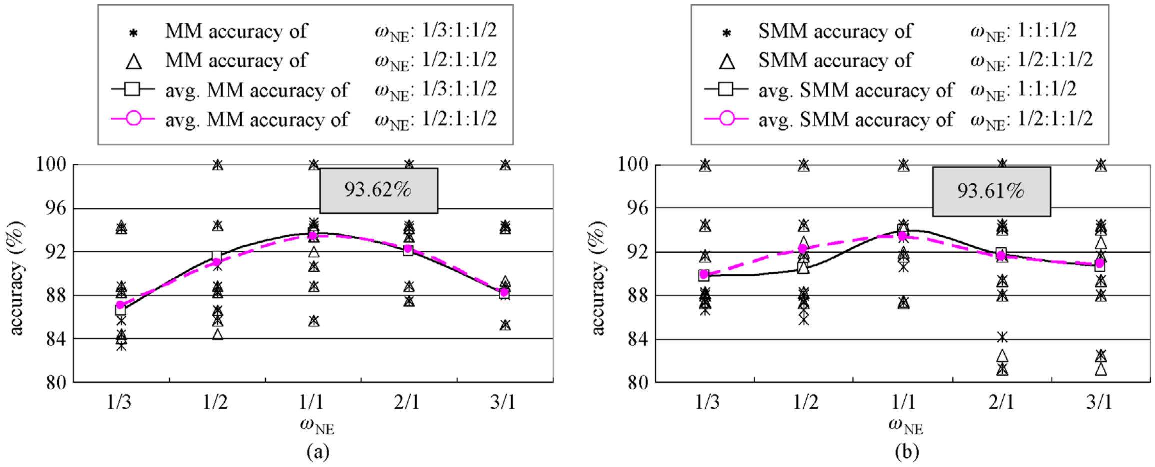


Fig. 4 Changes in the accuracy of XML Schema matching according to change in ω_{NE}: (a) MM module; (b) SMM module.

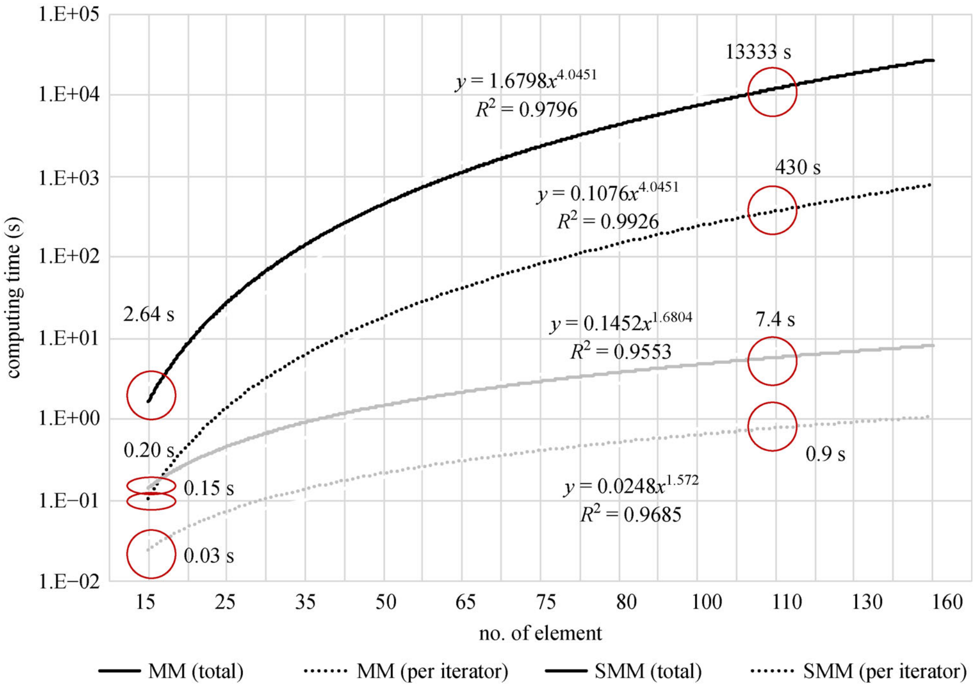


Fig. 5 Comparison of computing time between MM and SMM according to the number of elements in the model.

MM: 2.64 s.). However, as the number of elements increases, the SMM result becomes 1800 times faster or even more (SMM: 7.4 s, MM: 13333 s). This result suggests that the applicability can be appropriate for considering the inclusion of typical bridge SCDs.

Applying the described method, the SMM-based scheme matching was performed to select the weight factor of matching by using 20 bridge SCDs different from the previously used data. Figure 6 shows the results on matching accuracy for the bridge elements.

As shown in Fig. 6, the accuracy was maintained or improved in all items. When using an arbitrary matching weight factor, the average accuracy was approximately 82.7%, and when using the selected weight factor, ω_{NE}:ω_{S}:ω_{C}:ω_{P} = 1:1:1:1/2, the average accuracy was approximately 90.6%. In this case, the average number of matching repetitions was approximately 173 s per matching and the total computing time was approximately 1210 s.

3 Decision tree model-based efficient XML schema matching

Chapter 2 describes an effective application method of XML Schema matching to bridge SCDs. That method can

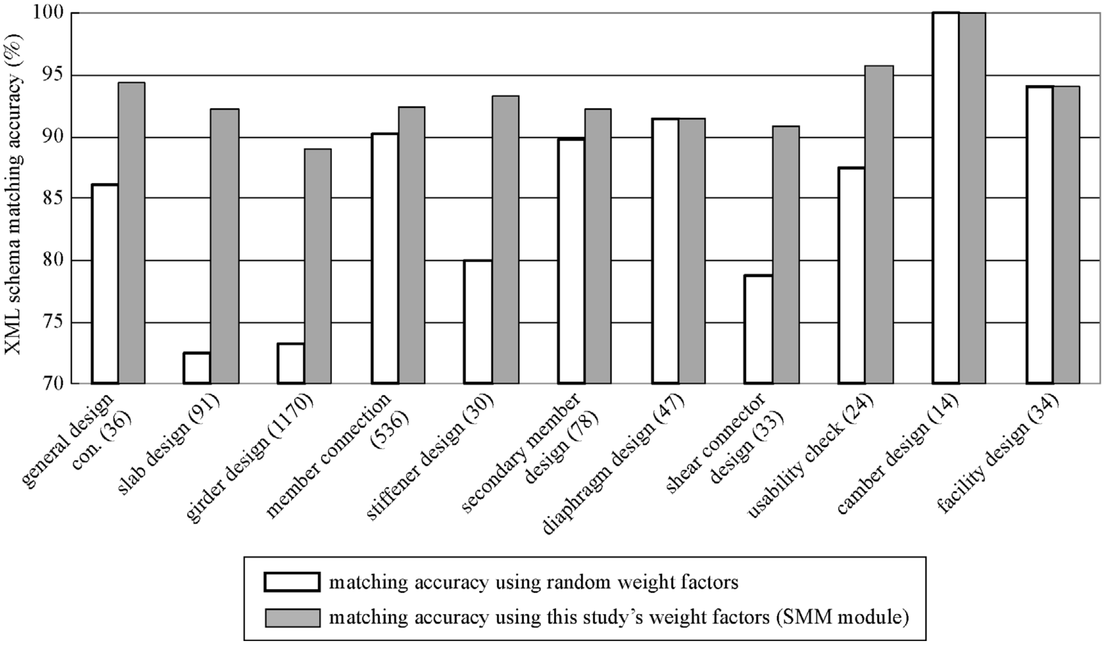


Fig. 6 Comparison of matching accuracy by SCD item.

be very effective when the document structure is deep, the document has numerous elements, and similar items appear repeatedly. However, significant time and efforts are required to calculate the matching weight factor for correcting the structural information of the items. This study investigated a method to derive an appropriate matching weight factor in a short period of time by using a data mining method based on the previously established schema matching weight factor and matching accuracy data.

3.1 Data mining and decision tree model

Typically, an optimal design is a process that derives the desired optimal result value (target variable) within the constraints satisfying a given condition (input variable). This study follows a similar process to the optimal design as it is aimed at selecting optimal values for the element-similarity weight factors in the similarity measuring process to improve the accuracy of XML Schema matching. However, as previously described, the relationship between the accuracy of the XML Schema matching and the similarity weight factor used at this time cannot be expressed through definite parameters, which indicates that the element-similarity weight factors cannot be a direct constraint for the XML Schema matching. For this reason, the optimal design method cannot be applied in this study.

According to Refs. [29,30], data mining is the process of discovery of new patterns within data by using various techniques, such as statistical techniques based on the previously collected data, as part of the Knowledge Discovery in Databases (KDD), which is a new knowledge discovery process and is shown in Fig. 7. The most important purpose of data mining is a prediction, and inductive knowledge can be constructed by using methods such as clustering, classification, and regression. Thus, data mining can be used to optimize the problem of obtaining an optimal solution under given constraints.

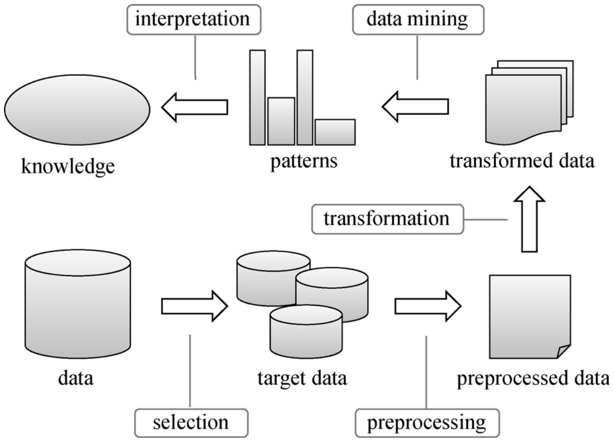


Fig. 7 KDD process.

The data mining classification methods typically include discriminant analysis, decision tree, and neural network. In this case, the decision tree constructs subgroups according to characteristics by using the relationship of input variables in the target group, and thereby selects the optimum solution of the target variable based on the collected data. The subgroups according to the characteristics are hierarchically classified from the root element to the bottom leaf, which is easy to interpret and highly applicable to the construction of the statistical model. The important results to be derived from this study are the high accuracy of the mapping and the element-similarity weight factor in the XML Schema matching process required to derive these results, which are independent of each other and cannot directly constrain the result. In this sense, the resulting values can be effectively applied to the decision tree. Thus, the decision tree was used as a basic model for efficiently calculating the element-similarity weight factors.

3.2 Decision tree model-based weight factor selection method of Extensible Markup Language schema matching

The type of input variables used in the decision tree can be variously composed of continuous and categorical variables, and the target variable is only derived as a categorical value. Furthermore, the decision tree calculates the results starting from the root and reaching the leaf as the target variable. Thus, it is difficult to use the general flow of the decision tree for selecting two or more variables. However, as previously described, it is necessary to select two or more variables because the input variable in the case where the target variable has the optimum value is what this study aims to derive as a result. Thus, this study used an inverse-calculation method that includes target variables to be calculated as input variables, as shown in Fig. 8, to calculate the optimal solution according to the conditions of several independent target variables as input variables, as shown in Fig. 8. The inverse-calculation method of the optimum solution proposed in this study basically applies the two-way utilization method rather than the one-way decision tree utilization. In this regard, if the node contains the contents of the input variable that can be judged, it follows the corresponding branch ([A] in Fig. 8). If a node contains an input variable that cannot be judged, after all the branches are temporarily saved ([B], [C] in Fig. 8), and a branch is selected according to the result of the final leaf and the corresponding solution is obtained ([D] in Fig. 8). This method provides the advantage of quickly deriving the optimal target variable and the corresponding input variable in the decision tree.

3.3 Generation of decision tree model for the selection of the matching weights

A decision tree model was constructed to calculate the optimum weight factor for XML Schema matching. The decision tree model uses 580 arbitrary scheme matching results from 20 types of bridge super-structure and sub-structure SCDs. Of the results, 60% were used as training data and 40% were used as validation data. Table 1 shows the variables for the model configuration.

In Table 1, the target variable refers to the accuracy obtained through the matching experiment using random weight factors. To represent the categorical value, the target variable was divided into six steps from A to F, and the number of elements in the entire document, bridge type, and document creation company, which are external features of the SCD, were added to create the model, and they were further used as a condition for weight factor selection. ω_{NE}, ω_{S}, and ω_{C} represent the weight factors for the name, sibling, and children elements used in XML Schema matching. In this study, the four weight factors used in XML Schema matching include the previously mentioned three elements and the parent element. However, ω_{P} was excluded because of the condition of $\sum\omega = 1$.

Based on the previous descriptions, this study constructed a decision tree for calculating the element weight factors for the XML Schema matching, as shown in Fig. 9.

In this case, for management efficiency and query

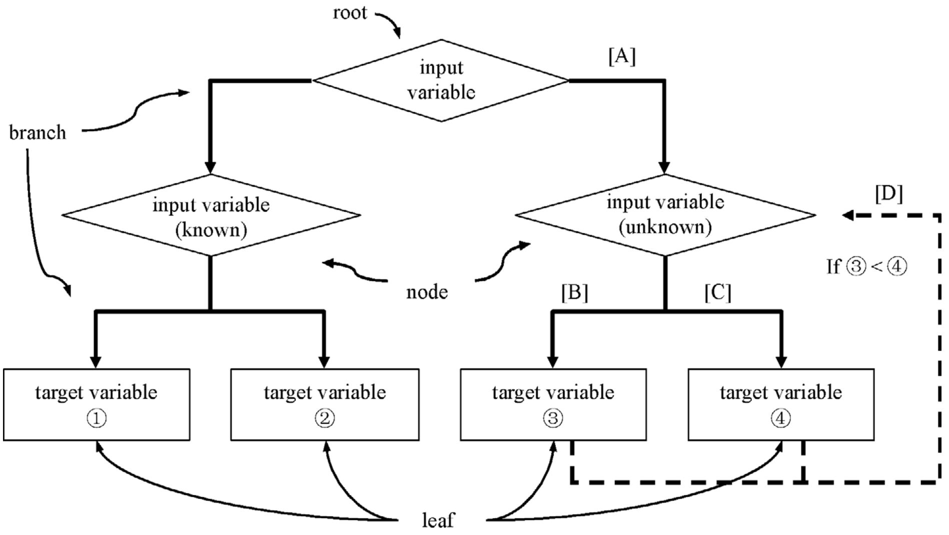


Fig. 8 Inverse-calculation applications of the decision tree.

Table 1 Variables used in the decision tree model configuration

item	variable	variable type and range
target var.	matching accuracy	A: 100% D: 85%–89% B: 95%–99% E: 80%–84% C: 90%–94% F:≤79%
input var.	ω_{NE}	continuous: 0–1
	ω_{S}	continuous: 0–1
	ω_{C}	continuous: 0–1
	no. of element	continuous
	structural type of bridge	cs: cable-stayed bridge sb: steel box girder bridge sp: steel plate bridge sub_v: v-type substructure sub_t: t-type substructure
	company	C_D: D E & C C_S: S Engineering C_Y: Y Engineering C_K: K E & C C_M: M Engineering

language use, the data was created as a database by using the Cubrid (TM) DBMS. Furthermore, the decision tree library of the Accord.NET framework was used for classification and learning of data. Accord.NET is an open source. NET machine learning framework written in C#. For the decision tree model generation algorithm, C4.5, which has lower classification error rate than the classification and regression tree (CART), was applied (misclassification rate: 0.27), and the number of leaves in the initially created decision tree was 95. The P-value of the created model was 0.0005, which was determined to be an appropriate model. The variance inflation of ω_{NE}, ω_{S}, and ω_{C} was 1.173, 1.125, and 1.089, respectively, and the multicollinearity condition was satisfied. Therefore, built tree model can be considered suitable for the regression model. Table 2 shows the 95 leaves obtained from the decision tree modeling process. This model was used as the initial model for the iterative decision tree generation.

3.4 Applications of the decision tree model-based XML schema matching techniques

As previously described, the decision tree-based heuristic solution for the weight factors of the XML Schema matching of a bridge SCD proposed in this study is as follows (Fig. 10): 1) The generated decision tree model is inserted into the database management system (DBMS); 2) the weight factors of the XML Schema matching are presented according to the process shown in Section 3.2 and Fig. 8 after receiving the input variables through the user; 3) the new matching accuracy calculated by using the presented matching weight factors is added to the training data to generate the updated decision tree model through the process shown in Fig. 9.

As shown in Fig. 11, the user-interface (UI) was implemented for selecting the XML Schema matching weight factors of the bridge SCD according to the procedure shown in Fig. 10. The initial database generation in process 1) is as follows: to extract and classify information on the number of input variables, variable names, and data types from the decision tree model data, as shown in Fig. 11(a), create dynamic UI elements based on the results and further, create a database through the user identification process. Figure 11(b) shows a module that proposes an XML Schema matching weight factor suitable for a corresponding engineering document according to the new conditions entered by the user, displaying the factors in a range that can be utilized for each weight factor.

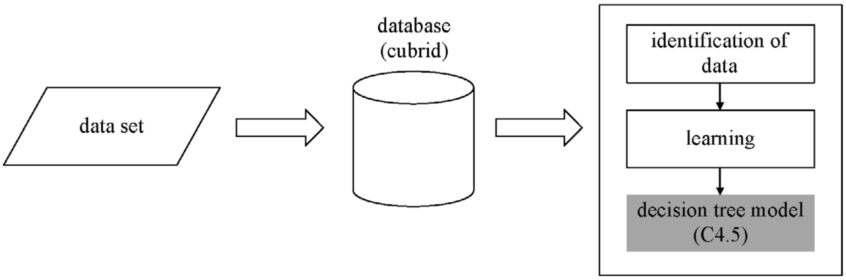


Fig. 9 Decision tree modeling process.

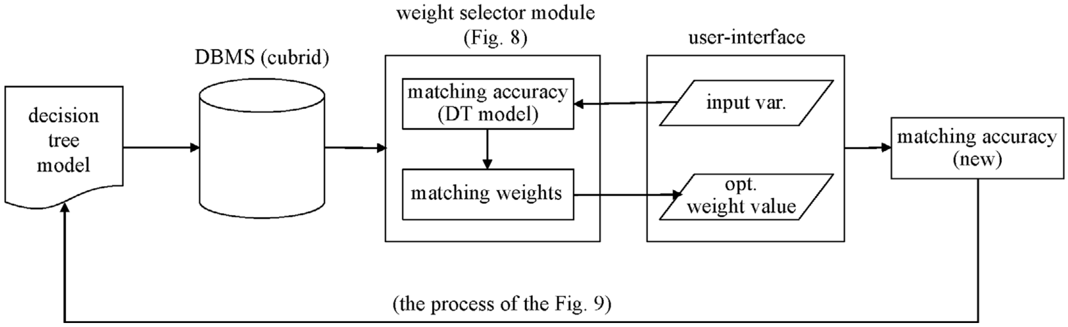


Fig. 10 XML Schema matching weight-factor selection process using the decision tree.

Table 2 The 95 leaves derived through the decision tree

item
A =: (COMPANY == C_M) && (NUM_LINE>1631.5) && (WNE>0.21111)
A =: (COMPANY == C_K) && (TYPE == sb) && (WNE < = 0.23611) && (WC>0.436505)
A =: (COMPANY == C_K) && (TYPE == sb) && (WNE>0.23611) && (WS>0.108825)
A =: (COMPANY == C_D) && (NUM_LINE>524) && (WNE>0.21111) && (TYPE == sub_v) && (WC < = 0.174245)
A =: (COMPANY == C_M) && (NUM_LINE>1631.5) && (WNE < = 0.21111) && (WS>0.207145) && (WC>0.19091)
B =: (COMPANY == C_K) && (TYPE == sub_v) && (WS < = 0.121325)
B =: (COMPANY == C_D) && (NUM_LINE < = 524) && (WC < = 0.13393) && (WNE>0.13393)
B =: (COMPANY == C_D) && (NUM_LINE>524) && (WNE>0.21111) && (TYPE == sp)
B =: (COMPANY == C_Y) && (NUM_LINE < = 1706) && (WC < = 0.267855) && (WNE < = 0.174245) && (TYPE == sub_t)
B =: (COMPANY == C_K) && (TYPE == sub_v) && (WS>0.121325) && (WNE>0.322915) && (WC>0.23611)
B =: (COMPANY == C_K) && (TYPE == sub_v) && (WS>0.121325) && (WNE < = 0.322915) && (WC>0.37647)
B =: (COMPANY == C_M) && (NUM_LINE < = 1631.5) && (TYPE == sub_t) && (WC < = 0.13393) && (WNE>0.13393)
B =: (COMPANY == C_M) && (NUM_LINE>1631.5) && (WNE < = 0.21111) && (WS>0.207145) && (WC < = 0.19091)
B =: (COMPANY == C_D) && (NUM_LINE < = 524) && (WC>0.13393) && (WNE>0.267855) && (WS>0.322915)
B =: (COMPANY == C_M) && (NUM_LINE < = 1631.5) && (TYPE == sub_t) && (WC>0.13393) && (WNE>0.174245) && (WS>0.267855)
B =: (COMPANY == C_D) && (NUM_LINE>524) && (WNE < = 0.21111) && (WC < = 0.436505) && (WS>0.39869) && (TYPE == sp)
B =: (COMPANY == C_D) && (NUM_LINE>524) && (WNE>0.21111) && (TYPE == sub_v) && (WC>0.174245) && (WS < = 0.23611)
B =: (COMPANY == C_D) && (NUM_LINE>524) && (WNE>0.21111) && (TYPE == cs) && (WS>0.23611) && (WC>0.23611)
B =: (COMPANY == C_D) && (NUM_LINE>524) && (WNE < = 0.21111) && (WC < = 0.436505) && (WS < = 0.39869) && (TYPE == cs)
B =: (COMPANY == C_S) && (NUM_LINE < = 1571.5) && (WNE < = 0.267855) && (WS < = 0.207145) && (TYPE == sub_t) && (WC>0.414285)
C =: (COMPANY == C_S) && (NUM_LINE < = 1571.5) && (WNE>0.267855)
C =: (COMPANY == C_D) && (NUM_LINE < = 524) && (WC>0.13393) && (WNE < = 0.267855)
C =: (COMPANY == C_K) && (TYPE == sb) && (WNE>0.23611) && (WS < = 0.108825)
C =: (COMPANY == C_M) && (NUM_LINE < = 1631.5) && (TYPE == sp) && (WC>0.21111)
C =: (COMPANY == C_Y) && (NUM_LINE>1706) && (WS>0.39869) && (TYPE == sb)
C =: (COMPANY == C_S) && (NUM_LINE>1571.5) && (TYPE == cs) && (WNE>0.267855)
C =: (COMPANY == C_D) && (NUM_LINE>524) && (WNE>0.21111) && (TYPE == cs) && (WS < = 0.23611)
C =: (COMPANY == C_K) && (TYPE == cs) && (WNE < = 0.39869) && (WS>0.322915) && (WC < = 0.174245)
C =: (COMPANY == C_K) && (TYPE == cs) && (WNE < = 0.39869) && (WS < = 0.322915) && (WC>0.414285)
C =: (COMPANY == C_M) && (NUM_LINE < = 1631.5) && (TYPE == sp) && (WC < = 0.21111) && (WNE>0.21111)
C =: (COMPANY == C_D) && (NUM_LINE>524) && (WNE < = 0.21111) && (WC>0.436505) && (TYPE == sp)
C =: (COMPANY == C_Y) && (NUM_LINE < = 1706) && (WC < = 0.267855) && (WNE < = 0.174245) && (TYPE == sub_v)
C =: (COMPANY == C_Y) && (NUM_LINE < = 1706) && (WC < = 0.267855) && (WNE < = 0.174245) && (TYPE == sp)

(*Continued*)

item
C =: (COMPANY == C_M) && (NUM_LINE>1631.5) && (WNE < = 0.21111) && (WS < = 0.207145) && (WC < = 0.21111)
C =: (COMPANY == C_M) && (NUM_LINE < = 1631.5) && (TYPE == sub_t) && (WC>0.13393) && (WNE < = 0.174245)
C =: (COMPANY == C_K) && (TYPE == sub_v) && (WS>0.121325) && (WNE>0.322915) && (WC < = 0.23611)
C =: (COMPANY == C_K) && (TYPE == sub_v) && (WS>0.121325) && (WNE < = 0.322915) && (WC < = 0.37647)
C =: (COMPANY == C_S) && (NUM_LINE>1571.5) && (TYPE == cs) && (WNE < = 0.267855) && (WS>0.39869)
C =: (COMPANY == C_K) && (TYPE == cs) && (WNE < = 0.39869) && (WS < = 0.322915) && (WC < = 0.414285)
C =: (COMPANY == C_D) && (NUM_LINE>524) && (WNE < = 0.21111) && (WC>0.436505) && (TYPE == sub_v)
C =: (COMPANY == C_S) && (NUM_LINE < = 1571.5) && (WNE < = 0.267855) && (WS>0.207145) && (WC>0.267855)
C =: (COMPANY == C_Y) && (NUM_LINE>1706) && (WS < = 0.39869) && (WNE>0.19091) && (WC>0.267855)
C =: (COMPANY == C_D) && (NUM_LINE < = 524) && (WC>0.13393) && (WNE>0.267855) && (WS < = 0.322915) && (TYPE == sub_t)
C =: (COMPANY == C_D) && (NUM_LINE>524) && (WNE>0.21111) && (TYPE == sub_v) && (WC>0.174245) && (WS>0.23611)
C =: (COMPANY == C_Y) && (NUM_LINE>1706) && (WS < = 0.39869) && (WNE < = 0.19091) && (TYPE == sb) && (WC>0.174245)
C =: (COMPANY == C_D) && (NUM_LINE>524) && (WNE>0.21111) && (TYPE == cs) && (WS>0.23611) && (WC < = 0.23611)
C =: (COMPANY == C_Y) && (NUM_LINE>1706) && (WS < = 0.39869) && (WNE < = 0.19091) && (TYPE == cs) && (WC < = 0.174245)
C =: (COMPANY == C_D) && (NUM_LINE>524) && (WNE < = 0.21111) && (WC < = 0.436505) && (WS < = 0.39869) && (TYPE == sp)
C =: (COMPANY == C_Y) && (NUM_LINE < = 1706) && (WC>0.267855) && (WS < = 0.322915) && (WNE>0.21111) && (TYPE == sub_t)
C =: (COMPANY == C_Y) && (NUM_LINE>1706) && (WS < = 0.39869) && (WNE>0.19091) && (WC < = 0.267855) && (TYPE == sb)
C =: (COMPANY == C_Y) && (NUM_LINE < = 1706) && (WC>0.267855) && (WS < = 0.322915) && (WNE < = 0.21111) && (TYPE == sp)
C =: (COMPANY == C_Y) && (NUM_LINE < = 1706) && (WC>0.267855) && (WS < = 0.322915) && (WNE < = 0.21111) && (TYPE == sub_v)
C =: (COMPANY == C_S) && (NUM_LINE < = 1571.5) && (WNE < = 0.267855) && (WS < = 0.207145) && (TYPE == sp) && (WC>0.21111)
C =: (COMPANY == C_M) && (NUM_LINE>1631.5) && (WNE < = 0.21111) && (WS < = 0.207145) && (WC>0.21111) && (TYPE == sb)
C =: (COMPANY == C_S) && (NUM_LINE < = 1571.5) && (WNE < = 0.267855) && (WS < = 0.207145) && (TYPE == sub_v) && (WC>0.21111)
C =: (COMPANY == C_S) && (NUM_LINE < = 1571.5) && (WNE < = 0.267855) && (WS < = 0.207145) && (TYPE == sub_t) && (WC < = 0.414285)
C =: (COMPANY == C_S) && (NUM_LINE < = 1571.5) && (WNE < = 0.267855) && (WS>0.207145) && (WC < = 0.267855) && (TYPE == sp)
C =: (COMPANY == C_S) && (NUM_LINE < = 1571.5) && (WNE < = 0.267855) && (WS>0.207145) && (WC < = 0.267855) && (TYPE == sub_v)
C =: (COMPANY == C_S) && (NUM_LINE < = 1571.5) && (WNE < = 0.267855) && (WS>0.207145) && (WC < = 0.267855) && (TYPE == sub_t)
C =: (COMPANY == C_M) && (NUM_LINE < = 1631.5) && (TYPE == sub_t) && (WC>0.13393) && (WNE>0.174245) && (WS < = 0.267855)
C =: (COMPANY == C_S) && (NUM_LINE>1571.5) && (TYPE == cs) && (WNE < = 0.267855) && (WS < = 0.39869) && (WC>0.436505)
C =: (COMPANY == C_Y) && (NUM_LINE < = 1706) && (WC>0.267855) && (WS < = 0.322915) && (WNE < = 0.21111) && (TYPE == sub_t)
C =: (COMPANY == C_D) && (NUM_LINE>524) && (WNE < = 0.21111) && (WC < = 0.436505) && (WS>0.39869) && (TYPE == sub_v)
C =: (COMPANY == C_Y) && (NUM_LINE < = 1706) && (WC < = 0.267855) && (WNE>0.174245) && (WS < = 0.267855) && (TYPE == sub_v)
C =: (COMPANY == C_Y) && (NUM_LINE < = 1706) && (WC < = 0.267855) && (WNE>0.174245) && (WS < = 0.267855) && (TYPE == sub_t)
C =: (COMPANY == C_M) && (NUM_LINE < = 1631.5) && (TYPE == sp) && (WC < = 0.21111) && (WNE < = 0.21111) && (WS>0.207145)
C =: (COMPANY == C_Y) && (NUM_LINE < = 1706) && (WC < = 0.267855) && (WNE>0.174245) && (WS>0.267855) && (TYPE == sub_t)
C =: (COMPANY == C_D) && (NUM_LINE>524) && (WNE < = 0.21111) && (WC < = 0.436505) && (WS>0.39869) && (TYPE == cs)
D =: (COMPANY == C_K) && (TYPE == cs) && (WNE>0.39869)
D =: (COMPANY == C_K) && (TYPE == cs) && (WNE < = 0.39869) && (WS>0.322915) && (WC>0.174245)
D =: (COMPANY == C_Y) && (NUM_LINE < = 1706) && (WC>0.267855) && (WS>0.322915) && (TYPE == sp)
D =: (COMPANY == C_D) && (NUM_LINE>524) && (WNE < = 0.21111) && (WC>0.436505) && (TYPE == cs)
D =: (COMPANY == C_M) && (NUM_LINE < = 1631.5) && (TYPE == sub_t) && (WC < = 0.13393) && (WNE < = 0.13393)
D =: (COMPANY == C_S) && (NUM_LINE>1571.5) && (TYPE == sb) && (WNE < = 0.39869) && (WS < = 0.39869)
D =: (COMPANY == C_M) && (NUM_LINE < = 1631.5) && (TYPE == sp) && (WC < = 0.21111) && (WNE < = 0.21111) && (WS < = 0.207145)
D =: (COMPANY == C_S) && (NUM_LINE < = 1571.5) && (WNE < = 0.267855) && (WS < = 0.207145) && (TYPE == sp) && (WC < = 0.21111)
D =: (COMPANY == C_D) && (NUM_LINE>524) && (WNE < = 0.21111) && (WC < = 0.436505) && (WS < = 0.39869) && (TYPE == sub_v)

(*Continued*)

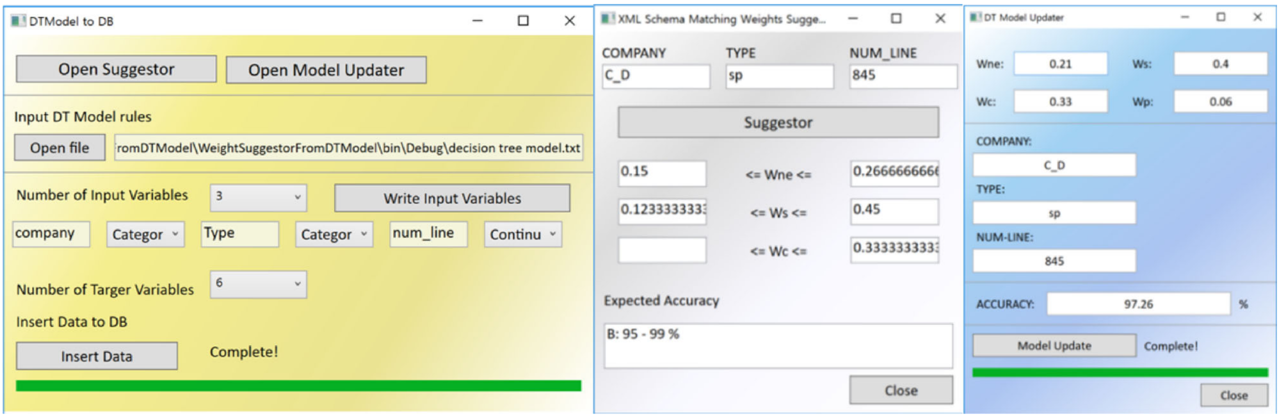
item
D =: (COMPANY == C_Y) && (NUM_LINE < = 1706) && (WC>0.267855) && (WS < = 0.322915) && (WNE>0.21111) && (TYPE == sp)
D =: (COMPANY == C_S) && (NUM_LINE < = 1571.5) && (WNE < = 0.267855) && (WS < = 0.207145) && (TYPE == sub_v) && (WC < = 0.21111)
D =: (COMPANY == C_Y) && (NUM_LINE < = 1706) && (WC>0.267855) && (WS < = 0.322915) && (WNE>0.21111) && (TYPE == sub_v)
D =: (COMPANY == C_S) && (NUM_LINE>1571.5) && (TYPE == cs) && (WNE < = 0.267855) && (WS < = 0.39869) && (WC < = 0.436505)
D =: (COMPANY == C_Y) && (NUM_LINE>1706) && (WS < = 0.39869) && (WNE>0.19091) && (WC < = 0.267855) && (TYPE == cs)
D =: (COMPANY == C_Y) && (NUM_LINE>1706) && (WS < = 0.39869) && (WNE < = 0.19091) && (TYPE == sb) && (WC < = 0.174245)
D =: (COMPANY == C_Y) && (NUM_LINE>1706) && (WS < = 0.39869) && (WNE < = 0.19091) && (TYPE == cs) && (WC>0.174245)
E =: (COMPANY == C_S) && (NUM_LINE>1571.5) && (TYPE == sb) && (WNE>0.39869)
E =: (COMPANY == C_Y) && (NUM_LINE>1706) && (WS>0.39869) && (TYPE == cs)
E =: (COMPANY == C_D) && (NUM_LINE < = 524) && (WC < = 0.13393) && (WNE < = 0.13393)
E =: (COMPANY == C_Y) && (NUM_LINE < = 1706) && (WC>0.267855) && (WS>0.322915) && (TYPE == sub_v)
E =: (COMPANY == C_S) && (NUM_LINE>1571.5) && (TYPE == sb) && (WNE < = 0.39869) && (WS>0.39869)
E =: (COMPANY == C_Y) && (NUM_LINE < = 1706) && (WC>0.267855) && (WS>0.322915) && (TYPE == sub_t)
E =: (COMPANY == C_Y) && (NUM_LINE < = 1706) && (WC < = 0.267855) && (WNE>0.174245) && (WS>0.267855) && (TYPE == sub_v)
E =: (COMPANY == C_Y) && (NUM_LINE < = 1706) && (WC < = 0.267855) && (WNE>0.174245) && (WS < = 0.267855) && (TYPE == sp)
F =: (COMPANY == C_K) && (TYPE == sb) && (WNE < = 0.23611) && (WC < = 0.436505) && (WS>0.218255)
F =: (COMPANY == C_K) && (TYPE == sb) && (WNE < = 0.23611) && (WC < = 0.436505) && (WS < = 0.218255)
F =: (COMPANY == C_Y) && (NUM_LINE < = 1706) && (WC < = 0.267855) && (WNE>0.174245) && (WS>0.267855) && (TYPE == sp)

Figure 11(c) shows how to update the decision tree model by utilizing the calculated matching accuracy criterion after performing XML Schema matching using the weight factors. The entire module was implemented through Visual C# in the NET Framework environment. The DBMS for managing variables, accuracy, and weight factor data uses the same Cubrid (TM) as in the decision-tree model generation.

To verify the applicability, this study repeatedly updated the decision tree model for 26 SCDs. In this case, a total of 121 leaves were generated, and it was confirmed that the condition and range of the weight were specified as the model update proceeded. Table 3 shows some examples for this verification. The experiment results on bridge SCDs showed an accuracy improvement of 9.4% or more on average.

4 Discussion and conclusions

Although the SCD generated in the design phase is the engineering document that has the most significant



DTModel to DB

Input DT Model rules

romDTModel\WeightSuggestorFromDTModel\bin\Debug\decision tree model.txt

Number of Input Variables: 3

company: Categor

Type: Categor

num_line: Continu

Number of Targer Variables: 6

Insert Data to DB

Complete!

XML Schema Matching Weights Sugge...

COMPANY: C_D

TYPE: sp

NUM_LINE: 845

0.15 <= Wne <= 0.2666666666

0.123333333 <= Ws <= 0.45

<= Wc <= 0.333333333

Expected Accuracy

B: 95 - 99 %

DT Model Updater

Wne: 0.21

Ws: 0.4

Wc: 0.33

Wp: 0.06

COMPANY: C_D

TYPE: sp

NUM-LINE: 845

ACCURACY: 97.26 %

Complete!

(a) (b) (c)

Fig. 11 Automatic scheme matching weight-factor selection modules using the DT model DB: (a) database building module based on decision tree model; (b) suggest module for suitable weight factors of XML Schema matching; (c) update module of the decision tree.

Table 3 Examples of the DT model-based SMM accuracy

input variable			MM module	SMM module	
type	company	No. of elements	accuracy (%)	used weight value	accuracy (%)
cable-stayed bridge	S engineering	1028	85.22	$\omega_{\mathrm{NE}} = 0.26$, $\omega_{\mathrm{S}} = 0.21$, $\omega_{\mathrm{C}} = 0.27$, $\omega_{\mathrm{P}} = 0.26$	95.08
steel plate bridge	D engineering	845	90.91	$\omega_{\mathrm{NE}} = 0.21$, $\omega_{\mathrm{S}} = 0.40$, $\omega_{\mathrm{C}} = 0.33$, $\omega_{\mathrm{P}} = 0.06$	97.26
v-type substructure	K engineering	549	87.50	$\omega_{\mathrm{NE}} = 0.32$, $\omega_{\mathrm{S}} = 0.13$, $\omega_{\mathrm{C}} = 0.38$, $\omega_{\mathrm{P}} = 0.17$	96.71
steel box girder bridge	Y engineering	1826	78.13	$\omega_{\mathrm{NE}} = 0.19$, $\omega_{\mathrm{S}} = 0.39$, $\omega_{\mathrm{C}} = 0.18$, $\omega_{\mathrm{P}} = 0.24$	94.58
cable-stayed bridge	M engineering	1933	93.33	$\omega_{\mathrm{NE}} = 0.21$, $\omega_{\mathrm{S}} = 0.21$, $\omega_{\mathrm{C}} = 0.20$, $\omega_{\mathrm{P}} = 0.38$	98.65

influence on the overall lifecycle of the structure, the quality of the internal data stored this document depends heavily on the document creator. Moreover, an automatic review of the content of these documents once they have been generated is rarely performed. This study introduced XML Schema matching as a method to improve the quality of document data of engineering documents, particularly those of bridge SCDs. However, it was hard to apply the Refs. [21–25]. methods because the schemas generated using SCDs do not include data types or constraints for each element. Furthermore, it had limitations in implementing the research of instance-based schema matching Ref. [28] according to SCDs's non-instantiated features. Thus, among the various methods for XML schema matching, this study selected the method proposed in Ref. [18]. This method does not require considering the data type of the elements included in the schema, and further, can be applied to the selected method to bridge SCDs. The study of Ref. [18], however, focused on the development of schema matching technique itself, and it experimented the developed method on only prototype schemas having a small number of elements. Accordingly, this method consumed an extreme amount of time for matching when applied to engineering documents with a deep structure and numerous elements. This resulted from consideration of the correlations and constraints of each element contained in the source and target schema. Unlike the general schemas, the schemas of SCDs do not have constraints on their elements, which can be key in reducing the correction of element relationships. Accordingly, the authors proposed a simplified XML schema matching method for reducing the computation time (see Fig. 5). The validity of the simplified XML schema matching process was verified by evaluating the accuracy when applied to practical bridge SCDs. During this process, this study confirmed that the influence on accuracy could be corrected by changing the weight factors between the elements used in XML schema matching. The result of schema matching according to the change in weight factors was confirmed through a parametric study applying the concept of equality constraint method. The results shown in Figs. 3–4 confirm that the simplified XML schema matching is a valid model that could reduce the computation time dramatically while maintaining the matching accuracies compared with the method proposed in Ref. [18]. However, this process is meaningful only when the accuracy of schema matching is maintained. The parametric study to ensure the accuracy is a process further requiring a long time of manual operation.

Data mining is a process of finding a new pattern based on the collected data. We concluded that data mining could replace parametric studies for selecting suitable weight factors owing to the accumulation of schema matching results for SCDs. Thus, this study constructed and utilized the decision tree model, one of the classifications for data mining, by using the accuracy of XML schema matching as the target value, along with the structural type of the bridge, SCD manufacturer, number of elements constituting the SCD, and corresponding matching weight factors as input values. The decision tree model was generated using a ratio of training to validation data of 6:4 with 580 data samples calculated in the previous parametric study. The P-value of the decision tree was 0.0005, which was determined to be an appropriate model. The variance inflation of the weight factors was 1.173, 1.125, and 1.089, respectively, and the multicollinearity condition was satisfied. Therefore, the built decision tree model can be considered valid for the regression model, and the inverse-calculation method proposed in this study can be applied to the selection of weight factors for simplified XML schema matching. In addition, the authors confirmed that the decision tree model is updated more precisely as the matching result is added.

Studies on reviewing the quality of non-geometric information are relatively neglected while BIM, which aims to improve productivity by integrating information technology in the construction field, has been actively promoted to generate and manage information based on open standards. The automatic review of the data quality has rarely been studied, although SCDs stores vast and deep-hierarchical contents, and the authors had difficulty finding suitable methodologies. In this study, the authors proposed a heuristic solution for reviewing the data quality of SCD contents by introducing and integrating various methods such as parametric studies based on the equality constraint method, decision tree model, and inverse-calculation method on weight factor applying XML schema matching. The limitation of this solution is the need to create a new decision tree model when the schema matching technique evolves or the format of SCD contents is significantly changed. Therefore, advanced studies should be continued to develop a schema matching technique that accurately reflects the features of bridge SCDs.

Acknowledgements This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2016R1A6A3A11934917).

References

1. Liu S, McMahon C A, Darlington M J, Culley S J, Wild P J. A computational framework for retrieval of document fragments based on decomposition schemes in engineering information management. Advanced Engineering Informatics, 2006, 20(4): 401–413
2. Tan X, Hammad A, Fazio P. Automated code compliance checking for building envelope design. Journal of Computing in Civil Engineering, 2010, 24(2): 203–211
3. Zhong B T, Ding L Y, Luo H B, Zhou Y, Hu Y Z, Hu H M. Ontology-based semantic modeling of regulation constraint for automated construction quality compliance checking. Automation in Construction, 2012, 28: 58–70
4. Zhang J, El-Gohary N M. Semantic NLP-based information extraction from construction regulatory documents for automated compliance checking. Journal of Computing in Civil Engineering, 2016, 30(2): 04015014
5. Lin K Y, Soibelman L. Incorporating domain knowledge and information retrieval techniques to develop an architectural/engineering/construction online product search engine. Journal of Computing in Civil Engineering, 2009, 23(4): 201–210
6. McGibbney L J, Kumar B. A knowledge-directed information retrieval and management framework for energy performance building regulations. In: Proceedings from International Workshop on Computing in Civil Engineering 2011. Miami, FL: American Society of Civil Engineers, 2011, 339–346
7. Zhang L, El-Gohary N M. Epistemology-based context-aware semantic model for sustainable construction practices. Journal of Construction Engineering and Management, 2016, 142(3): 04015084
8. Zhou P, El-Gohary N M. Automated matching of design information in BIM to regulatory information in energy codes. In: Proceedings from Construction Research Congress 2018. New Orleans, LA: American Society of Civil Engineers, 2018, 75–85
9. Sacks R, Bloch T, Katz M, Yosef R. Automating design review with artificial intelligence and BIM: State of the art and research framework. In: Proceedings from Computing in Civil Engineering 2019: Visualization, Information Modeling, and Simulation. Atlanta, GA: American Society of Civil Engineers, 2019, 353–360
10. Caldas C H, Soibelman L. Automating hierarchical document classification for construction management information systems. Automation in Construction, 2003, 12(4): 395–406
11. Ma Z, Li H, Shen Q P, Yang J. Using XML to support information exchange in construction projects. Automation in Construction, 2004, 13(5): 629–637
12. Park S I, Kim B G, Kim K H, Lee S H. A methodology for automatic hierarchy definition of sentences in engineering documents. Journal of Computational Structural Engineering Institute of Korea, 2009, 22: 323–330 (in Korean)
13. Kim B G, Park S I, Kim H J, Lee S H. Automatic extraction of apparent semantic structure from text contents of a structural calculation document. Journal of Computing in Civil Engineering, 2010, 24(3): 313–324
14. Lee S H, Kim B G, Kim H J, Kim S J. A strategy for IT-based lifetime management of bridge. In: Proceedings from Bridge Maintenance, Safety, Management, Health Monitoring and Informatics (IABMAS08). Seoul: CRC Press, 2008
15. Kim B G. Integration of a 3-D Bridge model and structured information of engineering documents. Dissertation for the Doctoral Degree. Seoul: Yonsei University, 2010
16. Rahm E, Bernstein P A. A survey of approaches to automatic schema matching. VLDB Journal, 2001, 10(4): 334–350
17. Lee S H, Kim B G, Kim D H, Jeong Y S. Development of standardized semantic model for structural calculation documents of bridges and XML schema matching technique. In: Proceedings from the 3rd International Conference on Bridge Maintenance Safety and Management (IABMAS). Porto: Taylor & Francis, 2006
18. Yi S, Huang B, Tatchan W. XML application schema matching using similarity measure and relaxation labeling. Information Sciences, 2005, 169(1-2): 27–46
19. Park S I, Kim B G, Lee S H. An efficient application of XML schema matching technique to structural calculation document of bridge. Journal of the Korean Society of Civil Engineers, 2012, 32: 51–59 (in Korean)
20. Lin J G. Multiple-objective problems: Pareto-optimal solutions by method of proper equality constraints. IEEE Transactions on Automatic Control, 1976, 21(5): 641–650
21. Li W S, Clifton C. SEMINT: A tool for identifying attribute correspondences in heterogeneous databases using neural networks. Data & Knowledge Engineering, 2000, 33(1): 49–84
22. Madhavan J, Bernstein P A, Rahm E. Generic schema matching with cupid. In: Proceedings of the 27th International Conference on Very Large Data Bases. San Francisco, CA: Morgan Kaufmann Publishers Inc., 2001, 49–58
23. Castano S, De Antonellis V. Global viewing of heterogeneous data sources. IEEE Transactions on Knowledge and Data Engineering, 2001, 13(2): 277–297
24. Algergawy A, Schallehn E, Saake G. Improving XML schema matching performance using Prüfer sequences. Data & Knowledge Engineering, 2009, 68(8): 728–747
25. Algergawy A, Massmann S, Rahm E. A clustering-based approach for large-scale ontology matching. In: Proceedings from ADBIS 2011. Berlin: Heidelberg, 2011, 415–428
26. Melnik S, Garcia-Molina H, Rahm E. Similarity flooding: A versatile graph matching algorithm and its application to schema matching. In: Proceedings from 18th International Conference on Data Engineering. San Jose, CA: IEEE Computer Society, 2002,

117–128

27. Doan A, Madhavan J, Domingos P, Halevy A. Learning to map between ontologies on the semantic web. In: Proceedings of the 11th International Conference on World Wide Web. Honolulu, HI: Association for Computing Machinery, 2002, 662–673
28. Doan A, Domingos P, Halevy A Y. Reconciling schemas of disparate data sources: A machine-learning approach. ACM SIGMOD Record Journal of Management in Engineering, 2001, 30(2): 509–520
29. Fayyad U, Piatetsky-Shapiro G, Smyth P. From data mining to knowledge discovery in databases. AI Magazine, 1996, 17(3): 37–54
30. Adriaans P, Zantinge D. Data Mining. Boston: Addison-Wesley, 1996