Selecting a semantic similarity measure for concepts in two different CAD model data ontologies

Wenlong Lu [a], Yuchu Qin [a,*], Qunfen Qi [b], Wenhan Zeng [b], Yanru Zhong [c], Xiaojun Liu [a], Xiangqian Jiang [b]

[a] *The State Key Laboratory of Digital Manufacturing Equipment and Technology, School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, PR China*

[b] *EPSRC Centre for Innovative Manufacturing in Advanced Metrology, University of Huddersfield, Huddersfield, HD1 3DH, UK*

[c] *School of Computer Science and Information Security, Guilin University of Electronic Technology, Guilin 541004, PR China*

Abstract: Semantic similarity measure technology based approach is one of the most popular approaches aiming at implementing semantic mapping between two different CAD model data ontologies. The most important problem in this approach is how to measure the semantic similarities of concepts between two different ontologies. A number of measure methods focusing on this problem have been presented in recent years. Each method can work well between its specific ontologies. But it is unclear how accurate the measured semantic similarities in these methods are. Moreover, there is yet no evidence that any of the methods presented how to select a measure with high similarity calculation accuracy. To compensate for such deficiencies, this paper proposes a method for selecting a semantic similarity measure with high similarity calculation accuracy for concepts in two different CAD model data ontologies. In this method, the similarity calculation accuracy of each candidate measure is quantified using Pearson correlation coefficient or residual sum of squares. The measure with high similarity calculation accuracy is selected through a comparison of the Pearson correlation coefficients or the residual sums of squares of all candidate measures. The paper also reports an implementation of the proposed method, provides an example to show how the method works, and evaluates the method by theoretical and experimental comparisons. The evaluation result suggests that the measure selected by the proposed method has good human correlation and high similarity calculation accuracy.

Keywords: Similarity measure selection; Semantic similarity measure; Similarity calculation accuracy; CAD model data ontology; Concept; Weight

1. Introduction

The division of labor among enterprises is becoming more and more refined with the deepening trend of manufacturing industry globalization. The development work of complex products (e.g. automobiles, ships, and planes) is collaboratively finished by multiple enterprises from different regions and even from different countries in most cases. For the design work in it alone, different enterprises are usually responsible for designing different parts or components of a complex product. The CAD systems used in these designs are also always different. To use one CAD system to pre-assemble the designed parts or components to perform engineering analysis on the product, the model data of the parts or components stored in other CAD systems must be completely transferred to this CAD system. However, since the design of the data structure, modeling manipulation, and data storage method of different CAD systems are always different, the model data is difficult

* Corresponding author.
E-mail address: qinyuchu@hust.edu.cn (Y. Qin).

to be directly exchanged among these heterogeneous CAD systems [1].

To implement the exchange of the CAD model data among heterogeneous CAD systems, the industry mainly uses the standard for the exchange of product model data (STEP) [2] neutral files based approach. The data modeling language used in these files is EXPRESS [3]. Even though EXPRESS can construct syntactically correct product data model, it cannot express and interpret the semantics assigned to the model explicitly [4]. For this reason, STEP neutral files are only capable of exchanging the syntaxes of the CAD model data and do not enable the exchange of the semantics of these data. The semantic interoperability of CAD model data among heterogeneous CAD systems is difficult to be truly implemented only by STEP neutral files based approach, which leads to a serious problem that all the data related to high-level design intent, such as design history, parameters, constraints, and features, are completely lost after the exchange [5].

In response to the CAD model data loss problem caused by the lack of explicit semantics in STEP neutral files, serval kinds of approaches have been proposed during the past decade. Among these kinds of approaches, Semantic Web technologies based approaches may be the most dominant kind [6]. This kind of approaches tries to use the technologies in the field of the Semantic Web [7] to implement the semantic interoperability of CAD model data among heterogeneous CAD systems. These used technologies mainly include rule reasoning technology and hybrid technologies which combine both rule reasoning and semantic similarity measure technologies. According to these used technologies, Semantic Web technologies based approaches are further classified into rule approaches [8-15] and hybrid approaches [16-19]. Rule approaches use the reasoning mechanism of web ontology language (OWL) [20] and semantic web rule language (SWRL) [21] to determine whether each two concepts that are respectively from two different CAD model data ontologies are semantically equivalent. If two concepts are semantically equivalent, all the individuals of one concept will be created as the individuals of the other concept. As an example, assume *PROE-Extrude* is a concept in Pro/Engineer (PROE) model data ontology, *UGNX-Extrude* is a concept in Unigraphics NX (UGNX) model data ontology, and these two concepts have the following OWL descriptions [9]:

$PROE\text{-}Extrude \equiv PROE\text{-}Feature \sqcap \exists proe\text{-}hasParent.PROE\text{-}Sketch$

$UGNX\text{-}Extrude \equiv UGNX\text{-}Feature \sqcap \exists ugnx\text{-}hasParent.UGNX\text{-}Sketch$

Assume further that EXTRUDE1 is an extrusion feature in PROE that has SKETCH1 as its parent and the following OWL axioms have been manually defined in a combination of the UGNX and PROE model data ontologies:

$PROE\text{-}Feature \equiv UGNX\text{-}Feature$, $PROE\text{-}Sketch \equiv UGNX\text{-}Sketch$, $proe\text{-}hasParent \equiv ugnx\text{-}hasParent$

Using the reasoning mechanism of OWL, it can be automatically inferred that the concepts *PROE-Extrude* and *UGNX-Extrude* are semantically equivalent. Therefore, EXTRUDE1 (initially, EXTRUDE1 is an individual of *PROE-Extrude*) is created as an individual of *UGNX-Extrude*. The process of transferring the extrusion feature from PROE to UGNX can be described by the following two groups of OWL assertions:

PROE: *PROE-Extrude*(EXTRUDE1), *PROE-Sketch*(SKETCH1), *proe-hasParent*(EXTRUDE1, SKETCH1)

UGNX: *UGNX-Extrude*(EXTRUDE1), *UGNX-Sketch*(SKETCH1), *ugnx-hasParent*(EXTRUDE1, SKETCH1)

As can be reflected from the above example, a major advantage of rule approaches is that the semantics of CAD model data can be explicitly represented in them (e.g. “a PROE extrude is a PROE feature that has a PROE sketch as its parent” is explicitly represented as “$PROE\text{-}Extrude \equiv PROE\text{-}Feature \sqcap \exists proe\text{-}hasParent$.

PROE-Sketch"), which makes it possible to automatically exchange such semantics. However, the approaches cannot be used to implement the individual data exchange between two concepts that are not exactly equivalent on semantics [16]. To overcome this limitation, rule approaches were extended through introducing semantic similarity measure technologies. These extended approaches attempt to use the assessment result of the semantic similarity between each two concepts which are not exactly equivalent on semantics to determine the mapping concept pairs. For example, assume *PROE-RectangleProfileHole* is a concept in PROE model data ontology, *UGNX-GeneralHole* is a concept in UGNX model data ontology, and these two concepts have the following OWL descriptions:

$$\begin{aligned}
\textit{PROE-RectangleProfileHole} \sqsubseteq\ & \textit{PROE-Hole} \sqcap {=}1\textit{proe-hasName} \sqcap \exists \textit{proe-hasName}.\text{string} \\
& \sqcap {=}1\textit{proe-hasPlacement} \sqcap \exists \textit{proe-hasPlacement}.(\textit{PROE-Point} \sqcup \textit{PROE-Axis} \sqcup \textit{PROE-Surface} \sqcup \textit{PROE-DatumPlane}) \\
& \sqcap {=}1\textit{proe-hasPlacementType} \sqcap \exists \textit{proe-hasPlacementType}.(\textit{PROE-Linear} \sqcup \textit{PROE-Radial} \sqcup \textit{PROE-Diameter}) \\
& \sqcap {=}1\textit{proe-hasDiameter} \sqcap \exists \textit{proe-hasDiameter}.\text{float} \\
& \sqcap {=}1\textit{proe-hasSideDepth} \sqcap \exists \textit{proe-hasSideDepth}.(\textit{PROE-Blind} \sqcup \textit{PROE-Symmetric} \sqcup \textit{PROE-ToNext} \sqcup \textit{PROE-ThroughAll} \sqcup \textit{PROE-ThroughUntil} \sqcup \textit{PROE-ToSelected}) \\
& \sqcap \leq 1\textit{proe-hasLightweight} \sqcap \exists \textit{proe-hasLightweight}.\textit{PROE-Lightweight} \\
& \sqcap {=}1\textit{proe-hasTolerance} \sqcap \exists \textit{proe-hasTolerance}.\text{float}
\end{aligned}$$

$$\begin{aligned}
\textit{UGNX-GeneralHole} \sqsubseteq\ & \textit{UGNX-Hole} \sqcap {=}1\textit{ugnx-hasName} \sqcap \exists \textit{ugnx-hasName}.\text{string} \\
& \sqcap {=}1\textit{ugnx-hasPosition} \sqcap \exists \textit{ugnx-hasPosition}.(\textit{UGNX-SketchSection} \sqcup \textit{UGNX-Point}) \\
& \sqcap {=}1\textit{ugnx-hasHoleDirection} \sqcap \exists \textit{ugnx-hasHoleDirection}.(\textit{UGNX-Normal2Face} \sqcup \textit{UGNX-AlongVector}) \\
& \sqcap {=}1\textit{ugnx-hasForm} \sqcap \exists \textit{ugnx-hasForm}.(\textit{UGNX-Simple} \sqcup \textit{UGNX-Counterbored} \sqcup \textit{UGNX-Countersunk} \sqcup \textit{UGNX-Tapered}) \\
& \sqcap {=}1\textit{ugnx-hasDiameter} \sqcap \exists \textit{ugnx-hasDiameter}.(\text{float} \sqcup \textit{UGNX-Measure} \sqcup \textit{UGNX-Formula} \sqcup \textit{UGNX-Function} \sqcup \textit{UGNX-Reference} \sqcup \textit{UGNX-Constant}) \\
& \sqcap {=}1\textit{ugnx-hasDepthLimit} \sqcap \exists \textit{ugnx-hasDepthLimit}.(\textit{UGNX-Value} \sqcup \textit{UGNX-UntilSelected} \sqcup \textit{UGNX-UntilNext} \sqcup \textit{UGNX-ThroughBody}) \\
& \sqcap \leq 1\textit{ugnx-hasBoolean} \sqcap \exists \textit{ugnx-hasBoolean}.\textit{UGNX-Subtract} \\
& \sqcap {=}1\textit{ugnx-hasTolerance} \sqcap \exists \textit{ugnx-hasTolerance}.\text{float}
\end{aligned}$$

Using the reasoning mechanism of OWL and SWRL, one cannot infer that the two concepts are semantically equivalent. However, one can find that their OWL descriptions have many similarities. If such similarities can be measured, it is possible to conclude that the two concepts are mapped or not mapped. In such a concluding process, the most critical problem is how to measure such similarities.

Focusing on this problem, many ontology-based measure methods have been proposed during the past two decades [22]. Based on the way in which ontologies are analyzed to estimate semantic similarities, these methods can be classified into edge counting, information content and attribute-based methods. Edge counting and information content methods are used to measure the semantic similarities of concepts in the same ontology. They cannot be directly used to estimate the semantic similarities between concepts in two different

ontologies. Differently from these two methods, attribute-based method can not only be applied to assess the semantic similarities of concepts in the same ontology, but also be applied to assess the semantic similarities between concepts in two different ontologies [22]. Since semantic interoperability of CAD model data needs the semantic similarities of concepts in two different CAD model data ontologies, the semantic similarity measures in edge counting and information content methods cannot be directly used and the measures in attribute-based method can be directly used when implementing semantic interoperability of CAD model data.

The commonly used semantic similarity measures in attribute-based method are Tversky's measure [23], Petrakis et al.'s measure [24], and Sánchez et al.'s measure [25]. Patil [16], Lee et al. [17], Zhan et al. [18], and Abdul-Ghafour et al. [19] have respectively proposed four methods to use some of these measures to assess the semantic similarities of concepts in different CAD model data ontologies. Each proposed method is capable of working well between its specific CAD model data ontologies. But it is not clear how accurate the assessed semantic similarities in these four methods are. Moreover, there is yet no evidence that any of these methods presents how to select a measure with high similarity calculation accuracy for concepts in two different CAD model data ontologies.

In this paper, a method for selecting a semantic similarity measure with high similarity calculation accuracy for concepts in two different CAD model data ontologies is proposed. This method is derived from Abdul-Ghafour et al.'s method for semantic interoperability of CAD model data [19], which presented that the semantic similarity of two concepts can be aggregated as a weighted sum of the similarity of their semantic descriptions and the similarity of their semantic relationships and left two questions: (1) How to obtain the weights? (2) How to select two measures to respectively calculate the similarity of semantic descriptions and the similarity of semantic relationships? The method solves these two questions by designing a weight calculation algorithm and presenting a measure selection algorithm. The measure selection algorithm firstly uses the weight calculation algorithm to calculate out the weights of the similarities of semantic descriptions and semantic relationships and then respectively selects two measures that make the aggregated measure have high similarity calculation accuracy for the similarities of semantic descriptions and semantic relationships. To the best of knowledge, this is the first consideration of the similarity calculation accuracy of the measures for concepts in CAD model data ontologies.

The remainder of the paper is organized as follows. An overview of related work is provided in Section 2. The details of the proposed measure selection method are explained in Section 3. Section 4 reports a prototype implementation of the method, presents an example to show how the method works, and evaluates the method through theoretical and experimental comparisons. Section 5 ends the paper with a conclusion.

2. Related work

2.1. Measures in attribute-based method

The semantic similarity measures in attribute-based method are rooted into Tversky's contrast model of similarity [23], which derives from the set theory and subtracts the non-common attributes of the compared terms from the common attributes of these terms. Actually, common attributes tend to increase the semantic similarity and non-common attributes tend to decrease it. Formally, let $S(A_{T1})$ and $S(A_{T2})$ respectively be the sets of the attributes of terms T_1 and T_2 and $S(A_{T1})\backslash S(A_{T2})$ be the set of attributes in $S(A_{T1})$ but not in $S(A_{T2})$

(the reverse for $S(A_{T2})\backslash S(A_{T1})$). Then the similarity of T_1 and T_2 is defined to be a function of $S(A_{T1})\cap S(A_{T2})$, $S(A_{T1})\backslash S(A_{T2})$, and $S(A_{T2})\backslash S(A_{T1})$:

$$Sim_{\text{Tversky\_C}}(T_1,T_2)=\alpha f(S(A_{T1})\cap S(A_{T2}))-\beta f(S(A_{T1})\setminus S(A_{T2}))-\gamma f(S(A_{T2})\setminus S(A_{T1})) \quad (1)$$

where f is a function reflecting the salience of a set of attributes and α, β, and γ ($\alpha, \beta, \gamma \geq 0$) are the weights of $f(S(A_{T1})\cap S(A_{T2}))$, $f(S(A_{T1})\backslash S(A_{T2}))$, and $f(S(A_{T2})\backslash S(A_{T1}))$, respectively. It can be proved that $Sim_{\text{Tversky\_C}}(T_1, T_2)$ is not a normalized similarity measure since not all of its values lie between 0 and 1. This measure was normalized by Tversky and a ratio model was proposed [**23**]:

$$Sim_{\text{Tversky\_R}}(T_1,T_2)=\frac{f(S(A_{T1})\cap S(A_{T2}))}{f(S(A_{T1})\cap S(A_{T2}))+\varphi f(S(A_{T1})\setminus S(A_{T2}))+\psi f(S(A_{T2})\setminus S(A_{T1}))} \quad (2)$$

where φ and ψ ($\varphi, \psi \geq 0$) are the weights of $f(S(A_{T1})\backslash S(A_{T2}))$ and $f(S(A_{T2})\backslash S(A_{T1}))$, respectively.

The definition of the sets of attributes (i.e. $S(A_{T1})$ and $S(A_{T2})$) is crucial in the ratio model. In existing ratio model based measures, attributes always include synonym sets (synsets), definitions (meanings), and contexts that are available in ontologies.

Taking the synonym sets, distinguishing features, and semantic neighborhoods of concepts as attributes, the measure of Rodr ́guez and Egenhofer [**26**] is defined as a weighted sum of the semantic similarity of these attributes:

$$Sim_{\text{Rodriguez}}(C_1,C_2)=uSim_{\text{synsets}}(C_1,C_2)+vSim_{\text{features}}(C_1,C_2)+wSim_{\text{neighborhoods}}(C_1,C_2) \quad (3)$$

where u, v, and w respectively weight the contributions of the components $Sim_{\text{synsets}}(C_1, C_2)$, $Sim_{\text{features}}(C_1, C_2)$, and $Sim_{\text{neighborhoods}}(C_1, C_2)$ which are the semantic similarities of the synonym sets of concepts C_1 and C_2, the distinguishing features of C_1 and C_2, and the semantic neighborhoods of C_1 and C_2, respectively. These three semantic similarities can all be computed according to the following variant of the ratio model (by assigning $\varphi = \theta(C_1, C_2)$ and $\psi = 1 - \theta(C_1, C_2)$ in $Sim_{\text{Tversky\_R}}(T_1, T_2)$):

$$Sim(C_1,C_2)=\frac{|S(A_{C1})\cap S(A_{C2})|}{|S(A_{C1})\cap S(A_{C2})|+\theta(C_1,C_2)|S(A_{C1})\setminus S(A_{C2})|+[1-\theta(C_1,C_2)]|S(A_{C2})\setminus S(A_{C1})|} \quad (4)$$

where $S(A_{C1})$ and $S(A_{C2})$ are respectively the sets of the attributes of concepts C_1 and C_2, $S(A_{C1})\backslash S(A_{C2})$ is the set of attributes in $S(A_{C1})$ but not in $S(A_{C2})$ (the reverse for $S(A_{C2})\backslash S(A_{C1})$), and $\theta(C_1, C_2)$ can be calculated as a function of the depth of C_1 and C_2 in the graph representations of their respective ontologies:

$$\theta(C_1,C_2)=\begin{cases}Depth(C_1)/[Depth(C_1)+Depth(C_2)], & \text{If } Depth(C_1)\leq Depth(C_2)\\ 1-Depth(C_1)/[Depth(C_1)+Depth(C_2)], & \text{Otherwise}\end{cases} \quad (5)$$

In the measure of Petrakis et al. [**24**], synsets, glosses, and semantic neighborhoods of concepts are considered as attributes. This measure thinks that two concepts are semantically similar if their synsets, glosses, and neighborhoods (those concepts linked via semantic relations) are lexically similar. It can be expressed as follow:

$$Sim_{\mathbf{Petrakis}}(C_1, C_2) = \begin{cases} 1, & \text{If } Sim_{\text{synsets}}(C_1, C_2) > 0 \\ \max\left\{Sim_{\text{neighborhoods}}(C_1, C_2), Sim_{\text{glosses}}(C_1, C_2)\right\}, & \text{If } Sim_{\text{synsets}}(C_1, C_2) = 0 \end{cases} \quad (6)$$

The similarity for semantic neighborhoods (i.e. $Sim_{\text{neighborhoods}}(C_1, C_2)$) can be computed by the following expression:

$$Sim_{\text{neighborhoods}}(C_1, C_2) = \max\left\{\left|S_i(N_{C1}) \cap S_i(N_{C2})\right| / \left|S_i(N_{C1}) \cup S_i(N_{C2})\right|\right\} \quad (7)$$

where $S(N_{C1})$ and $S(N_{C2})$ are respectively the sets of the semantic neighborhoods of concepts C_1 and C_2 and i is the semantic relation type. Since not all concepts in the semantic neighborhood of a concept are connected with the same semantic relation, the similarity for each different semantic relation type is assessed separately and the maximum one is taken.

Likewise, the similarities for synsets and glosses (i.e. $Sim_{\text{synsets}}(C_1, C_2)$ and $Sim_{\text{glosses}}(C_1, C_2)$) can be both calculated through the following variant of the ratio model (by assigning $\varphi = 1$ and $\psi = 1$ in $Sim_{\mathbf{Tversky\_R}}(T_1, T_2)$):

$$Sim_{\text{synsets/glosses}}(C_1, C_2) = \left|S(X_{C1}) \cap S(X_{C2})\right| / \left|S(X_{C1}) \cup S(X_{C2})\right| \quad (8)$$

where $S(X_{C1})$ and $S(X_{C2})$ are respectively the sets of the synsets or the glosses of concepts C_1 and C_2.

Unlike the measures of Rodr ́guez and Egenhofer [**26**] and Petrakis et al. [**24**], the measure proposed by S ́anchez et al. [**25**] only considers the taxonomic relationships among concepts, which is the most commonly available kind of attributes in ontologies, as its attributes to overcome the limitation that synsets, glosses, distinguishing features, and semantic neighborhoods of concepts are sometimes hardly found in ontologies. This measure introduces the logarithm to a variant of the ratio model (by letting the numerator be $f(S(A_{T1}) \cup S(A_{T2}))$ $- f(S(A_{T1}) \cap S(A_{T2}))$ and assigning $\varphi = 1$ and $\psi = 1$ in $Sim_{\mathbf{Tversky\_R}}(T_1, T_2)$):

$$Sim_{\mathbf{Sanchez}}(C_1, C_2) = 1 - \log_2\left(1 + \frac{\left|\phi(C_1) \setminus \phi(C_2)\right| + \left|\phi(C_2) \setminus \phi(C_1)\right|}{\left|\phi(C_1) \setminus \phi(C_2)\right| + \left|\phi(C_2) \setminus \phi(C_1)\right| + \left|\phi(C_1) \cap \phi(C_2)\right|}\right) \quad (9)$$

where $\phi(C_1) = \{ C \mid C \equiv C_1 \text{ or } C \sqsupseteq C_1 \}$, $\phi(C_2) = \{ C \mid C \equiv C_2 \text{ or } C \sqsupseteq C_2 \}$, and $\phi(C_1) \setminus \phi(C_2)$ is the set of concepts in $\phi(C_1)$ but not in $\phi(C_2)$ (the reverse for $\phi(C_2) \setminus \phi(C_1)$).

In the measure of Jiang et al. [**27**], synonyms, glosses, anchors, and categories of Wikipedia concepts are considered as attributes. This measure thinks that two Wikipedia concepts are semantically similar when their synonyms, glosses, anchors, and categories are lexically similar. It can be expressed as:

$$Sim_{\mathbf{Jiang}}(C_1, C_2) = f(Sim_{\text{synonyms}}(S_{C1}, S_{C2}), Sim_{\text{glosses}}(G_{C1}, G_{C2}), Sim_{\text{anchors}}(A_{C1}, A_{C2}), Sim_{\text{categories}}(C_{C1}, C_{C2})) \quad (10)$$

where f is a weighting or maximum function and $Sim_{\text{synonyms}}(S_{C1}, S_{C2})$, $Sim_{\text{glosses}}(G_{C1}, G_{C2})$, $Sim_{\text{anchors}}(A_{C1}, A_{C2})$, and $Sim_{\text{categories}}(C_{C1}, C_{C2})$ are respectively the similarities of the synonyms, glosses, anchors, and categories of the Wikipedia concepts C_1 and C_2. $Sim_{\text{synonyms}}(S_{C1}, S_{C2})$ is computed by the following expression:

$$Sim_{\text{synonyms}}(S_{C1}, S_{C2}) = \begin{cases} 1, & \text{If } S_{C1} \cap S_{C2} \neq \varnothing \\ 0, & \text{Otherwise} \end{cases} \quad (11)$$

$Sim_{\text{glosses}}(G_{C1}, G_{C2})$, $Sim_{\text{anchors}}(A_{C1}, A_{C2})$, and $Sim_{\text{categories}}(C_{C1}, C_{C2})$ are calculated through simultaneously using

the measure of Rodr ́guez and Egenhofer [26] or the measure of Petrakis et al. [24].

2.2. Measures for concepts in CAD model data ontologies

Semantic similarity measures in attribute-based method are useful in the semantic mapping between two CAD model data ontologies since not every concept in one ontology has semantically equivalent counterpart in the other ontology [16]. Some of these measures have been applied to determine the mapping concept pairs in the semantic interoperability approaches of CAD model data of Patil [16], Lee et al. [17], Zhu et al. [28], and Abdul-Ghafour et al. [19].

The ratio model of Tversky (i.e. $Sim_{\mathbf{Tversky\_R}}(T_1, T_2)$) [23] was used to calculate the semantic similarity of the concepts pairs whose two components are respectively from two different CAD model data ontologies by Patil [16]. The semantic similarity measure for this calculation was defined as:

$$Sim_{\mathbf{Patil}}(C_1, C_2) = \frac{\left|S(C_{DC1}) \cap S(C_{DC2})\right|}{\left|S(C_{DC1}) \cap S(C_{DC2})\right| + u\left|S(C_{DC1}) \setminus S(C_{DC2})\right| + v\left|S(C_{DC2}) \setminus S(C_{DC1})\right|} \tag{12}$$

where $Sim_{\mathbf{Patil}}(C_1, C_2)$ is the semantic similarity of concepts C_1 and C_2, $S(C_{DC1})$ and $S(C_{DC2})$ are respectively the sets of the language constructors in the description logic [29] descriptions of C_1 and C_2, $S(C_{DC1})\backslash S(C_{DC2})$ is the set of language constructors in $S(C_{DC1})$ but not in $S(C_{DC2})$ (the reverse for $S(C_{DC2})\backslash S(C_{DC1})$), $|\cdot|$ is the cardinality of a set, and u and v are respectively the weights of $S(C_{DC1})\backslash S(C_{DC2})$ and $S(C_{DC2})\backslash S(C_{DC1})$ and they are respectively assigned 0.75 and 0.25.

Lee et al. [17] presented the following method to measure the semantic similarity of two concepts:

$$Sim_{\mathbf{Lee}}(C_1, C_2) = \alpha Sim_{\text{name}}(C_1, C_2) + \beta Sim_{\text{definition}}(C_1, C_2) \tag{13}$$

where $Sim_{\mathbf{Lee}}(C_1, C_2)$ is the semantic similarity of concepts C_1 and C_2, $Sim_{\text{name}}(C_1, C_2)$ is the character similarity between the names of C_1 and C_2, $Sim_{\text{definition}}(C_1, C_2)$ is the similarity between the ontological definitions of C_1 and C_2 and is computed by a variant of the measure of Petrakis et al. [24], and α and β are respectively the weights of $Sim_{\text{name}}(C_1, C_2)$ and $Sim_{\text{definition}}(C_1, C_2)$ and are respectively assigned 0.4 and 0.6.

In the approach of Zhu et al. [28], the semantic similarity of two concepts was calculated by the semantic similarity measure of Zhan et al. [18], which is derived from the measure of Petrakis et al. [24]:

$$Sim_{\mathbf{Zhan}}(C_1, C_2) = \left|S(R_{C1}) \cap S(R_{C2})\right| / \left|S(R_{C1}) \cup S(R_{C2})\right| \tag{14}$$

where $Sim_{\mathbf{Zhan}}(C_1, C_2)$ is the semantic similarity of concepts C_1 and C_2 and $S(R_{C1})$ and $S(R_{C2})$ are respectively the sets of the semantic relationships of C_1 and C_2. This measure takes three types of semantic relationships as its attributes when calculating concept similarities: property-of (property) relationship, part-of (composition) relationship, and is-a (inheritance) relationship.

Abdul-Ghafour et al. [19] respectively defined a local similarity measure to compute the similarity of the semantic descriptions of two concepts and a global similarity measure to assess the similarity of the semantic relationships of two concepts. The defined local similarity measure $Sim_{\mathbf{des}}(C_1, C_2)$ is also defined by using the ratio model of Tversky (i.e. $Sim_{\mathbf{Tversky\_R}}(T_1, T_2)$) [23]. So it is identical to $Sim_{\mathbf{Patil}}(C_1, C_2)$. u and v are also respectively assigned 0.75 and 0.25 in this measure. The defined global similarity measure is as follow:

$$Sim_{\mathbf{rel}}(C_1, C_2) = w_1^{\mathbf{C}} Sim(E(C_1), E(C_2)) + w_2^{\mathbf{C}} Sim(S(C_1), S(C_2)) + w_3^{\mathbf{C}} MSim(A_O(C_1), A_O(C_2)) + w_4^{\mathbf{C}} MSim(A_D(C_1), A_D(C_2)) \quad (15)$$

where $Sim_{\mathbf{rel}}(C_1, C_2)$ is the similarity of the semantic relationships of concepts C_1 and C_2, $Sim(E(C_1), E(C_2))$ is the similarity of the equivalent concepts of C_1 and C_2, $Sim(S(C_1), S(C_2))$ is the similarity of the specification (ascendant and descendant) concepts of C_1 and C_2, $MSim(A_O(C_1), A_O(C_2))$ is the multiple similarity of the object roles related to C_1 and C_2, $MSim(A_D(C_1), A_D(C_2))$ is the multiple similarity of the data roles related to C_1 and C_2, $w_1^{\mathbf{C}}$, $w_2^{\mathbf{C}}$, $w_3^{\mathbf{C}}$, and $w_4^{\mathbf{C}}$ are respectively the weights of the contribution components $Sim(E(C_1), E(C_2))$, $Sim(S(C_1), S(C_2))$, $MSim(A_O(C_1), A_O(C_2))$, and $MSim(A_D(C_1), A_D(C_2))$ and they are all assumed as 0.25. After defining the local and global similarity measures, Abdul-Ghafour et al. [19] presented that the semantic similarity of concepts C_1 and C_2 that are respectively from two different CAD model data ontologies can be defined as a weighted sum of $Sim_{\mathbf{des}}(C_1, C_2)$ and $Sim_{\mathbf{rel}}(C_1, C_2)$. They also mentioned that one can select different measures to calculate $Sim_{\mathbf{des}}(C_1, C_2)$ and $Sim_{\mathbf{rel}}(C_1, C_2)$ for different CAD model data ontologies. Such selection was planned in their future work. But there is yet no evidence that a selection method has been proposed.

This paper continues this line of research and proposes a method for selecting a measure with high similarity calculation accuracy for concepts in two different CAD model data ontologies. The main contributions of the paper can be briefly summarized as follows:

- The paper designs an algorithm to compute the weights of the contribution components in a measure. In each method of [16-19], the overall semantic similarity of two concepts is defined as a weighted sum of some contribution components. Thus the weights of the contribution components directly affect the value of the overall semantic similarity and indirectly affect the ontology mapping result. In the methods of [16-19], weights are all manually assigned by the authors. Different authors may possibly assign different weights in an identical situation. As a result, the stability of the measures in these methods is difficult to be ensured. The designed algorithm attempts to calculate the weights according to a certain amount of sample data. It is capable of overcoming this limitation.
- The paper presents an algorithm to find out a measure with high similarity calculation accuracy for concepts in two different CAD model data ontologies. Similarity calculation accuracy is the most important indicator to evaluate a similarity measure. The methods [16-19] only presented their respective semantic similarity measures. There is yet no evidence that they have considered and evaluated the similarity calculation accuracies of their measures. Hence it is not clear how accurate their calculated semantic similarities are. The presented algorithm tries to find out a measure with high similarity calculation accuracy by comparing the similarity calculation accuracies of all candidate measures. To the best of knowledge, this is the first consideration of the similarity calculation accuracy of the measures for concepts in CAD model data ontologies.

3. Measure selection method

This section describes a method to select a semantic similarity measure with high similarity calculation accuracy for concepts in two different CAD model data ontologies. This method is rooted in Abdul-Ghafour et al.'s approach for the semantic interoperability of CAD model data [19], which presented that the semantic

similarity of two concepts can be defined as a weighted sum of the similarity of their semantic descriptions and the similarity of their semantic relationships. Formally, let C_1 and C_2 be two concepts, $Sim_{\mathbf{des}}(D_{C1}, D_{C2})$ be the similarity of the semantic descriptions of C_1 and C_2, and $Sim_{\mathbf{rel}}(R_{C1}, R_{C2})$ be the similarity of the semantic relationships of C_1 and C_2. Then the semantic similarity of C_1 and C_2 can be defined as:

$$Sim(C_1, C_2) = w_1 Sim_{\mathbf{des}}(D_{C1}, D_{C2}) + w_2 Sim_{\mathbf{rel}}(R_{C1}, R_{C2}) \tag{16}$$

where w_1 and w_2 are weights such that $0 \le w_1, w_2 \le 1$ and $w_1 + w_2 = 1$. Abdul-Ghafour et al.'s approach [19] also mentioned that one can select different measures to calculate $Sim_{\mathbf{des}}(D_{C1}, D_{C2})$ and $Sim_{\mathbf{rel}}(R_{C1}, R_{C2})$ for different CAD model data ontologies. Now two questions arise from this approach: (1) How to obtain the weights w_1 and w_2 that can ensure the stability of the measure $Sim(C_1, C_2)$? (2) How to select two measures that make $Sim(C_1, C_2)$ have high similarity calculation accuracy to respectively calculate out $Sim_{\mathbf{des}}(D_{C1}, D_{C2})$ and $Sim_{\mathbf{rel}}(R_{C1}, R_{C2})$? Because there is yet no evidence that a solution for these two questions has been proposed, the described method in the section will respectively answer them through designing a weight calculation algorithm and a measure selection algorithm.

3.1. Weight calculation algorithm

In each method of [16-19], the overall semantic similarity of two concepts is defined as a weighted sum of two or more contribution components. Without loss of generality, the situation of a measure with n (n = 1, 2, …) contribution components is considered. Let C_1 and C_2 be two concepts that are respectively from two CAD model data ontologies, $f_1(C_1, C_2)$, $f_2(C_1, C_2)$, …, $f_n(C_1, C_2)$ be n contribution components of the overall semantic similarity of C_1 and C_2 (denoted as $Sim(C_1, C_2)$), and w_1, w_2, …, w_n be respectively the weights of $f_1(C_1, C_2)$, $f_2(C_1, C_2)$, …, $f_n(C_1, C_2)$ such that $0 \le w_1, w_2, \ldots, w_n \le 1$ and $w_1 + w_2 + \ldots + w_n = 1$. The measures of the overall semantic similarity in the methods [16-19] can be expressed in the following unified form:

$$Sim(C_1, C_2) = w_1 f_1(C_1, C_2) + w_2 f_2(C_1, C_2) + \cdots + w_n f_n(C_1, C_2) \tag{17}$$

As can be seen from this expression, when the values of $f_1(C_1, C_2)$, $f_2(C_1, C_2)$, …, $f_n(C_1, C_2)$ are certain, the value of $Sim(C_1, C_2)$ is determined by the values of w_1, w_2, …, w_n. That is to say, the weights of the contribution components have a direct influence on the result and the similarity calculation accuracy of a semantic similarity measure. Thus a weight calculation algorithm which can ensure similarity calculation accuracy is of great necessity for a semantic similarity measure. Since there is yet no evidence that the methods [16-19] have provided such an algorithm, this paper designs one to further improve these methods and to calculate the weights in the presented measure selection algorithm.

In general, the similarity calculation accuracy of a measure can be evaluated by the Pearson correlation coefficient between the similarities of a certain number of sample concept pairs calculated by the measure and the actual similarities of these sample concept pairs (normally it is impossible to obtain the actual similarity of a sample concept pair and thus this actual similarity is always replaced by a mean value of the similarities of this sample concept pair judged by a certain number of domain experts). The greater this correlation coefficient, the higher the similarity calculation accuracy is [24-27]. However, if the correlation coefficients of different measures only have minor differences (the difference between each two of them is less than 0.1), it would not be able to conclude which measure is better. In this situation, the present paper uses the re-

sidual sum of squares between the calculated similarities and the actual similarities to evaluate a measure. The smaller this residual sum of squares, the higher the similarity calculation accuracy is.

Based on these two situations, the designed algorithm should firstly calculate a group of weights that can maximize the Pearson correlation coefficient. If the correlation coefficients of different measures have significant differences (the difference between two of them is greater than 0.1), which measure is better will be directly concluded. Otherwise, the algorithm should calculate another group of weights that can minimize the residual sum of squares. Then a better measure will be determined by comparing all calculated residual sums of squares. Thus the algorithm consists of two procedures: A procedure of computing weights by maximizing Pearson correlation coefficient and a procedure of computing weights by minimizing residual sum of squares. The details of the procedure of computing weights by maximizing Pearson correlation coefficient are firstly explained.

Formally, let N be the number of the sample concept pairs whose semantic similarities need to be measured (N is at least equal to 30), $A_i(C_{i,1}, C_{i,2})$ (i = 1, 2, …, N) be the actual semantic similarity of the i-th concept pair $(C_{i,1}, C_{i,2})$, $\boldsymbol{U} = [f_{i,1}(C_{i,1}, C_{i,2}), f_{i,2}(C_{i,1}, C_{i,2}),\ldots, f_{i,n}(C_{i,1}, C_{i,2})]^{\mathrm{T}}$ be a vector, $V = [A_i(C_{i,1}, C_{i,2})]^{\mathrm{T}}$ be a vector, and $\boldsymbol{w} = [w_1, w_2, \ldots, w_n]^{\mathrm{T}}$ be a vector. Then the Pearson correlation coefficient between the actual semantic similarities and the calculated semantic similarities of these N concept pairs is the Pearson correlation coefficient between $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{U}$ and $\boldsymbol{V}$:

$$\mathrm{corr}(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{U},\ \boldsymbol{V}) = \frac{\boldsymbol{w}^{\mathrm{T}}\,\mathrm{cov}(\boldsymbol{U},\boldsymbol{V})}{\sqrt{\boldsymbol{w}^{\mathrm{T}}\,\mathrm{cov}(\boldsymbol{U},\boldsymbol{U})\boldsymbol{w}}\sqrt{\mathrm{cov}(\boldsymbol{V},\boldsymbol{V})}} = \frac{\boldsymbol{w}^{\mathrm{T}}\sum_{\boldsymbol{UV}}}{\sqrt{\boldsymbol{w}^{\mathrm{T}}\sum_{\boldsymbol{UU}}\boldsymbol{w}}\sqrt{\sum_{\boldsymbol{VV}}}} \tag{18}$$

where cov is short for covariance and $\mathrm{cov}(\boldsymbol{U}, \boldsymbol{U}) = \sum_{\boldsymbol{UU}}$, $\mathrm{cov}(\boldsymbol{U}, \boldsymbol{V}) = \sum_{\boldsymbol{UV}}$, and $\mathrm{cov}(\boldsymbol{V}, \boldsymbol{V}) = \sum_{\boldsymbol{VV}}$ are respectively the following matrices:

$$\sum_{\boldsymbol{UU}} = \begin{bmatrix} \mathrm{cov}(f_{i,1}(C_{i,1},C_{i,2}), f_{i,1}(C_{i,1},C_{i,2})) & \mathrm{cov}(f_{i,1}(C_{i,1},C_{i,2}), f_{i,2}(C_{i,1},C_{i,2})) & \cdots & \mathrm{cov}(f_{i,1}(C_{i,1},C_{i,2}), f_{i,n}(C_{i,1},C_{i,2})) \\ \mathrm{cov}(f_{i,2}(C_{i,1},C_{i,2}), f_{i,1}(C_{i,1},C_{i,2})) & \mathrm{cov}(f_{i,2}(C_{i,1},C_{i,2}), f_{i,2}(C_{i,1},C_{i,2})) & \cdots & \mathrm{cov}(f_{i,2}(C_{i,1},C_{i,2}), f_{i,n}(C_{i,1},C_{i,2})) \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{cov}(f_{i,n}(C_{i,1},C_{i,2}), f_{i,1}(C_{i,1},C_{i,2})) & \mathrm{cov}(f_{i,n}(C_{i,1},C_{i,2}), f_{i,2}(C_{i,1},C_{i,2})) & \cdots & \mathrm{cov}(f_{i,n}(C_{i,1},C_{i,2}), f_{i,n}(C_{i,1},C_{i,2})) \end{bmatrix} \tag{19}$$

$$\sum_{\boldsymbol{UV}} = \begin{bmatrix} \mathrm{cov}(f_{i,1}(C_{i,1},C_{i,2}), A_i(C_{i,1},C_{i,2})) \\ \mathrm{cov}(f_{i,2}(C_{i,1},C_{i,2}), A_i(C_{i,1},C_{i,2})) \\ \vdots \\ \mathrm{cov}(f_{i,n}(C_{i,1},C_{i,2}), A_i(C_{i,1},C_{i,2})) \end{bmatrix} \tag{20}$$

$$\sum_{\boldsymbol{VV}} = \begin{bmatrix} \mathrm{cov}(A_i(C_{i,1},C_{i,2}), A_i(C_{i,1},C_{i,2})) \end{bmatrix} \tag{21}$$

To solve the vector $\boldsymbol{w}$ that can maximize $\mathrm{corr}(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{U}, \boldsymbol{V})$, the canonical correlation analysis method [**30**] is used. The obtained solution is: $\boldsymbol{w}$ is an eigenvector with the maximum eigenvalue for the matrix $\sum_{\boldsymbol{UU}}^{-1}\sum_{\boldsymbol{UV}}\sum_{\boldsymbol{VV}}^{-1}\sum_{\boldsymbol{VU}}$, where $\sum_{\boldsymbol{VU}}$ is the following matrix:

$$\sum_{\boldsymbol{VU}} = \begin{bmatrix} \mathrm{cov}(A_i(C_{i,1},C_{i,2}), f_{i,1}(C_{i,1},C_{i,2})) & \mathrm{cov}(A_i(C_{i,1},C_{i,2}), f_{i,2}(C_{i,1},C_{i,2})) & \cdots & \mathrm{cov}(A_i(C_{i,1},C_{i,2}), f_{i,n}(C_{i,1},C_{i,2})) \end{bmatrix} \tag{22}$$

However, the elements of $\boldsymbol{w}$ are not the final weights of the contribution components $f_1(C_1, C_2)$, $f_2(C_1, C_2)$, …,

$f_n(C_1, C_2)$ because some of these elements may be smaller than 0 and the sum of the remaining elements (the elements that are not smaller than 0) is usually not equal to 1. The final weights are solved by the following normalization: Let $\boldsymbol{w}^{\mathbf{o}} = [w_1^{\mathbf{o}}, w_2^{\mathbf{o}}, \ldots, w_n^{\mathbf{o}}]^{\mathrm{T}}$ be the vector solved by the canonical correlation analysis method and $\boldsymbol{w} = [w_1, w_2, \ldots, w_n]^{\mathrm{T}}$ be the final weight vector. For all $w_j^{\mathbf{o}} < 0$ ($j = 1, 2, \ldots, n$), let $w_j^{\mathbf{o}} = 0$ and $w_j = 0$ and then let $w_j = w_j^{\mathbf{o}} / (w_1^{\mathbf{o}} + w_2^{\mathbf{o}} + \ldots + w_n^{\mathbf{o}})$. The next section will prove that the vector $\boldsymbol{w} = [w_1, w_2, \ldots, w_n]^{\mathrm{T}}$ obtained from such normalization is also a vector that can maximize the Pearson correlation coefficient.

Now the details of the procedure of how to compute weights by minimizing residual sum of squares are explained. Likewise, let N be the number of the sample concept pairs whose semantic similarities require to be measured (N is at least equal to 30), $A_i(C_{i,1}, C_{i,2})$ ($i = 1, 2, \ldots, N$) be the actual semantic similarity of the i-th concept pair $(C_{i,1}, C_{i,2})$, $\boldsymbol{X} = [f_{i,1}(C_{i,1}, C_{i,2}), f_{i,2}(C_{i,1}, C_{i,2}), \ldots, f_{i,n}(C_{i,1}, C_{i,2})]^{\mathrm{T}}$ be a vector, $\boldsymbol{Y} = [A_i(C_{i,1}, C_{i,2})]^{\mathrm{T}}$ be a vector, and $\boldsymbol{w} = [w_1, w_2, \ldots, w_n]^{\mathrm{T}}$ be a vector. Then the residual sum of squares between the actual semantic similarities and the calculated semantic similarities of these N concept pairs is the residual sum of squares between $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{X}$ and $\boldsymbol{Y}$:

$$\mathrm{rss}(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{X}, \boldsymbol{Y}) = \sum_{i=1}^{N}\left[Sim(C_{i,1}, C_{i,2}) - A_i(C_{i,1}, C_{i,2})\right]^2 = \sum_{i=1}^{N}(w_1 f_{i,1} + w_2 f_{i,2} + \cdots + w_n f_{i,n} - A_i)^2 \tag{23}$$

where $f_{i,j}$ ($j = 1, 2, \ldots, n$) and A_i are respectively $f_{i,j}(C_{i,1}, C_{i,2})$ and $A_i(C_{i,1}, C_{i,2})$ for simplicity. Since $w_n = 1 - w_1 - w_2 - \ldots - w_{n-1}$, $\mathrm{rss}(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{X}, \boldsymbol{Y})$ can be transformed as:

$$\mathrm{rss}(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{X}, \boldsymbol{Y}) = \sum_{i=1}^{N}\left[(f_{i,1} - f_{i,n})w_1 + (f_{i,2} - f_{i,n})w_2 + \cdots + (f_{i,n-1} - f_{i,n})w_{n-1} + (f_{i,n} - A_i)\right]^2 \tag{24}$$

Rrespectively take partial differentiation on $\mathrm{rss}(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{X}, \boldsymbol{Y})$ with respect to $w_1, w_2, \ldots, w_{n-1}$:

$$\begin{cases} \dfrac{\partial RSS}{\partial w_1} = 2\sum\limits_{i=1}^{N}\left[(f_{i,1} - f_{i,n})^2 w_1 + (f_{i,1} - f_{i,n})(f_{i,2} - f_{i,n})w_2 + \cdots + (f_{i,1} - f_{i,n})(f_{i,n-1} - f_{i,n})w_{n-1} + (f_{i,1} - f_{i,n})(f_{i,n} - A_i)\right] \\ \dfrac{\partial RSS}{\partial w_2} = 2\sum\limits_{i=1}^{N}\left[(f_{i,1} - f_{i,n})(f_{i,2} - f_{i,n})w_1 + (f_{i,2} - f_{i,n})^2 w_2 + \cdots + (f_{i,2} - f_{i,n})(f_{i,n-1} - f_{i,n})w_{n-1} + (f_{i,2} - f_{i,n})(f_{i,n} - A_i)\right] \\ \vdots \\ \dfrac{\partial RSS}{\partial w_{n-1}} = 2\sum\limits_{i=1}^{N}\left[(f_{i,1} - f_{i,n})(f_{i,n-1} - f_{i,n})w_1 + (f_{i,2} - f_{i,n})(f_{i,n-1} - f_{i,n})w_2 + \cdots + (f_{i,n-1} - f_{i,n})^2 w_{n-1} + (f_{i,n-1} - f_{i,n})(f_{i,n} - A_i)\right] \end{cases} \tag{25}$$

After setting each partial differentiation equal to 0, a linear equation set $\boldsymbol{A}\boldsymbol{w}' = \boldsymbol{b}$ is obtained, where $\boldsymbol{w}' = [w_1, w_2, \ldots, w_{n-1}]^{\mathrm{T}}$ and

$$\boldsymbol{A} = \begin{bmatrix} \sum\limits_{i=1}^{N}(f_{i,1} - f_{i,n})(f_{i,1} - f_{i,n}) & \sum\limits_{i=1}^{N}(f_{i,1} - f_{i,n})(f_{i,2} - f_{i,n}) & \cdots & \sum\limits_{i=1}^{N}(f_{i,1} - f_{i,n})(f_{i,n-1} - f_{i,n}) \\ \sum\limits_{i=1}^{N}(f_{i,1} - f_{i,n})(f_{i,2} - f_{i,n}) & \sum\limits_{i=1}^{N}(f_{i,2} - f_{i,n})(f_{i,2} - f_{i,n}) & \cdots & \sum\limits_{i=1}^{N}(f_{i,2} - f_{i,n})(f_{i,n-1} - f_{i,n}) \\ \vdots & \vdots & \ddots & \vdots \\ \sum\limits_{i=1}^{N}(f_{i,1} - f_{i,n})(f_{i,n-1} - f_{i,n}) & \sum\limits_{i=1}^{N}(f_{i,2} - f_{i,n})(f_{i,n-1} - f_{i,n}) & \cdots & \sum\limits_{i=1}^{N}(f_{i,n-1} - f_{i,n})(f_{i,n-1} - f_{i,n}) \end{bmatrix} \tag{26}$$

$$\boldsymbol{b} = \left[\sum_{i=1}^{N} (f_{i,n} - f_{i,1})(f_{i,n} - A_i) \quad \sum_{i=1}^{N} (f_{i,n} - f_{i,2})(f_{i,n} - A_i) \quad \cdots \quad \sum_{i=1}^{N} (f_{i,n} - f_{i,n-1})(f_{i,n} - A_i) \right]^{\mathrm{T}} \tag{27}$$

This is an $n-1$-variables non-homogeneous linear equation set. The coefficient matrix of this equation set (i.e. the matrix $\boldsymbol{A}$) is a symmetric matrix. All leading diagonal elements of this matrix are greater than 0. Based on these characteristics, there are a number of methods that could be used to solve the equation set, where representative ones are Gauss elimination, Gauss-Jordan elimination, Dolittle decomposition, Courant decomposition, Jacobi iteration, and Gauss-Seidel iteration methods. The Gauss elimination method is chose to solve the equation set for these two reasons: (1) The characteristics of $\boldsymbol{A}$ exactly meet the application condition of the Gauss elimination method. (2) Whether the characteristics of $\boldsymbol{A}$ can match the application conditions of other methods relies on the semantic similarities computed by each contribution component. Sometimes these methods cannot be applied to solve the equation set. After applying the Gauss elimination method, a solution vector $\boldsymbol{w'} = [w_1^{\mathbf{o}}, w_2^{\mathbf{o}}, \ldots, w_{n-1}^{\mathbf{o}}]^{\mathrm{T}}$ is obtained. For each $w_i^{\mathbf{o}}$ ($i = 1, 2, \ldots, n-1$): (1) If $w_i^{\mathbf{o}} > 1$, then $w_i = 1$ and $w_j = 0$ ($j = 1, 2, \ldots, n$ and $j \neq i$). (2) If $w_i^{\mathbf{o}} < 0$, then $w_i = 0$; if $0 \leq w_i^{\mathbf{o}} \leq 1$, then $w_i = w_i^{\mathbf{o}}$; and finally $w_n = 1 - w_1 - w_2 - \ldots - w_{n-1}$.

Based on the above explanations, an algorithm for calculating the n ($n = 1, 2, \ldots$) weights of the n contribution components in a semantic similarity measure can be designed. The designed algorithm, whose description is provided in Appendix A, consists of two procedures. The first procedure is used to compute the weights that can maximize Pearson correlation coefficient. The main ideas behind this procedure are informally described as follows. The procedure firstly calculates out the values of all of the elements of $\sum_{\boldsymbol{UU}}$, $\sum_{\boldsymbol{UV}}$, $\sum_{\boldsymbol{VV}}$, and $\sum_{\boldsymbol{VU}}$ and the matrix $\sum_{\boldsymbol{UU}}^{-1} \sum_{\boldsymbol{UV}} \sum_{\boldsymbol{VV}}^{-1} \sum_{\boldsymbol{VU}}$ according to the input n sets $\{f_{i,1}(C_{i,1}, C_{i,2})\}$, $\{f_{i,2}(C_{i,1}, C_{i,2})\}$,…, $\{f_{i,n}(C_{i,1}, C_{i,2})\}$ ($i = 1, 2,\ldots, N$) and set $\{A_i(C_{i,1}, C_{i,2})\}$. Then it seeks the eigenvector with the maximum eigenvalue for $\sum_{\boldsymbol{UU}}^{-1} \sum_{\boldsymbol{UV}} \sum_{\boldsymbol{VV}}^{-1} \sum_{\boldsymbol{VU}}$ and computes out the n weights of the n contribution components $w_1, w_2, \ldots, w_n$ based on this eigenvector. The second procedure is used to compute the weights that can minimize residual sum of squares. The main ideas behind this procedure are also informally described as follows. The procedure firstly computes out the coefficient matrix $\boldsymbol{A}$ and the right end constant vector $\boldsymbol{b}$ of $\boldsymbol{Aw'} = \boldsymbol{b}$ according to the input n sets $\{f_{i,1}(C_{i,1}, C_{i,2})\}$, $\{f_{i,2}(C_{i,1}, C_{i,2})\}$,…, $\{f_{i,n}(C_{i,1}, C_{i,2})\}$ ($i = 1, 2,\ldots, N$) and set $\{A_i(C_{i,1}, C_{i,2})\}$. Then it applies the Gauss elimination method to solve the equation set and obtain the solution $w_1^{\mathbf{o}}, w_2^{\mathbf{o}}, \ldots, w_{n-1}^{\mathbf{o}}$. Finally, it computes out the n weights of the n contribution components $w_1, w_2, \ldots, w_n$.

The complexity of the designed weight calculation algorithm is analyzed as follows: (1) The complexity of the first procedure. 1) The computation amount of **Step 1**. The computation of each covariance requires N multiplications and $2N + 1$ divisions. Since there are totally $n^2 + 2n + 1$ covariances in the four matrices $\sum_{\boldsymbol{UU}}$, $\sum_{\boldsymbol{UV}}$, $\sum_{\boldsymbol{VV}}$, and $\sum_{\boldsymbol{VU}}$, the computation of the values of all of the elements in these four matrices needs $(3N + 1)(n^2 + 2n + 1)$ multiplications/divisions. The computation of the matrix $\sum_{\boldsymbol{UU}}^{-1} \sum_{\boldsymbol{UV}} \sum_{\boldsymbol{VV}}^{-1} \sum_{\boldsymbol{VU}}$ requires $2n^2 + 2n + 1$ multiplications/divisions. So this step totally needs $(3N+3)n^2 + (6N+4)n + 3N + 2$ multiplications/divisions. 2) The computation amount of **Step 2**. The complexity of seeking the eigenvector with the maximum eigenvalue for $\sum_{\boldsymbol{UU}}^{-1} \sum_{\boldsymbol{UV}} \sum_{\boldsymbol{VV}}^{-1} \sum_{\boldsymbol{VU}}$ is $O(n^3)$. 3) The computation amount of **Step 3**. This step requires n divisions. As can be concluded from the computation amounts of these three steps, the complexity of the first procedure is $O(n^3)$. (2) The complexity of the second procedure. 1) The computation amount of **Step 1**. The computation

of each element in $\boldsymbol{A}$ requires N multiplications. Since there are totally $(n-1)^2$ elements in $\boldsymbol{A}$, the computation of $\boldsymbol{A}$ needs $N(n-1)^2$ multiplications. The computation of each element in $\boldsymbol{b}$ needs N multiplications. There are totally $n-1$ elements in $\boldsymbol{b}$. Thus the computation of $\boldsymbol{b}$ needs $N(n-1)$ multiplications and this step totally requires $N(n-1)^2+ N(n-1)$ multiplications. 2) The computation amount of **Step 2**. The complexity of the Gauss elimination method is $O(n^3)$. It can be concluded from the computation amounts of these two steps that the complexity of the second procedure is $O(n^3)$.

3.2. Measure selection algorithm

The commonly used semantic similarity measures in attribute-based method are Tversky's measure (Expression (4)) [23], Petrakis et al.'s measure (Expression (8)) [24], and Sánchez et al.'s measure (Expression (9)) [25]. These three measures are taken as candidate measures to illustrate how to select two measures that make $Sim(C_1, C_2)$ (Expression (16)) have high similarity calculation accuracy to respectively calculate out the contribution components $Sim_{\mathbf{des}}(D_{C1}, D_{C2})$ and $Sim_{\mathbf{rel}}(R_{C1}, R_{C2})$. Since each contribution component has three options, the following nine measures can be derived from Expression (16):

$$\begin{cases} Sim_1(C_1,C_2) = w_{1,1}Sim_{\mathbf{Tversky}}(D_{C1},D_{C2}) + w_{1,2}Sim_{\mathbf{Tversky}}(R_{C1},R_{C2}) \\ Sim_2(C_1,C_2) = w_{2,1}Sim_{\mathbf{Tversky}}(D_{C1},D_{C2}) + w_{2,2}Sim_{\mathbf{Petrakis}}(R_{C1},R_{C2}) \\ Sim_3(C_1,C_2) = w_{3,1}Sim_{\mathbf{Tversky}}(D_{C1},D_{C2}) + w_{3,2}Sim_{\mathbf{Sanchez}}(R_{C1},R_{C2}) \\ Sim_4(C_1,C_2) = w_{4,1}Sim_{\mathbf{Petrakis}}(D_{C1},D_{C2}) + w_{4,2}Sim_{\mathbf{Tversky}}(R_{C1},R_{C2}) \\ Sim_5(C_1,C_2) = w_{5,1}Sim_{\mathbf{Petrakis}}(D_{C1},D_{C2}) + w_{5,2}Sim_{\mathbf{Petrakis}}(R_{C1},R_{C2}) \\ Sim_6(C_1,C_2) = w_{6,1}Sim_{\mathbf{Petrakis}}(D_{C1},D_{C2}) + w_{6,2}Sim_{\mathbf{Sanchez}}(R_{C1},R_{C2}) \\ Sim_7(C_1,C_2) = w_{7,1}Sim_{\mathbf{Sanchez}}(D_{C1},D_{C2}) + w_{7,2}Sim_{\mathbf{Tversky}}(R_{C1},R_{C2}) \\ Sim_8(C_1,C_2) = w_{8,1}Sim_{\mathbf{Sanchez}}(D_{C1},D_{C2}) + w_{8,2}Sim_{\mathbf{Petrakis}}(R_{C1},R_{C2}) \\ Sim_9(C_1,C_2) = w_{9,1}Sim_{\mathbf{Sanchez}}(D_{C1},D_{C2}) + w_{9,2}Sim_{\mathbf{Sanchez}}(R_{C1},R_{C2}) \end{cases} \tag{28}$$

where $Sim_j(C_1, C_2)$ (j = 1, 2, …, 9) is the j-th semantic similarity measure for C_1 and C_2, and $w_{j,1}$ and $w_{j,2}$ are nine pairs of weights such that $0 \le w_{j,1}, w_{j,2} \le 1$ and $w_{j,1} + w_{j,2} = 1$. The values of $w_{j,1}$ and $w_{j,2}$ can be worked out by the two procedures in the designed weight calculation algorithm.

Now another question arises: How to make a choice among the nine derived measures? A direct and effective solution is to choose the measure that obtains the highest similarity calculation accuracy. Since generally the similarity calculation accuracy of a measure can be quantified by the Pearson correlation coefficient between the similarities of a certain number of sample concept pairs computed by this measure and the actual similarities of these sample concept pairs (normally it is impossible to obtain the actual similarity of a sample concept pair and thus this actual similarity is always replaced by a mean value of the similarities of this sample concept pair judged by a certain number of domain experts) [24-27], one can use the first procedure of the designed weight calculation algorithm to compute nine groups of weights and then calculate nine correlation coefficients and choose the measure that obtains the greatest Pearson correlation coefficient from these nine derived measures. However, if the correlation coefficients of the nine derived measures only have minor differences (the difference between each two of them is less than 0.1), it would not be able to conclude which measure is the best. In this situation, one can apply the second procedure of the designed weight calculation algorithm to compute nine groups of weights and then calculate nine residual sums of squares. The measure

that has the least residual sum of squares is selected for concepts in two different CAD model data ontologies.

Formally, let O_1 and O_2 be two different CAD model data ontologies, N_1 be the number of sample concepts that are arbitrarily extracted from O_1, and N_2 be the number of sample concepts that are arbitrarily extracted from O_2. The number of all possible sample concept pairs is $N = N_1N_2$ (generally N is at least 30 when extracting the sample concepts from O_1 and O_2). The semantic similarities of these N sample concept pairs can be respectively assessed by $Sim_j(C_{i,1}, C_{i,2})$ (i = 1, 2, …, N and j = 1, 2, …, 9). For all these assessed semantic similarities, let $Sim_{i,j}(C_{i,1}, C_{i,2})$ (i = 1, 2, …, N and j = 1, 2, …, 9) be the sematic similarity of the i-th sample concept pair that is assessed by $Sim_j(C_{i,1}, C_{i,2})$ and $A_i(C_{i,1}, C_{i,2})$ (i = 1, 2, …, N) be the actual semantic similarity of the i-th sample concept pair. The Pearson correlation coefficient and the residual sum of squares between vectors $\boldsymbol{A} = [Sim_{i,j}(C_{i,1}, C_{i,2})]^{\mathrm{T}}$ and $\boldsymbol{B} = [A_i(C_{i,1}, C_{i,2})]^{\mathrm{T}}$ can be respectively expressed as:

$$\mathrm{corr}(\boldsymbol{A}, \boldsymbol{B}) = \frac{\mathrm{cov}([Sim_{i,j}(C_{i,1}, C_{i,2})]^{\mathrm{T}}, [A_i(C_{i,1}, C_{i,2})]^{\mathrm{T}})}{\sqrt{\mathrm{cov}([Sim_{i,j}(C_{i,1}, C_{i,2})]^{\mathrm{T}}, [Sim_{i,j}(C_{i,1}, C_{i,2})]^{\mathrm{T}})}\sqrt{\mathrm{cov}([A_i(C_{i,1}, C_{i,2})]^{\mathrm{T}}, [A_i(C_{i,1}, C_{i,2})]^{\mathrm{T}})}} \tag{29}$$

$$\mathrm{rss}(\boldsymbol{A}, \boldsymbol{B}) = \sum_{i=1}^{N}\left[Sim_{i,j}(C_{i,1}, C_{i,2}) - A_i(C_{i,1}, C_{i,2})\right]^2 \tag{30}$$

where cov is short for covariance. One can firstly use Expression (29) to calculate out $\mathrm{corr}([Sim_{i,1}(C_{i,1}, C_{i,2})]^{\mathrm{T}}, \boldsymbol{B})$, $\mathrm{corr}([Sim_{i,2}(C_{i,1}, C_{i,2})]^{\mathrm{T}}, \boldsymbol{B})$,…, $\mathrm{corr}([Sim_{i,9}(C_{i,1}, C_{i,2})]^{\mathrm{T}}, \boldsymbol{B})$ and then choose the measure $Sim_j(C_1, C_2)$ (where $\mathrm{corr}([Sim_{i,j}(C_{i,1}, C_{i,2})]^{\mathrm{T}}, \boldsymbol{B})$ is the greatest correlation coefficient among these nine correlation coefficients) to compute the semantic similarities of all possible concept pairs whose two components are respectively from O_1 and O_2 and are not semantically equivalent. However, if the difference between each two of the nine correlation coefficients is less than 0.1, one can utilize Expression (30) to calculate out $\mathrm{rss}([Sim_{i,1}(C_{i,1}, C_{i,2})]^{\mathrm{T}}, \boldsymbol{B})$, $\mathrm{rss}([Sim_{i,2}(C_{i,1}, C_{i,2})]^{\mathrm{T}}, \boldsymbol{B})$,…, $\mathrm{rss}([Sim_{i,9}(C_{i,1}, C_{i,2})]^{\mathrm{T}}, \boldsymbol{B})$ and select $Sim_j(C_1, C_2)$ (where $\mathrm{rss}([Sim_{i,j}(C_{i,1}, C_{i,2})]^{\mathrm{T}}, \boldsymbol{B})$ is the least residual sum of squares among these nine residual sums of squares) for concepts in O_1 and O_2.

Based on the above selection process, an algorithm for determining a measure with high similarity calculation accuracy for concepts in two different CAD model data ontologies can be designed. The designed algorithm, whose description is provided in Appendix B, takes as input two different CAD model data ontologies O_1 and O_2 which are represented in OWL DL (i.e. description logic SHOIN(D) [**31**]), and it returns as output a measure with high similarity calculation accuracy for the concepts between O_1 and O_2. The main ideas behind the algorithm are informally described as follows. The reasoning on a combination of O_1 and O_2 is firstly performed to find out the semantically equivalent pairs. After this reasoning, N_1 sample concepts in O_1 that do not have semantically equivalent concepts in O_2 and N_2 sample concepts in O_2 that do not have semantically equivalent concepts in O_1 are respectively extracted. Then $N = N_1N_2$ sample concept pairs are constructed and their semantic similarities are judged by a certain number of domain experts. These judged semantic similarities are taken as the actual semantic similarities of these N sample concept pairs. For the N sample concept pairs, the similarities between their semantic descriptions and between their semantic relationships are calculated and then the nine groups of weights are computed by the first procedure in the weight calculation algorithm. After that, the semantic similarity of each pair and the Pearson correlation coefficient of each measure are successively worked out. If the difference between some two of the nine correlation co-

efficients is greater than or equal to 0.1, the measure for the concepts between O_1 and O_2 is selected as the one having the greatest correlation coefficient. Otherwise, the second procedure in the weight calculation algorithm is used to compute another nine groups of weights and similarly the semantic similarity of each pair and the residual sum of squares of each measure are successively worked out. Then the measure for the concepts between O_1 and O_2 is selected as the one that has the least residual sum of squares.

The complexity of the designed measure selection algorithm is analyzed as follows: (1) The complexity of reasoning on the two CAD model data ontologies (**Step 1**). It has been proved that the concept satisfiability problem is NExpTime-complete for SHOIN(D) [32] and the reasoning problem is ExpTime-complete for OWL DL ontologies combining with DL-safe SWRL rules [33]. (2) The computation amount of **Step 2**. The calculation of the semantic similarities between the semantic descriptions and between the semantic relationships of the N sample concept pairs respectively needs $4N$ and $3N$ divisions. (3) The computation amount of **Step 3**. The computation complexity of calculating the nine groups of weights is $9O(2^3)$. (4) The computation amount of **Step 4**. Using the nine measures for concepts to calculate the semantic similarities of the N sample concept pairs needs $18N$ multiplications. (5) The computation amount of **Step 5**. The computation of the nine correlation coefficients requires $9(3N+11)$ multiplications/divisions. (6) The computation amount of **Step 6**. The computation complexity of calculating the nine groups of weights is $9O(2^3)$. Using the nine measures to calculate the semantic similarities of the N sample concept pairs needs $18N$ multiplications. The computation of the nine residual sums of squares requires $9N$ multiplications. As can be concluded from the above analysis of the computation amounts of the six steps, except for the complexity of reasoning on ontologies, the complexity of the measure selection algorithm is $O(N)$.

4. Implementation, example, and evaluation

This section first reports a prototype implementation of the proposed measure selection method. It then presents an example to illustrate how the proposed method works. Finally, the section evaluates the proposed method through theoretical and experimental comparisons.

4.1. Implementation

The CAD model data ontologies can be manually developed with the OWL DL and SWRL languages in Protégé [34], a free and open-source ontology editor providing an integration environment of creating, editing and saving OWL DL/SWRL ontologies in a visual way. Because this paper does not aim to explain how to develop a comprehensive CAD model data ontology, it just develops a PROE feature data ontology and a UGNX feature data ontology for the purpose of simplicity. Zhu et al. [28] has developed a PROE assembly feature data ontology and Patil [16] has developed a UGNX feature data ontology. These two ontologies are reused when developing the PROE and UGNX feature data ontologies.

After developing the PROE and UGNX feature data ontologies, the reasoning on a combination of them is performed by using the Jess reasoner [35]. The proposed measure selection method is then developed using Protégé-OWL application program interface (API) and Java programming language.

4.2. Example

An example is presented to illustrate how the proposed measure selection method works. An initializa-

tion work of the illustration is to perform reasoning on a combination of the PROE and UGNX feature data ontologies, which can be done by the Jess reasoner [35]. After performing the reasoning, the concepts in each ontology which do not have semantically equivalent counterparts are obtained. Then eleven sample concepts which respectively represent six different types of PROE hole features and five different types of UGNX hole features are extracted from these obtained concepts. Hence, as shown in Table 1, there are totally thirty possible sample concept pairs between these extracted sample concepts.

Table 1

All possible sample concept pairs between the six sample concepts extracted from the PROE feature data ontology and the five sample concepts extracted from the UGNX feature data ontology.

Concepts in UGNX Concepts in PROE	*UGNX-GeneralHole*	*UGNX-DrillSizeHole*	*UGNX-ScrewClearanceHole*	*UGNX-ThreadedHole*	*UGNX-HoleSeries*
PROE-RectangleProfileHole	$CP_{1,1}$	$CP_{1,2}$	$CP_{1,3}$	$CP_{1,4}$	$CP_{1,5}$
PROE-StandardProfileHole	$CP_{2,1}$	$CP_{2,2}$	$CP_{2,3}$	$CP_{2,4}$	$CP_{2,5}$
PROE-SketchProfileHole	$CP_{3,1}$	$CP_{3,2}$	$CP_{3,3}$	$CP_{3,4}$	$CP_{3,5}$
PROE-DrilledHole	$CP_{4,1}$	$CP_{4,2}$	$CP_{4,3}$	$CP_{4,4}$	$CP_{4,5}$
PROE-ClearanceHole	$CP_{5,1}$	$CP_{5,2}$	$CP_{5,3}$	$CP_{5,4}$	$CP_{5,5}$
PROE-TaperedHole	$CP_{6,1}$	$CP_{6,2}$	$CP_{6,3}$	$CP_{6,4}$	$CP_{6,5}$

Now the thirty sample concept pairs are used as sample concept pairs to select a measure with high similarity calculation accuracy for concepts in the PROE and UGNX feature data ontologies. The selection process can be divided into four steps.

The first step is to get the actual semantic similarities of these thirty sample concept pairs. As mentioned before, it is always impossible to obtain the actual semantic similarity of a concept pair because the notion of similarity is a subjective human judgement. Many researchers in the field of semantic similarity measure (e.g. Petrakis et al. [24], S ánchez et al. [25], Rodr íguez and Egenhofer [26], and Jiang et al. [27]) presented to use a mean value of the semantic similarities of a concept pair judged by a certain number of domain experts to replace the actual semantic similarity of this concept pair. The actual semantic similarities of the thirty sample concept pairs are also obtained by this way. Specifically, thirty-six identical questionnaires are distributed to six teachers and thirty students who have ever used PROE and UGNX to do mechanical design and are familiar with the processes of designing different types of hole features with these two systems. These teachers and students are asked to judge the semantic similarity of each pair in Table 1 on a scale 0, 0.1, 0.2, …, 0.9, 1. The thirty-six judgement semantic similarities of each pair are sorted in descending order and the first three and last three ones are removed. Then the mean value of the rest thirty ones is worked out (see Table 2).

The second step is to calculate out the similarities of the semantic descriptions and the similarities of the semantic relationships of the thirty concept pairs. The similarities of the semantic descriptions of the thirty

concept pairs can be respectively calculated out by Tversky's measure (Expression (4)) [23], Petrakis et al.'s measure (Expression (8)) [24], and Sánchez et al.'s measure (Expression (9)) [25]. For example, consider the semantic descriptions of the concepts *PROE-RectangleProfileHole* (in the PROE feature data ontology) and *UGNX-GeneralHole* (in the UGNX feature data ontology) in the introduction. Because both of them are in layer four in their respective ontologies and the following equivalent correspondences are asserted when developing the PROE and UGNX feature data ontologies or inferred after performing reasoning on a combination of these two ontologies:

$PROE\text{-}Hole \equiv UGNX\text{-}Hole$, $proe\text{-}hasPlacement \equiv ugnx\text{-}hasPosition$,

$PROE\text{-}Point \equiv UGNX\text{-}Point$, $proe\text{-}hasSideDepth \equiv ugnx\text{-}hasDepthLimit$,

$PROE\text{-}ThroughAll \equiv UGNX\text{-}ThroughBody$, $proe\text{-}hasDiameter \equiv ugnx\text{-}hasDiameter$,

$PROE\text{-}Lightweight \equiv UGNX\text{-}Subtract$, $proe\text{-}hasLightweight \equiv ugnx\text{-}hasBoolean$,

$proe\text{-}hasName \equiv ugnx\text{-}hasName$, $proe\text{-}hasTolerance \equiv ugnx\text{-}hasTolerance$,

According to the above conditions, the following results are obtained based on Expression (5) and the semantic descriptions of the two concepts in the introduction:

$$\theta(PROE\text{-}RectangleProfileHole, UGNX\text{-}GeneralHole) = 4/(4+4) = 0.5$$

$C_{DPROE\text{-}RectangleProfileHole}$ = {*PROE-Hole*, =1*proe-hasName*, ∃*proe-hasName*.string, =1*proe-hasPlacement*, ∃*proe-hasPlacement*.(*PROE-Point* ⊔ *PROE-Axis* ⊔ *PROE-Surface* ⊔ *PROE-DatumPlane*), =1*proe-hasPlacementType*, ∃*proe-hasPlacementType*.(*PROE-Linear* ⊔ *PROE-Radial* ⊔ *PROE-Diameter*), =1*proe-hasDiameter*, ∃*proe-hasDiameter*.float, =1*proe-hasSideDepth*, ∃*proe-hasSideDepth*.(*PROE-Blind* ⊔ *PROE-Symmetric* ⊔ *PROE-ToNext* ⊔ *PROE-ThroughAll* ⊔ *PROE-ThroughUntil* ⊔ *PROE-ToSelected*), ≤1*proe-hasLightweight*, ∃*proe-hasLightweight*.*PROE-Lightweight*, =1*proe-hasTolerance*, ∃*proe-hasTolerance*.float}

$C_{DUGNX\text{-}GeneralHole}$ = {*UGNX-Hole*, =1*ugnx-hasName*, ∃*ugnx-hasName*.string, =1*ugnx-hasPosition*, ∃*ugnx-hasPosition*.(*UGNX-SketchSection* ⊔ *UGNX-Point*), =1*ugnx-hasHoleDirection*, ∃*ugnx-hasHoleDirection*.(*UGNX-Normal2Face* ⊔ *UGNX-AlongVector*), =1*ugnx-hasForm*, ∃*ugnx-hasForm*.(*UGNX-Simple* ⊔ *UGNX-Counterbored* ⊔ *UGNX-Countersunk* ⊔ *UGNX-Tapered*), =1*ugnx-hasDiameter*, ∃*ugnx-hasDiameter*.(float ⊔ *UGNX-Measure* ⊔ *UGNX-Formula* ⊔ *UGNX-Function* ⊔ *UGNX-Reference* ⊔ *UGNX-Constant*), =1*ugnx-hasDepthLimit*, ∃*ugnx-hasDepthLimit*.(*UGNX-Value* ⊔ *UGNX-UntilSelected* ⊔ *UGNX-UntilNext* ⊔ *UGNX-ThroughBody*), ≤1*ugnx-hasBoolean*, ∃*ugnx-hasBoolean*.*UGNX-Subtract*, =1*ugnx-hasTolerance*, ∃*ugnx-hasTolerance*.float}

$$\left|S(C_{DPROE\text{-}RectangleProfileHole}) \cap S(C_{DUGNX\text{-}GeneralHole})\right| = 13,\ \left|S(C_{DPROE\text{-}RectangleProfileHole}) \cup S(C_{DUGNX\text{-}GeneralHole})\right| = 19$$

$$\left|S(C_{DPROE\text{-}RectangleProfileHole}) \setminus S(C_{DUGNX\text{-}GeneralHole})\right| = 2,\ \left|S(C_{DUGNX\text{-}GeneralHole}) \setminus S(C_{DPROE\text{-}RectangleProfileHole})\right| = 4$$

Then the similarity of the two semantic descriptions is computed by $Sim_{\mathbf{Tversky}}(D_{C1}, D_{C2})$, $Sim_{\mathbf{Petrakis}}(D_{C1}, D_{C2})$, or $Sim_{\mathbf{Sanchez}}(D_{C1}, D_{C2})$:

$$Sim_{\mathbf{Tversky}}(D_{PROE\text{-}RectangleProfileHole}, D_{UGNX\text{-}GeneralHole}) = 13/[13 + 0.5\times 2 + (1-0.5)\times 4] = 0.8125$$

$$Sim_{\mathbf{Petrakis}}(D_{PROE\text{-}RectangleProfileHole}, D_{UGNX\text{-}GeneralHole}) = 13/19 = 0.6842$$

$$Sim_{\mathbf{Sanchez}}(D_{PROE\text{-}RectangleProfileHole}, D_{UGNX\text{-}GeneralHole}) = 1 - \log_2\left[1 + (2+4)/(2+4+13)\right] = 0.6041$$

By a similar way, the similarities of the semantic descriptions (semantic relationships) of the thirty concept pairs are worked out by $Sim_{\mathbf{Tversky}}(D_{C1}, D_{C2})$, $Sim_{\mathbf{Petrakis}}(D_{C1}, D_{C2})$, or $Sim_{\mathbf{Sanchez}}(D_{C1}, D_{C2})$ ($Sim_{\mathbf{Tversky}}(R_{C1}, R_{C2})$, $Sim_{\mathbf{Petrakis}}(R_{C1}, R_{C2})$, or $Sim_{\mathbf{Sanchez}}(R_{C1}, R_{C2})$) (also see Table 2).

Table 2

The mean value of the thirty judgement semantic similarities, the similarities of the semantic descriptions, and the similarities of the semantic relationships of each sample concept pair in Table 1.

Concept pair *CP*	Mean value of 30 judgement results	Similarities of semantic descriptions			Similarities of semantic relationships		
		$Sim_{\mathbf{Tversky}}$	$Sim_{\mathbf{Petrakis}}$	$Sim_{\mathbf{Sanchez}}$	$Sim_{\mathbf{Tversky}}$	$Sim_{\mathbf{Petrakis}}$	$Sim_{\mathbf{Sanchez}}$
$CP_{1,1}$	0.6667	0.8125	0.6842	0.6041	0.8235	0.7000	0.6215
$CP_{1,2}$	0.5033	0.6500	0.4815	0.3973	0.7000	0.5385	0.4525
$CP_{1,3}$	0.4667	0.5909	0.4194	0.3395	0.6087	0.4375	0.3561
$CP_{1,4}$	0.3700	0.4583	0.2973	0.2322	0.4800	0.3158	0.2479
$CP_{1,5}$	0.2100	0.3824	0.2364	0.1814	0.4000	0.2500	0.1926
$CP_{2,1}$	0.7600	0.8824	0.7895	0.7244	0.8889	0.8000	0.7370
$CP_{2,2}$	0.4567	0.6190	0.4483	0.3661	0.6364	0.4667	0.3833
$CP_{2,3}$	0.5300	0.6522	0.4839	0.3996	0.6667	0.5000	0.4150
$CP_{2,4}$	0.3100	0.4400	0.2821	0.2193	0.4615	0.3000	0.2345
$CP_{2,5}$	0.1967	0.4286	0.2727	0.2115	0.4571	0.2963	0.2313
$CP_{3,1}$	0.5200	0.6000	0.4286	0.3479	0.6250	0.4545	0.3720
$CP_{3,2}$	0.4167	0.4737	0.3103	0.2433	0.5263	0.3125	0.2838
$CP_{3,3}$	0.2766	0.4286	0.2727	0.2115	0.4545	0.2941	0.2295
$CP_{3,4}$	0.1900	0.3913	0.2432	0.1871	0.4167	0.2632	0.2035
$CP_{3,5}$	0.0967	0.2571	0.1475	0.1106	0.2941	0.1724	0.1301
$CP_{4,1}$	0.6800	0.7895	0.6522	0.5694	0.8000	0.6667	0.5850
$CP_{4,2}$	0.9267	0.9200	0.8519	0.8007	0.9231	0.8571	0.8074
$CP_{4,3}$	0.6333	0.7600	0.6129	0.5279	0.7692	0.6250	0.5406
$CP_{4,4}$	0.4400	0.4815	0.3171	0.2490	0.5000	0.3333	0.2630
$CP_{4,5}$	0.3600	0.5135	0.3455	0.2736	0.5405	0.3704	0.2955
$CP_{5,1}$	0.7333	0.7500	0.6000	0.5146	0.7619	0.6154	0.5305
$CP_{5,2}$	0.6000	0.7083	0.5484	0.4623	0.7500	0.6000	0.5146
$CP_{5,3}$	0.8400	0.8929	0.8065	0.7447	0.8966	0.8125	0.7521
$CP_{5,4}$	0.3000	0.4643	0.3023	0.2364	0.4828	0.3182	0.2500
$CP_{5,5}$	0.4167	0.5833	0.4118	0.3326	0.6111	0.4400	0.3585
$CP_{6,1}$	0.6033	0.7222	0.5652	0.4792	0.7368	0.5833	0.4975
$CP_{6,2}$	0.4767	0.5909	0.4194	0.3395	0.6364	0.4667	0.3833
$CP_{6,3}$	0.5200	0.7083	0.5484	0.4623	0.7200	0.5625	0.4764
$CP_{6,4}$	0.3333	0.5000	0.3333	0.2630	0.5185	0.3500	0.2775
$CP_{6,5}$	0.2400	0.4474	0.2881	0.2244	0.4615	0.3000	0.2345

The third step is to calculate the nine pairs of weights and the nine groups of the semantic similarities of

the thirty sample concept pairs. The first procedure in the designed weight calculation algorithm is firstly applied to calculate out the nine pairs of weights in $Sim_{i,1}(C_{i,1}, C_{i,2})$, $Sim_{i,2}(C_{i,1}, C_{i,2})$, …, $Sim_{i,9}(C_{i,1}, C_{i,2})$ (see Table 3). According to these weights and the similarities in Table 2, nine groups of the semantic similarities of the thirty sample concept pairs in Table 1 can be respectively worked out by $Sim_{i,1}(C_{i,1}, C_{i,2})$, $Sim_{i,2}(C_{i,1}, C_{i,2})$, …, $Sim_{i,9}(C_{i,1}, C_{i,2})$ (also see Table 3).

Table 3

The worked out nine pairs of weights by the first procedure in the weight calculation algorithm, nine groups of the semantic similarities of the thirty sample concept pairs in Table 1, and nine Pearson correlation coefficients.

w&*CP*&corr	$Sim_{i,1}$	$Sim_{i,2}$	$Sim_{i,3}$	$Sim_{i,4}$	$Sim_{i,5}$	$Sim_{i,6}$	$Sim_{i,7}$	$Sim_{i,8}$	$Sim_{i,9}$
$w_{j,1}$	0.5388	0.8503	0.7170	0.3586	0.8727	1.0000	0.2801	0.0636	0.0000
$w_{j,2}$	0.4612	0.1497	0.2830	0.6414	0.1273	0.0000	0.7199	0.9364	1.0000
$CP_{1,1}$	0.8176	0.7957	0.7584	0.7735	0.6862	0.6842	0.7620	0.6939	0.6215
$CP_{1,2}$	0.6731	0.6333	0.5941	0.6216	0.4888	0.4815	0.6152	0.5295	0.4525
$CP_{1,3}$	0.5991	0.5679	0.5245	0.5408	0.4217	0.4194	0.5333	0.4313	0.3561
$CP_{1,4}$	0.4683	0.4370	0.3988	0.4145	0.2997	0.2973	0.4106	0.3105	0.2479
$CP_{1,5}$	0.3905	0.3626	0.3287	0.3413	0.2381	0.2364	0.3388	0.2456	0.1926
$CP_{2,1}$	0.8854	0.8701	0.8413	0.8533	0.7908	0.7895	0.8428	0.7952	0.7370
$CP_{2,2}$	0.6270	0.5962	0.5523	0.5689	0.4506	0.4483	0.5607	0.4603	0.3833
$CP_{2,3}$	0.6589	0.6294	0.5851	0.6011	0.4859	0.4839	0.5919	0.4936	0.4150
$CP_{2,4}$	0.4499	0.4190	0.3818	0.3972	0.2844	0.2821	0.3937	0.2949	0.2345
$CP_{2,5}$	0.4417	0.4088	0.3728	0.3910	0.2757	0.2727	0.3883	0.2909	0.2313
$CP_{3,1}$	0.6115	0.5782	0.5355	0.5546	0.4319	0.4286	0.5474	0.4477	0.3720
$CP_{3,2}$	0.4980	0.4496	0.4200	0.4488	0.3106	0.3103	0.4470	0.3081	0.2838
$CP_{3,3}$	0.4405	0.4085	0.3723	0.3893	0.2754	0.2727	0.3864	0.2888	0.2295
$CP_{3,4}$	0.4030	0.3721	0.3382	0.3545	0.2457	0.2432	0.3524	0.2584	0.2035
$CP_{3,5}$	0.2742	0.2444	0.2212	0.2415	0.1507	0.1475	0.2427	0.1685	0.1301
$CP_{4,1}$	0.7943	0.7711	0.7316	0.7470	0.6540	0.6522	0.7354	0.6605	0.5850
$CP_{4,2}$	0.9214	0.9106	0.8881	0.8976	0.8526	0.8519	0.8888	0.8535	0.8074
$CP_{4,3}$	0.7642	0.7398	0.6979	0.7132	0.6144	0.6129	0.7016	0.6188	0.5406
$CP_{4,4}$	0.4900	0.4593	0.4197	0.4344	0.3192	0.3171	0.4297	0.3279	0.2630
$CP_{4,5}$	0.5260	0.4921	0.4518	0.4706	0.3487	0.3455	0.4657	0.3642	0.2955
$CP_{5,1}$	0.7555	0.7299	0.6879	0.7038	0.6020	0.6000	0.6926	0.6090	0.5305
$CP_{5,2}$	0.7275	0.6921	0.6535	0.6777	0.5550	0.5484	0.6694	0.5912	0.5146
$CP_{5,3}$	0.8946	0.8809	0.8531	0.8643	0.8073	0.8065	0.8541	0.8082	0.7521
$CP_{5,4}$	0.4728	0.4424	0.4037	0.4181	0.3043	0.3023	0.4138	0.3130	0.2500
$CP_{5,5}$	0.5961	0.5618	0.5197	0.5396	0.4154	0.4118	0.5331	0.4332	0.3585
$CP_{6,1}$	0.7289	0.7014	0.6586	0.6753	0.5675	0.5652	0.6646	0.5767	0.4975
$CP_{6,2}$	0.6119	0.5723	0.5321	0.5586	0.4254	0.4194	0.5532	0.4586	0.3833
$CP_{6,3}$	0.7137	0.6865	0.6427	0.6585	0.5502	0.5484	0.6478	0.5561	0.4764
$CP_{6,4}$	0.5085	0.4775	0.4370	0.4521	0.3354	0.3333	0.4469	0.3445	0.2775
$CP_{6,5}$	0.4539	0.4253	0.3871	0.3993	0.2896	0.2881	0.3951	0.2952	0.2345
corr	0.9717	0.9714	0.9725	0.9729	0.9682	0.9681	0.9732	0.9665	0.9651

The last step is to select a measure with high similarity calculation accuracy. According to the calculated

nine groups of the semantic similarities in Table 3, the Pearson correlation coefficients of $Sim_{i,1}(C_{i,1}, C_{i,2})$, $Sim_{i,2}(C_{i,1}, C_{i,2})$, …, $Sim_{i,9}(C_{i,1}, C_{i,2})$ are respectively computed out and listed in Table 3. As can be seen from the last row of this table, the difference between each two of the nine correlation coefficients is less than 0.1. So the second procedure in the weight calculation algorithm is applied to calculate out the nine pairs of weights in $Sim_{i,1}(C_{i,1}, C_{i,2})$, $Sim_{i,2}(C_{i,1}, C_{i,2})$, …, $Sim_{i,9}(C_{i,1}, C_{i,2})$ (see Table 4). Based on these weights and the similarities in Table 2, nine groups of the semantic similarities of the thirty sample concept pairs in Table 1 are respectively worked out by $Sim_{i,1}(C_{i,1}, C_{i,2})$, $Sim_{i,2}(C_{i,1}, C_{i,2})$, …, $Sim_{i,9}(C_{i,1}, C_{i,2})$ (also see Table 4). Then the residual sums of squares of $Sim_{i,1}(C_{i,1}, C_{i,2})$, $Sim_{i,2}(C_{i,1}, C_{i,2})$, …, $Sim_{i,9}(C_{i,1}, C_{i,2})$ are respectively computed out and listed in Table 4. As can be seen from the last row of this table, the residual sum of squares of $Sim_{i,4}(C_{i,1}, C_{i,2})$ is the least. Thus the measure with high similarity calculation accuracy for concepts in the PROE and UGNX feature data ontologies is selected as $Sim(C_1, C_2) = 0.8666Sim_{\mathbf{Petrakis}}(D_{C1}, D_{C2}) + 0.1334Sim_{\mathbf{Tversky}}(R_{C1}, R_{C2})$. This measure can be directly used to compute the remainder possible concepts pairs whose two components are respectively from the PROE feature data ontology and the UGNX feature data ontology.

Table 4

The worked out nine pairs of weights by the second procedure in the weight calculation algorithm, nine groups of the semantic similarities of the thirty sample concept pairs in Table 1, and nine residual sums of squares.

w&*CP*&rss	$Sim_{i,1}$	$Sim_{i,2}$	$Sim_{i,3}$	$Sim_{i,4}$	$Sim_{i,5}$	$Sim_{i,6}$	$Sim_{i,7}$	$Sim_{i,8}$	$Sim_{i,9}$
$w_{j,1}$	1.0000	0.0294	0.3706	0.8666	0.1957	1.0000	0.6112	0.0000	0.0000
$w_{j,2}$	0.0000	0.9706	0.6294	0.1334	0.8043	0.0000	0.3888	1.0000	1.0000
$CP_{1,1}$	0.8125	0.7033	0.6923	0.7028	0.6969	0.6842	0.6894	0.7000	0.6215
$CP_{1,2}$	0.6500	0.5418	0.5257	0.5106	0.5273	0.4815	0.5150	0.5385	0.4525
$CP_{1,3}$	0.5909	0.4420	0.4431	0.4447	0.4340	0.4194	0.4442	0.4375	0.3561
$CP_{1,4}$	0.4583	0.3200	0.3259	0.3217	0.3122	0.2973	0.3285	0.3158	0.2479
$CP_{1,5}$	0.3824	0.2539	0.2629	0.2582	0.2473	0.2364	0.2664	0.2500	0.1926
$CP_{2,1}$	0.8824	0.8024	0.7909	0.8028	0.7979	0.7895	0.7884	0.8000	0.7370
$CP_{2,2}$	0.6190	0.4712	0.4707	0.4734	0.4631	0.4483	0.4712	0.4667	0.3833
$CP_{2,3}$	0.6522	0.5045	0.5029	0.5083	0.4968	0.4839	0.5034	0.5000	0.4150
$CP_{2,4}$	0.4400	0.3041	0.3107	0.3060	0.2965	0.2821	0.3135	0.3000	0.2345
$CP_{2,5}$	0.4286	0.3002	0.3044	0.2973	0.2917	0.2727	0.3070	0.2963	0.2313
$CP_{3,1}$	0.6000	0.4588	0.4565	0.4548	0.4494	0.4286	0.4556	0.4545	0.3720
$CP_{3,2}$	0.4737	0.3172	0.3542	0.3391	0.3121	0.3103	0.3533	0.3125	0.2838
$CP_{3,3}$	0.4286	0.2981	0.3033	0.2970	0.2899	0.2727	0.3060	0.2941	0.2295
$CP_{3,4}$	0.3913	0.2670	0.2731	0.2663	0.2593	0.2432	0.2764	0.2632	0.2035
$CP_{3,5}$	0.2571	0.1749	0.1772	0.1671	0.1675	0.1475	0.1819	0.1724	0.1301
$CP_{4,1}$	0.7895	0.6703	0.6608	0.6719	0.6639	0.6522	0.6591	0.6667	0.5850
$CP_{4,2}$	0.9200	0.8589	0.8491	0.8614	0.8561	0.8519	0.8483	0.8571	0.8074
$CP_{4,3}$	0.7600	0.6290	0.6219	0.6338	0.6226	0.6129	0.6217	0.6250	0.5406
$CP_{4,4}$	0.4815	0.3377	0.3440	0.3415	0.3301	0.3171	0.3466	0.3333	0.2630
$CP_{4,5}$	0.5135	0.3746	0.3763	0.3715	0.3655	0.3455	0.3774	0.3704	0.2955
$CP_{5,1}$	0.7500	0.6194	0.6118	0.6216	0.6124	0.6000	0.6108	0.6154	0.5305
$CP_{5,2}$	0.7083	0.6032	0.5864	0.5753	0.5899	0.5484	0.5742	0.6000	0.5146
$CP_{5,3}$	0.8929	0.8149	0.8043	0.8185	0.8113	0.8065	0.8038	0.8125	0.7521

$CP_{5,4}$	0.4643	0.3225	0.3294	0.3264	0.3151	0.3023	0.3322	0.3182	0.2500
$CP_{5,5}$	0.5833	0.4442	0.4418	0.4384	0.4345	0.4118	0.4409	0.4400	0.3585
$CP_{6,1}$	0.7222	0.5874	0.5808	0.5881	0.5798	0.5652	0.5794	0.5833	0.4975
$CP_{6,2}$	0.5909	0.4704	0.4602	0.4483	0.4574	0.4194	0.4549	0.4667	0.3833
$CP_{6,3}$	0.7083	0.5668	0.5623	0.5713	0.5597	0.5484	0.5625	0.5625	0.4764
$CP_{6,4}$	0.5000	0.3544	0.3600	0.3580	0.3467	0.3333	0.3623	0.3500	0.2775
$CP_{6,5}$	0.4474	0.3043	0.3134	0.3112	0.2977	0.2881	0.3166	0.3000	0.2345
rss	0.5780	0.0847	0.0844	**0.0789**	0.0846	0.0953	0.0876	0.0852	0.2555

Here is an example of the application of the selected measure. In this example, the selected measure is used to transfer a PROE part including four identical standard profile holes (see Fig. 1) to UGNX. This transferring process mainly contains four steps (see Fig. 2).

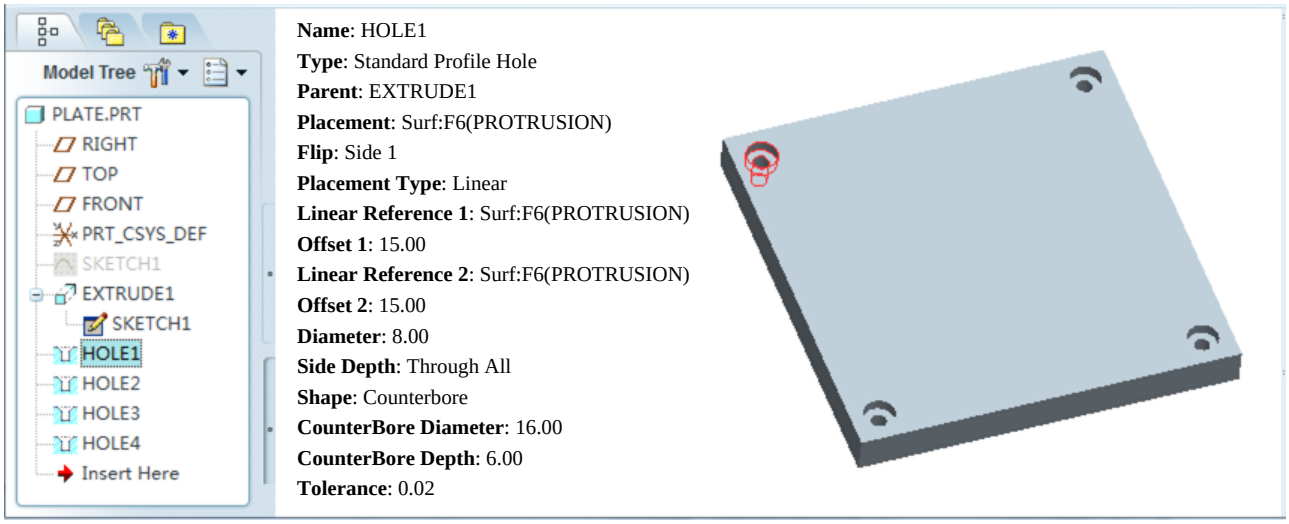

Fig. 1. A part including four identical standard profile holes in PROE.

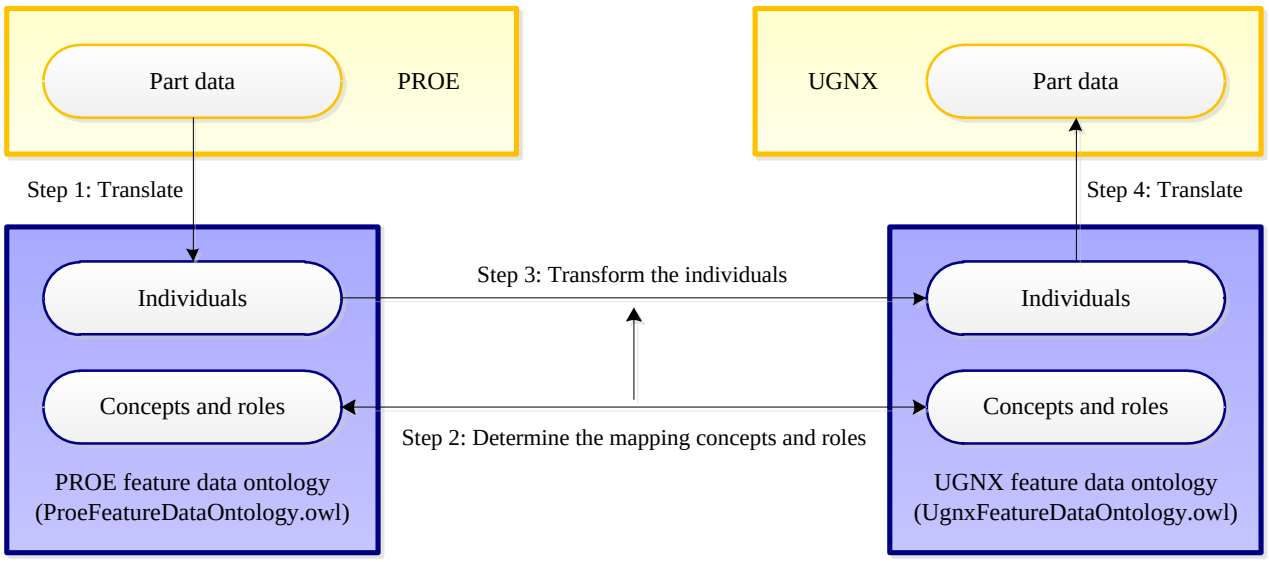

Fig. 2. Part data transferring process based on Semantic Web technologies.

The first step is to instantiate the PROE feature data ontology. Using PROE J-Link API and Protégé-

g é-OWL API, the PROE feature data ontology is instantiated by the designed part in Fig. 1. For example, the instantiation of the feature HOLE1 in the PROE feature data ontology is shown in Fig. 3.

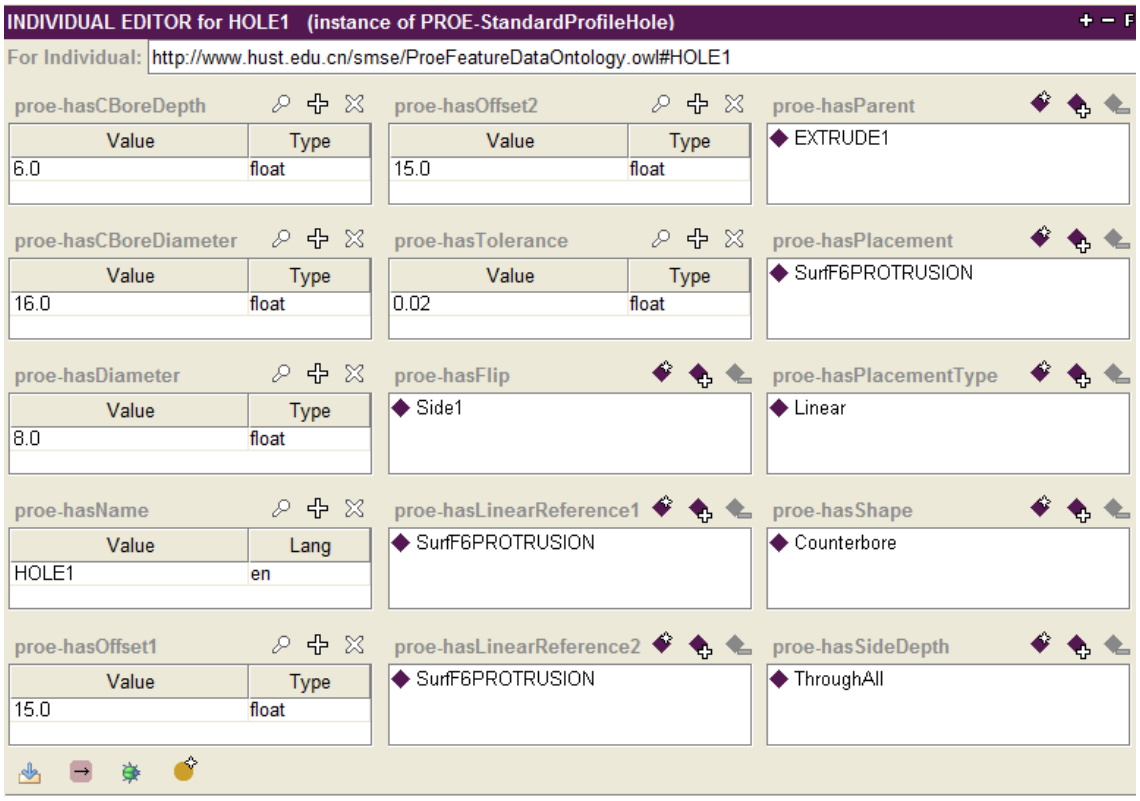

Fig. 3. Instantiation of the feature HOLE1 in the PROE feature data ontology.

The second step is to determine the mapping concepts and roles between the PROE feature data ontology and the UGNX feature data ontology. Using the technologies of rule reasoning and semantic similarity assessment, the mapping concepts and roles between the two ontologies are determined. For instance, using the selected measure $Sim(C_1, C_2) = 0.8666Sim_{\mathbf{Petrakis}}(D_{C1}, D_{C2}) + 0.1334Sim_{\mathbf{Tversky}}(R_{C1}, R_{C2})$, the semantic similarities of the concept pairs $CP_{2,1}$, $CP_{2,2}$, $CP_{2,3}$, $CP_{2,4}$, and $CP_{2,5}$ in Table 1 are respectively computed out as 0.8028, 0.4734, 0.5083, 0.3060, and 0.2973. Thus the mapping concept from the UGNX feature data ontology for *PROE-StandardProfileHole* is determined as *UGNX-GeneralHole* according to these similarity scores.

The third step is to transform the individuals of the PROE feature data ontology to the individuals of the UGNX feature data ontology. Based on the semantic descriptions of the mapping concept and role pairs between the PROE feature data ontology and the UGNX feature data ontology (including the mapping concept pair (*PROE-StandardProfileHole*, *UGNX-GeneralHole*)), the individuals of the PROE feature data ontology are transformed to the individuals of the UGNX feature data ontology using Prot ég é-OWL API. For instance, the transformed individuals related to the feature HOLE1 is shown in Fig. 4.

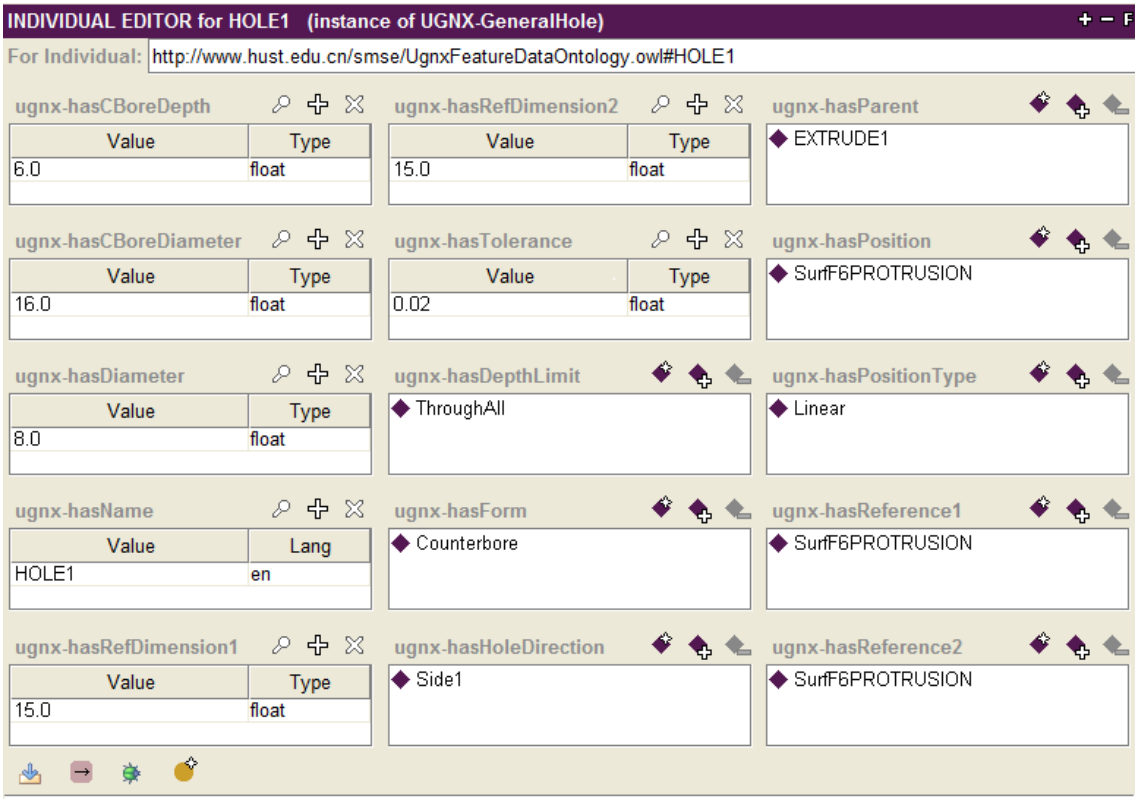

Fig. 4. Transformed individuals related to the feature HOLE1.

The last step is to transfer the individuals of the UGNX feature data ontology to UGNX. Using Protégé-OWL API and NX Open API, the individuals of the UGNX feature data ontology are transferred to UGNX so that the designed part in PROE is successfully transferred to UGNX. The transferred part data, as shown in Fig. 5, not only contains geometry, but also contains design history, parameters, and features. One can directly carry out the modification, extension, and other higher-level operations on the part in UGNX.

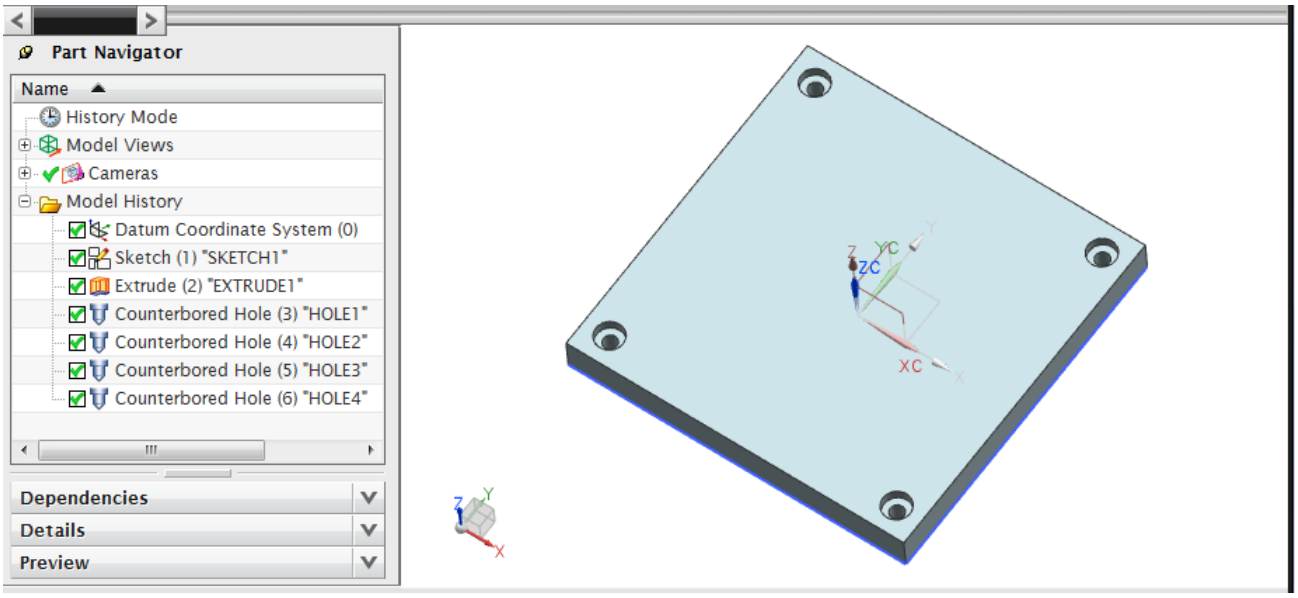

Fig. 5. Transferred part data in UGNX.

4.3. Evaluation

4.3.1 Theoretical comparison

The theoretical comparison in Appendix C proves that the similarity calculation accuracy of the measure selected by the designed measure selection algorithm is the highest among the accuracies of all possible linear combinations of any one of $Sim_{\mathbf{Tversky}}(D_{C1}, D_{C2})$, $Sim_{\mathbf{Petrakis}}(D_{C1}, D_{C2})$, and $Sim_{\mathbf{Sanchez}}(D_{C1}, D_{C2})$ and any one of $Sim_{\mathbf{Tversky}}(R_{C1}, R_{C2})$, $Sim_{\mathbf{Petrakis}}(R_{C1}, R_{C2})$, and $Sim_{\mathbf{Sanchez}}(R_{C1}, R_{C2})$. This theoretical comparison also proves that the accuracy of each of the nine measures is higher than the accuracies of all possible linear combinations of its two contribution components.

For the measures in the existing analogous methods [**16**-**19**], the measure in Patil's method [**16**] is actually the measure $Sim_{\mathbf{Tversky}}(D_{C1}, D_{C2})$, the measure $Sim_{\text{definition}}(C_1, C_2)$ (the measure $Sim_{\text{name}}(C_1, C_2)$ is not considered since it belongs to a kind of syntax similarities and the present paper only discusses semantic similarities) in Lee et al.'s method [**17**] can be considered as $Sim_{\mathbf{Petrakis}}(D_{C1}, D_{C2})$, the measure in Zhan et al.'s method [**18**] is in fact the measure $Sim_{\mathbf{Petrakis}}(R_{C1}, R_{C2})$, and the measure in Abdul-Ghafour et al.'s method [**19**] is the measure $w_1 Sim_{\mathbf{Tversky}}(D_{C1}, D_{C2}) + w_2 Sim_{\mathbf{Tversky}}(R_{C1}, R_{C2})$. As can be seen from the theoretical comparison, the similarity calculation accuracy of the measure selected by the measure selection algorithm is higher than or equal to the accuracies of these measures.

4.3.2 Experimental comparison

In general, an experimental comparison for evaluating the similarity calculation accuracies of measures can be made using standard benchmarks consisting of a certain number of term pairs whose similarities are judged by a group of domain experts. The most widely used three standard benchmarks in such evaluations are Rubenstein and Goodenough's benchmark [**36**], Miller and Charles' benchmark [**37**], and the WordSimilarity-353 Test Collection [**38**]. These three standard benchmarks cannot be used to evaluate the similarity calculation accuracies of the measures for concepts in two different CAD model data ontologies since most of the terms in them are not the terms in CAD modeling domain. A new benchmark consisting of CAD modeling domain terms is required to be designed.

In the previous example, thirty sample concept pairs whose two components are respectively from the PROE and UGNX feature data ontologies have been extracted (see Table 1). The mean value of the semantic similarities of each pair judged by thirty domain experts has been worked out (see Table 2). These mean values have been used to evaluate the similarity calculation accuracies of the nine derived measures in the proposed measure selection method (see Table 3 and Table 4). Here the thirty sample concept pairs and their semantic similarities judged by domain experts can be directly used as a benchmark to evaluate and compare the similarity calculation accuracies of the selected measure in the proposed measure selection method and the measures in the existing analogous methods [**16**-**19**].

The measure with high similarity calculation accuracy for concepts in the PROE and UGNX feature data ontologies is selected as $Sim(C_1, C_2) = 0.8666 Sim_{\mathbf{Petrakis}}(D_{C1}, D_{C2}) + 0.1334 Sim_{\mathbf{Tversky}}(R_{C1}, R_{C2})$. This measure simultaneously takes the semantic descriptions and relationships of concepts as attributes. The semantic similarities of the thirty sample concept pairs have been calculated by this measure and the similarity calculation accuracy of the measure has been evaluated as 0.9698 and 0.0789 (see Table 5). Thus, to make an evaluation and comparison between the similarity calculation accuracies of the selected measure and the measures in the

existing analogous methods [16-19], the semantic similarities of the thirty sample concept pairs are required to be respectively calculated by the measures in [16-19] and then the similarity calculation accuracy of each measure can be quantified.

For the measure in [16] (Expression (12)), the attribute of the semantic descriptions of concepts is considered and u and v are respectively assigned as 0.75 and 0.25 (just like the values assigned by Patil) when using this measure to compute the semantic similarities of the thirty sample concept pairs. After calculating the semantic similarities, the Pearson correlation coefficient and the residual sum of squares between the calculated semantic similarities and the judgement semantic similarities are worked out (see Table 5).

For the measure in [17] (Expression (13)), this paper only uses $Sim_{\text{definition}}(C_1, C_2)$ to calculate the semantic similarities of the thirty sample concept pairs for the purpose of enabling fair comparison since $Sim_{\text{name}}(C_1, C_2)$ belongs to a kind of syntax similarities and the paper only discusses semantic similarities. This calculation also considers the attribute of semantic descriptions. The Pearson correlation coefficient and the residual sum of squares are respectively calculated after computing the semantic similarities (see Table 5).

Through comparing the measure in [18] (Expression (14)) and the measure $Sim_{\mathbf{Petrakis}}(R_{C1}, R_{C2})$, it can be seen that these two measures are all originated from Petrakis et al.'s measure [24]. Although the measure in [18] only takes the property-of, part-of, and is-a relationships as its attributes when calculating concept similarities, all types of semantic relationships are considered here for the purpose of enabling fair comparison. Thus the semantic similarities of the thirty sample concept pairs calculated by the measure in [18] are respectively equal to those calculated by $Sim_{\mathbf{Petrakis}}(R_{C1}, R_{C2})$. Based on these semantic similarities, the Pearson correlation coefficient and the residual sum of squares are respectively computed (see Table 5).

For the measure in [19] (Expression (16)), the attributes of the semantic descriptions and relationships of concepts are considered, w_1 and w_2 are all assigned as 0.5, and $Sim_{\mathbf{Tversky}}(D_{C1}, D_{C2})$ and $Sim_{\mathbf{Tversky}}(R_{C1}, R_{C2})$ are respectively applied to calculate out $Sim_{\mathbf{des}}(D_{C1}, D_{C2})$ and $Sim_{\mathbf{rel}}(R_{C1}, R_{C2})$ when using this measure to compute the semantic similarities of the thirty sample concept pairs. After calculating the semantic similarities, the Pearson correlation coefficient and the residual sum of squares are respectively calculated out (see Table 5).

Table 5

The semantic similarities of the thirty sample concept pairs in Table 1 calculated by the measures in the proposed measure selection method and four existing analogous methods and the Pearson correlation coefficient and the residual sum of squares of the measure in each method.

CP&corr&rss	The measure in the proposed method	The measure in Patil's method [**16**]	The measure in Lee et al.'s method [**17**]	The measure in Zhan et al.'s method [**18**]	The measure in Abdul-Ghafour et al.'s method [**19**]
$CP_{1,1}$	0.7028	0.8387	0.8235	0.7000	0.8180
$CP_{1,2}$	0.5106	0.7429	0.7000	0.5385	0.6750
$CP_{1,3}$	0.4447	0.7027	0.6087	0.4375	0.5998
$CP_{1,4}$	0.3217	0.5641	0.4800	0.3158	0.4691
$CP_{1,5}$	0.2582	0.5306	0.5000	0.2500	0.3912
$CP_{2,1}$	0.8028	0.8824	0.8889	0.8000	0.8857
$CP_{2,2}$	0.4734	0.6842	0.6667	0.4667	0.6277
$CP_{2,3}$	0.5083	0.7500	0.6667	0.5000	0.6594
$CP_{2,4}$	0.3060	0.5238	0.4615	0.3000	0.4507

$CP_{2,5}$	0.2973	0.5769	0.5517	0.2963	0.4428
$CP_{3,1}$	0.4548	0.6429	0.6250	0.4545	0.6125
$CP_{3,2}$	0.3391	0.5625	0.5263	0.3125	0.5000
$CP_{3,3}$	0.2970	0.5294	0.4545	0.2941	0.4416
$CP_{3,4}$	0.2663	0.5000	0.6667	0.2632	0.4040
$CP_{3,5}$	0.1671	0.3750	0.3704	0.1724	0.2756
$CP_{4,1}$	0.6719	0.7500	0.8000	0.6667	0.7948
$CP_{4,2}$	0.8614	0.9200	0.9231	0.8571	0.9216
$CP_{4,3}$	0.6338	0.8261	0.7692	0.6250	0.7646
$CP_{4,4}$	0.3415	0.5417	0.5000	0.3333	0.4908
$CP_{4,5}$	0.3715	0.6552	0.6452	0.3704	0.5270
$CP_{5,1}$	0.6216	0.6977	0.7619	0.6154	0.7560
$CP_{5,2}$	0.5753	0.7234	0.7500	0.6000	0.7291
$CP_{5,3}$	0.8185	0.9091	0.8966	0.8125	0.8947
$CP_{5,4}$	0.3264	0.5098	0.4828	0.3182	0.4736
$CP_{5,5}$	0.4384	0.7119	0.6875	0.4400	0.5972
$CP_{6,1}$	0.5881	0.7027	0.7368	0.5833	0.7295
$CP_{6,2}$	0.4483	0.6341	0.6364	0.4667	0.6137
$CP_{6,3}$	0.5713	0.7907	0.7200	0.5625	0.7142
$CP_{6,4}$	0.3580	0.5778	0.5185	0.3500	0.5092
$CP_{6,5}$	0.3112	0.5965	0.6000	0.3000	0.4545
corr	0.9698	0.9012	0.8787	0.9665	**0.9717**
rss	**0.0789**	1.4492	1.2632	0.0852	0.6686

As can be seen from Table 5, the measures in Abdul-Ghafour et al.'s method [**19**], the proposed measure selection method, and Zhan et al.'s method [**18**], obtain relatively large Pearson correlation coefficient, which indicates that these three measures correlate with domain expert judgement in a relatively high degree. Correspondingly, the Pearson correlation coefficient of the measure in Patil's method [**16**] is less than the coefficient of the three measures and the Pearson correlation coefficient of the measure in Lee et al.'s method [**17**] is the minimum one. This may be due to the fact that the character similarity between concept names was not considered in the measure or the measure for the similarities between concept definitions which was used to compute the coefficient cannot correlate with domain expert judgement in a high degree.

As can also be seen from Table 5, the measure in the proposed method obtains the minimum residual sum of squares 0.0789. This corresponds to the theoretical comparison result and shows that most of the semantic similarities of the thirty sample concept pairs computed out by this measure are close to their counterparts judged by domain experts. The residual sum of squares of the measure in Zhan et al.'s method [**18**] is 0.0852. It is slightly greater than the residual sum of squares of the measure in the proposed method. This indicates that the accuracies of these two measures are close for the thirty sample concept pairs. The residual sums of squares of the measures in Patil's method [**16**], Lee et al.'s method [**17**], and Abdul-Ghafour et al.'s method [**19**] are obvious greater than the residual sums of squares of the measures in the proposed method and Zhan et al.'s method [**18**]. This signifies that a certain number of the semantic similarities of the thirty sample concept pairs calculated out by these measures are not close to their judgement counterparts.

Since the measure in the proposed method achieves the minimum residual sum of squares and there is slight difference between the Pearson correlation coefficients of it and the measures in Abdul-Ghafour et al.'s method [19] and Zhan et al.'s method [18], it can be concluded that the measure selected by the proposed method can provide high similarity calculation accuracy for the thirty sample concept pairs in the benchmark. The measure in Zhan et al.'s method [18] also has high similarity calculation accuracy for the thirty sample concept pairs. By contrast, the similarity calculation accuracies of the measures in Patil's method [16], Lee et al.'s method [17], and Abdul-Ghafour et al.'s method [19] are apparently lower than the similarity calculation accuracies of the measures in the proposed method and Zhan et al.'s method [18].

In summary, the measure selected by the proposed method can offer high similarity calculation accuracy for concepts in two different CAD model data ontologies. Even though the measure in Zhan et al.'s method [18] also has high similarity calculation accuracy, this is just for the thirty sample concept pairs in a specific benchmark. If the sample concept pairs are altered, the accuracy of the measure may not be high because the measure cannot be adjusted in accordance with different sample concept pairs. Rather, the measure selected by the proposed method can be adjusted (through determining two appropriate weights and a measure with high similarity calculation accuracy) with the changed sample concept pairs. So in this respect, the measure selected by the proposed method is better than the measure in Zhan et al.'s method [18].

5. Conclusions

In this paper, a method for selecting a semantic similarity measure with high similarity calculation accuracy for concepts in two different CAD model data ontologies has been proposed. This method mainly consists of two parts: A weight calculation algorithm and a measure selection algorithm. The weight calculation algorithm calculates the weights in a measure according to a certain amount of sample data but not according to human assignment, which provides an effective way to improve the similarity calculation accuracy of the measure. The measure selection algorithm is capable of choosing different measures with high similarity calculation accuracies for different CAD model data ontologies. To the best of knowledge, this is the first consideration of the similarity calculation accuracy of the measures for concepts in CAD model data ontologies. The paper also describes the implementation, illustration, and evaluation of the proposed method. The evaluation result shows that the measure selected by the proposed method has good human correlation and high similarity calculation accuracy.

In the future, the authors of the paper aim especially at solving the following two limitations of the proposed measure selection method: (1) The method does not take into account the similarities of the syntaxes of compared concepts. The syntax similarities of two compared concepts also make a contribution to their overall similarities. A comprehensive mapping between two different CAD model data ontologies should consider syntax similarities. Thus a future work is to extend the method by considering the similarities of the syntaxes (e.g. names, annotations, comments) of compared concepts. (2) The method is insufficient for 1:*n*, *m*:1, and *m*:*n* mappings between concepts. In actual CAD model data exchange, the mappings between two CAD systems can be 1:1, 1:*n*, *m*:1, and *m*:*n*. The method can only deal with 1:1 mapping. This will inevitably result in the loss of some model data if it is used to exchange CAD model data. So another future work is to study how to implement 1:*n*, *m*:1, and *m*:*n* mappings between CAD model data ontologies.

Acknowledgements

The authors would like to appreciate the insightful comments from the two anonymous reviewers for the improvement of the paper. The authors also would like to acknowledge the financial supports by the National Natural Science Foundation of China (No. 51475190), the National Basic Research Program of China (No. 2014CB046705), the Hubei Provincial Natural Science Foundation of China (No. 2015CFA109), the Innovation Foundation of Graduate Innovation and Entrepreneurship Base of Huazhong University of Science and Technology (No. 2015650011), the Doctoral Dissertation Innovation Foundation of Huazhong University of Science and Technology, and the National Scholarship of China Scholarship Council.

Disclaimer

Certain commercial software products are mentioned in this paper. These products were used only for citation and demonstration purposes. This use does not imply the approval or endorsement by our institutions, nor does it imply that these products are necessarily the best available for the purpose.

Appendix A. Description of the weight calculation algorithm

Weight calculation algorithm

Function: Calculate the weights of the contribution components in a semantic similarity measure

Composition: A procedure of computing weights by maximizing Pearson correlation coefficient (corr);
A procedure of computing weights by minimizing residual sum of squares (rss)

procedure ComputingWeightsByMaximizingCorr

Input: A positive integer n that stands for the number of contribution components;
n sets of N (N is the number of sample concept pairs) values of the n contribution components $\{f_{i,1}(C_{i,1}, C_{i,2})\}$, $\{f_{i,2}(C_{i,1}, C_{i,2})\}$,…, $\{f_{i,n}(C_{i,1}, C_{i,2})\}$ ($i = 1, 2,\ldots, N$);
A set of the actual semantic similarities of the N sample concept pairs $\{A_i(C_{i,1}, C_{i,2})\}$ ($i = 1, 2,\ldots, N$)

Output: n weights of the n contribution components $w_1, w_2, \ldots, w_n$ that can maximize corr

Step 1. **for integer** $i \leftarrow 1$ **to** N **do**
Calculate out the values of all of the elements of $\sum_{UU}$, $\sum_{UV}$, $\sum_{VV}$, and $\sum_{VU}$
Calculate out the matrix $\sum_{UU}^{-1} \sum_{UV} \sum_{VV}^{-1} \sum_{VU}$

Step 2. Seek the eigenvector with the maximum eigenvalue for $\sum_{UU}^{-1} \sum_{UV} \sum_{VV}^{-1} \sum_{VU}$ ($[w_1^{\mathbf{o}}, w_2^{\mathbf{o}}, \ldots, w_n^{\mathbf{o}}]^{\mathrm{T}}$)

Step 3. **for integer** $j \leftarrow 1$ **to** n **do**
if $w_j^{\mathbf{o}} < 0$, **then** $w_j^{\mathbf{o}} \leftarrow 0$ and $w_j \leftarrow 0$
for integer $j \leftarrow 1$ **to** n **do**
$w_j \leftarrow w_j^{\mathbf{o}} / (w_1^{\mathbf{o}} + w_2^{\mathbf{o}} + \ldots + w_n^{\mathbf{o}})$

Step 4. Output the n weights $w_1, w_2, \ldots, w_n$

end ComputingWeightsByMaximizingCorr

procedure ComputingWeightsByMinimizingRss

Input: A positive integer n that stands for the number of contribution components;
n sets of N (N is the number of sample concept pairs) values of the n contribution components

$\{f_{i,1}(C_{i,1}, C_{i,2})\}$, $\{f_{i,2}(C_{i,1}, C_{i,2})\}$,…, $\{f_{i,n}(C_{i,1}, C_{i,2})\}$ (i = 1, 2,…, N);

A set of the actual semantic similarities of the N sample concept pairs $\{A_i(C_{i,1}, C_{i,2})\}$ (i = 1, 2,…, N)

Output: n weights of the n contribution components $w_1, w_2, \ldots, w_n$ that can minimize rss

Step 1. **for integer** $i \leftarrow 1$ **to** N **do**

Calculate out the values of all of the elements of $\boldsymbol{A}$ and $\boldsymbol{b}$ in $\boldsymbol{Aw'} = \boldsymbol{b}$

Step 2. Apply the Gauss elimination method to solve the vector $\boldsymbol{w'} = [w_1^{\mathbf{o}}, w_2^{\mathbf{o}}, \ldots, w_{n-1}^{\mathbf{o}}]^{\mathrm{T}}$

Step 3. **for integer** $j \leftarrow 1$ **to** n–1 **do**

if $w_j^{\mathbf{o}} > 1$, **then** $w_j \leftarrow 1$ and $w_k \leftarrow 0$ (k = 1, 2, …, n and $k \neq j$) **return**

else if $0 \leq w_r^{\mathbf{o}} \leq 1$, **then** $w_j \leftarrow w_j^{\mathbf{o}}$

else, **then** $w_j \leftarrow 0$

$w_n \leftarrow 1 - w_1 - w_2 - \ldots - w_{n-1}$

Step 4. Output the n weights $w_1, w_2, \ldots, w_n$

end ComputingWeightsByMinimizingRss

Appendix B. Description of the measure selection algorithm

Measure selection algorithm

Function: Select a measure with high accuracy for concepts in two different CAD model data ontologies

Composition: A procedure of selecting a semantic similarity measure with high accuracy

Input: Two different CAD model data ontologies O_1 and O_2 that are represented in OWL DL

Output: A semantic similarity measure with high similarity calculation accuracy for concepts in O_1 and O_2

procedure SelectAMeasureWithHighAccuracy

Step 1. Perform reasoning on a combination of O_1 and O_2 to find out semantically equivalent pairs

Extract N_1 sample concepts that do not have semantically equivalent concepts from O_1

Extract N_2 sample concepts that do not have semantically equivalent concepts from O_2

Construct $N = N_1N_2$ sample concept pairs $(C_{1,1}, C_{1,2})$, $(C_{2,1}, C_{2,2})$, …, $(C_{N,1}, C_{N,2})$

Judge the semantic similarities of $(C_{i,1}, C_{i,2})$ (i = 1, 2, …, N) and take them as $A_i(C_{i,1}, C_{i,2})$

Step 2. **for integer** $i \leftarrow 1$ **to** N **do**

Use Expression (4), Expression (8), and Expression (9) to calculate the semantic similarities of the semantic descriptions and the semantic relationships of $(C_{i,1}, C_{i,2})$

Step 3. **for integer** $j \leftarrow 1$ **to** 9 **do**

Use ComputingWeightsByMaximizingCorr to compute $w_{j,1}$ and $w_{j,2}$ in $Sim_j(C_1, C_2)$

Step 4. **for integer** $j \leftarrow 1$ **to** 9 **do**

for integer $i \leftarrow 1$ **to** N **do**

Use Expression (28) to calculate $Sim_{i,j}(C_{i,1}, C_{i,2})$

Step 5. **for integer** $j \leftarrow 1$ **to** 9 **do**

Use Expression (29) to calculate $\mathrm{corr}([Sim_{i,j}(C_{i,1}, C_{i,2})]^{\mathrm{T}}, [A_i(C_{i,1}, C_{i,2})]^{\mathrm{T}})$

Step 6. **for integer** $j \leftarrow 1$ **to** 9 **do**

if the difference between some two of $\mathrm{corr}([Sim_{i,j}(C_{i,1}, C_{i,2})]^{\mathrm{T}}, [A_i(C_{i,1}, C_{i,2})]^{\mathrm{T}}) \geq 0.1$

then Find out $Sim_p(C_1, C_2)$ where corr($[Sim_{i,p}(C_{i,1}, C_{i,2})]^T$, $[A_i(C_{i,1}, C_{i,2})]^T$) is the greatest one among corr($[Sim_{i,j}(C_{i,1}, C_{i,2})]^T$, $[A_i(C_{i,1}, C_{i,2})]^T$) and output it

else, **then for integer** $j \leftarrow 1$ **to** 9 **do**

Use ComputingWeightsByMinimizingRss to compute $w_{j,1}$ and $w_{j,2}$ in $Sim_j(C_1, C_2)$

for integer $i \leftarrow 1$ **to** N **do**

Use Expression (28) to calculate $Sim_{i,j}(C_{i,1}, C_{i,2})$

Use Expression (30) to calculate rss($[Sim_{i,j}(C_{i,1}, C_{i,2})]^T$, $[A_i(C_{i,1}, C_{i,2})]^T$)

Find out $Sim_q(C_1, C_2)$ where rss($[Sim_{i,q}(C_{i,1}, C_{i,2})]^T$, $[A_i(C_{i,1}, C_{i,2})]^T$) is the least one among rss($[Sim_{i,j}(C_{i,1}, C_{i,2})]^T$, $[A_i(C_{i,1}, C_{i,2})]^T$) and output it

end SelectAMeasureWithHighAccuracy

Appendix C. Proof of the validity of the proposed method

Proof. Let $(C_{1,1}, C_{1,2})$, $(C_{2,1}, C_{2,2})$, …, $(C_{N,1}, C_{N,2})$ be the N sample concept pairs whose two components are respectively extracted from two different CAD model data ontologies, $A_i(C_{i,1}, C_{i,2})$ (i = 1, 2, …, N) be the actual semantic similarities of $(C_{i,1}, C_{i,2})$, corr($\boldsymbol{U}$, $\boldsymbol{V}$) be the Pearson correlation coefficient between vector $\boldsymbol{U}$ and vector $\boldsymbol{V}$, and rss($\boldsymbol{U}$, $\boldsymbol{V}$) be the residual sum of squares between $\boldsymbol{U}$ and $\boldsymbol{V}$. Further, let:

vector $\boldsymbol{X_1} = [Sim_{i,1}(C_{i,1}, C_{i,2})]^T$, vector $\boldsymbol{X_2} = [Sim_{i,2}(C_{i,1}, C_{i,2})]^T$, vector $\boldsymbol{X_3} = [Sim_{i,3}(C_{i,1}, C_{i,2})]^T$,

vector $\boldsymbol{X_4} = [Sim_{i,4}(C_{i,1}, C_{i,2})]^T$, vector $\boldsymbol{X_5} = [Sim_{i,5}(C_{i,1}, C_{i,2})]^T$, vector $\boldsymbol{X_6} = [Sim_{i,6}(C_{i,1}, C_{i,2})]^T$,

vector $\boldsymbol{X_7} = [Sim_{i,7}(C_{i,1}, C_{i,2})]^T$, vector $\boldsymbol{X_8} = [Sim_{i,8}(C_{i,1}, C_{i,2})]^T$, vector $\boldsymbol{X_9} = [Sim_{i,9}(C_{i,1}, C_{i,2})]^T$,

vector $\boldsymbol{Y_1} = [Sim_{\mathbf{Tversky}}(D_{C,i,1}, D_{C,i,2})]^T$, vector $\boldsymbol{Y_2} = [Sim_{\mathbf{Petrakis}}(D_{C,i,1}, D_{C,i,2})]^T$,

vector $\boldsymbol{Y_3} = [Sim_{\mathbf{Sanchez}}(D_{C,i,1}, D_{C,i,2})]^T$, vector $\boldsymbol{Y_4} = [Sim_{\mathbf{Tversky}}(R_{C,i,1}, R_{C,i,2})]^T$,

vector $\boldsymbol{Y_5} = [Sim_{\mathbf{Petrakis}}(R_{C,i,1}, R_{C,i,2})]^T$, vector $\boldsymbol{Y_6} = [Sim_{\mathbf{Sanchez}}(R_{C,i,1}, R_{C,i,2})]^T$,

vector $\boldsymbol{Z} = [A_i(C_{i,1}, C_{i,2})]^T$, and vector $\boldsymbol{S} = [Sim_{i,\mathbf{S}}(C_{i,1}, C_{i,2})]^T$,

where $Sim_{i,j}(C_{i,1}, C_{i,2})$ (j = 1, 2,…, 9) is the semantic similarities of $(C_{i,1}, C_{i,2})$ assessed by the measure $Sim_j(C_1, C_2)$ in Expression (28)), $Sim_{\mathbf{Tversky}}(D_{C,i,1}, D_{C,i,2})$, $Sim_{\mathbf{Petrakis}}(D_{C,i,1}, D_{C,i,2})$, and $Sim_{\mathbf{Sanchez}}(D_{C,i,1}, D_{C,i,2})$ are respectively the similarities of the semantic descriptions of $(C_{i,1}, C_{i,2})$ assessed by Tversky's measure (Expression (4)), Petrakis et al.'s measure (Expression (8)), and Sánchez et al.'s measure (Expression (9)), $Sim_{\mathbf{Tversky}}(R_{C,i,1}, R_{C,i,2})$, $Sim_{\mathbf{Petrakis}}(R_{C,i,1}, R_{C,i,2})$, and $Sim_{\mathbf{Sanchez}}(R_{C,i,1}, R_{C,i,2})$ are respectively the similarities of the semantic relationships of $(C_{i,1}, C_{i,2})$ assessed by Tversky's measure, Petrakis et al.'s measure, and Sánchez et al.'s measure, and $Sim_{i,\mathbf{S}}(C_{i,1}, C_{i,2})$ is the similarities of $(C_{i,1}, C_{i,2})$ assessed by the measure selected by the designed measure selection algorithm.

(1) If the difference between some two of corr($\boldsymbol{X_j}$, $\boldsymbol{Z}$) ($\boldsymbol{j}$ = 1, 2,…, 9) is greater than or equal to 0.1, then $Sim_{i,\mathbf{S}}(C_{i,1}, C_{i,2})$ is the measure whose correlation coefficient with $A_i(C_{i,1}, C_{i,2})$ (i.e. corr($\boldsymbol{S}$, $\boldsymbol{Z}$)) is the greatest one among all corr($\boldsymbol{X_j}$, $\boldsymbol{Z}$) according to the measure selection algorithm. Thus: (**a**) corr($\boldsymbol{S}$, $\boldsymbol{Z}$) ≥ corr($\boldsymbol{X_j}$, $\boldsymbol{Z}$).

For $Sim_1(C_1, C_2)$ in Expression (28), let vector $\boldsymbol{A_1} = [Sim_{\mathbf{Tversky}}(D_{C,i,1}, D_{C,i,2}), Sim_{\mathbf{Tversky}}(R_{C,i,1}, R_{C,i,2})]^T$, $\boldsymbol{w_1^o} = [w_{1,1}^{\mathbf{o}}, w_{1,2}^{\mathbf{o}}]^T$ be a vector that can maximize corr($\boldsymbol{w_1^{oT}A_1}$, $\boldsymbol{Z}$), and $\boldsymbol{w_1} = [w_{1,1}, w_{1,2}]^T$ be the final weight vector obtained from normalizing the vector $\boldsymbol{w_1^o}$. Then when $w_{1,1}^{\mathbf{o}} \geq 0$ and $w_{1,2}^{\mathbf{o}} \geq 0$ (Please note that corr($\boldsymbol{w_1^T A_1}$, $\boldsymbol{Z}$) = corr($\boldsymbol{X_1}$, $\boldsymbol{Z}$) = corr($\boldsymbol{Y_1}$, $\boldsymbol{Z}$) when $w_{1,2}^{\mathbf{o}} < 0$ and corr($\boldsymbol{w_1^T A_1}$, $\boldsymbol{Z}$) = corr($\boldsymbol{X_1}$, $\boldsymbol{Z}$) = corr($\boldsymbol{Y_4}$, $\boldsymbol{Z}$) when $w_{1,1}^{\mathbf{o}} < 0$):

$$\boldsymbol{w}_1 = \boldsymbol{w}_1^{\mathbf{o}} / (w_{1,1}^{\mathbf{o}} + w_{1,2}^{\mathbf{o}}) = \boldsymbol{w}_1^{\mathbf{o}} / \mathrm{c}$$

$$\mathrm{corr}(\boldsymbol{w}_1^{\mathrm{T}} \boldsymbol{A}_1,\ \boldsymbol{Z}) = \frac{\boldsymbol{w}_1^{\mathrm{T}} \mathrm{cov}(\boldsymbol{A}_1, \boldsymbol{Z})}{\sqrt{\boldsymbol{w}_1^{\mathrm{T}} \mathrm{cov}(\boldsymbol{A}_1, \boldsymbol{A}_1) \boldsymbol{w}_1} \sqrt{\mathrm{cov}(\boldsymbol{Z}, \boldsymbol{Z})}} = \frac{\left(\boldsymbol{w}_1^{\mathbf{o}} / \mathrm{c}\right)^{\mathrm{T}} \mathrm{cov}(\boldsymbol{A}_1, \boldsymbol{Z})}{\sqrt{\left(\boldsymbol{w}_1^{\mathbf{o}} / \mathrm{c}\right)^{\mathrm{T}} \mathrm{cov}(\boldsymbol{A}_1, \boldsymbol{A}_1) \left(\boldsymbol{w}_1^{\mathbf{o}} / \mathrm{c}\right)} \sqrt{\mathrm{cov}(\boldsymbol{Z}, \boldsymbol{Z})}} = \mathrm{corr}(\boldsymbol{w}_1^{\mathbf{o}\mathrm{T}} \boldsymbol{A}_1,\ \boldsymbol{Z})$$

Since corr($\boldsymbol{w}_1^{\mathbf{o}\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) is the greatest correlation coefficient, corr($\boldsymbol{w}_1^{\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) is also the greatest correlation coefficient among the correlation coefficients between the similarities of ($C_{i,1}$, $C_{i,2}$) assessed by all possible linear combinations of $Sim_{\mathbf{Tversky}}(D_{C,i,1}, D_{C,i,2})$ and $Sim_{\mathbf{Tversky}}(R_{C,i,1}, R_{C,i,2})$ and the actual semantic similarities of ($C_{i,1}$, $C_{i,2}$) (i.e. $A_i(C_{i,1}, C_{i,2})$). This includes the following cases:

1) If the correlation coefficient corr($\boldsymbol{w}_1^{\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) obtains the greatest value when $w_{1,1} = 1$ and $w_{1,2} = 0$, then **(b)** corr($\boldsymbol{w}_1^{\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) = corr($\boldsymbol{X}_1$, $\boldsymbol{Z}$) = corr($\boldsymbol{Y}_1$, $\boldsymbol{Z}$);

2) If the correlation coefficient corr($\boldsymbol{w}_1^{\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) obtains the greatest value when $w_{1,1} = 0$ and $w_{1,2} = 1$, then **(c)** corr($\boldsymbol{w}_1^{\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) = corr($\boldsymbol{X}_1$, $\boldsymbol{Z}$) = corr($\boldsymbol{Y}_4$, $\boldsymbol{Z}$);

3) If the correlation coefficient corr($\boldsymbol{w}_1^{\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) obtains the greatest value when $0 < w_{1,1} < 1$ and $0 < w_{1,2} < 1$, then **(d)** corr($\boldsymbol{w}_1^{\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) = corr($\boldsymbol{X}_1$, $\boldsymbol{Z}$) > corr($\boldsymbol{Y}_1$, $\boldsymbol{Z}$) and corr($\boldsymbol{w}_1^{\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) = corr($\boldsymbol{X}_1$, $\boldsymbol{Z}$) > corr($\boldsymbol{Y}_4$, $\boldsymbol{Z}$).

Based on (**a**), (**b**), (**c**), and (**d**), corr($\boldsymbol{S}$, $\boldsymbol{Z}$) ≥ corr($\boldsymbol{X}_1$, $\boldsymbol{Z}$) ≥ {corr($\boldsymbol{Y}_1$, $\boldsymbol{Z}$), corr($\boldsymbol{Y}_4$, $\boldsymbol{Z}$)} holds.

Similarly, for $Sim_k(C_1, C_2)$ (k = 2, 3,…, 9) in Expression (28), it can be proved that:

corr($\boldsymbol{S}$, $\boldsymbol{Z}$) ≥ corr($\boldsymbol{X}_2$, $\boldsymbol{Z}$) ≥ {corr($\boldsymbol{Y}_1$, $\boldsymbol{Z}$), corr($\boldsymbol{Y}_5$, $\boldsymbol{Z}$)}, corr($\boldsymbol{S}$, $\boldsymbol{Z}$) ≥ corr($\boldsymbol{X}_3$, $\boldsymbol{Z}$) ≥ {corr($\boldsymbol{Y}_1$, $\boldsymbol{Z}$), corr($\boldsymbol{Y}_6$, $\boldsymbol{Z}$)},

corr($\boldsymbol{S}$, $\boldsymbol{Z}$) ≥ corr($\boldsymbol{X}_4$, $\boldsymbol{Z}$) ≥ {corr($\boldsymbol{Y}_2$, $\boldsymbol{Z}$), corr($\boldsymbol{Y}_4$, $\boldsymbol{Z}$)},corr($\boldsymbol{S}$, $\boldsymbol{Z}$) ≥ corr($\boldsymbol{X}_5$, $\boldsymbol{Z}$) ≥ {corr($\boldsymbol{Y}_2$, $\boldsymbol{Z}$), corr($\boldsymbol{Y}_5$, $\boldsymbol{Z}$)},

corr($\boldsymbol{S}$, $\boldsymbol{Z}$) ≥ corr($\boldsymbol{X}_6$, $\boldsymbol{Z}$) ≥ {corr($\boldsymbol{Y}_2$, $\boldsymbol{Z}$), corr($\boldsymbol{Y}_6$, $\boldsymbol{Z}$)},corr($\boldsymbol{S}$, $\boldsymbol{Z}$) ≥ corr($\boldsymbol{X}_7$, $\boldsymbol{Z}$) ≥ {corr($\boldsymbol{Y}_3$, $\boldsymbol{Z}$), corr($\boldsymbol{Y}_4$, $\boldsymbol{Z}$)},

corr($\boldsymbol{S}$, $\boldsymbol{Z}$) ≥ corr($\boldsymbol{X}_8$, $\boldsymbol{Z}$) ≥ {corr($\boldsymbol{Y}_3$, $\boldsymbol{Z}$), corr($\boldsymbol{Y}_5$, $\boldsymbol{Z}$)}, corr($\boldsymbol{S}$, $\boldsymbol{Z}$) ≥ corr($\boldsymbol{X}_9$, $\boldsymbol{Z}$) ≥ {corr($\boldsymbol{Y}_3$, $\boldsymbol{Z}$), corr($\boldsymbol{Y}_6$, $\boldsymbol{Z}$)}.

(2) If the difference between some two of corr($\boldsymbol{X}_j$, $\boldsymbol{Z}$) ($\boldsymbol{j}$ = 1, 2,…, 9) is less than 0.1, then $Sim_{i,\mathbf{S}}(C_{i,1}, C_{i,2})$ is the measure whose residual sum of squares with $A_i(C_{i,1}, C_{i,2})$ (i.e. rss($\boldsymbol{S}$, $\boldsymbol{Z}$)) is the least one among all rss($\boldsymbol{X}_j$, $\boldsymbol{Z}$) according to the measure selection algorithm. Thus: (**e**) rss($\boldsymbol{S}$, $\boldsymbol{Z}$) ≤ rss($\boldsymbol{X}_j$, $\boldsymbol{Z}$).

For $Sim_1(C_1, C_2)$ in Expression (28), let vector $\boldsymbol{A}_1 = [Sim_{\mathbf{Tversky}}(D_{C,i,1}, D_{C,i,2}), Sim_{\mathbf{Tversky}}(R_{C,i,1}, R_{C,i,2})]^{\mathrm{T}}$, $\boldsymbol{w}_1^{\mathbf{o}} = [w_{1,1}^{\mathbf{o}}, w_{1,2}^{\mathbf{o}}]^{\mathrm{T}}$ be a vector that can minimize rss($\boldsymbol{w}_1^{\mathbf{o}\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$), and $\boldsymbol{w}_1 = [w_{1,1}, w_{1,2}]^{\mathrm{T}}$ be the final weight vector obtained from normalizing the vector $\boldsymbol{w}_1^{\mathbf{o}}$. Then: rss($\boldsymbol{w}_1^{\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) = rss($\boldsymbol{w}_1^{\mathbf{o}\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) since $\boldsymbol{w}_1 = \boldsymbol{w}_1^{\mathbf{o}}$ when $0 \le w_{1,1}^{\mathbf{o}} \le 1$ and $0 \le w_{1,2}^{\mathbf{o}} \le 1$ (Please note that rss($\boldsymbol{w}_1^{\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) = rss($\boldsymbol{X}_1$, $\boldsymbol{Z}$) = rss($\boldsymbol{Y}_1$, $\boldsymbol{Z}$) when $w_{1,1}^{\mathbf{o}} > 1$ or $w_{1,2}^{\mathbf{o}} < 0$ and rss($\boldsymbol{w}_1^{\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) = rss($\boldsymbol{X}_1$, $\boldsymbol{Z}$) = rss($\boldsymbol{Y}_4$, $\boldsymbol{Z}$) when $w_{1,1}^{\mathbf{o}} < 0$ or $w_{1,2}^{\mathbf{o}} > 1$). Because rss($\boldsymbol{w}_1^{\mathbf{o}\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) is the least residual sum of squares, rss($\boldsymbol{w}_1^{\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) is also the least residual sum of squares among the residual sums of squares between the similarities of ($C_{i,1}$, $C_{i,2}$) assessed by all possible linear combinations of $Sim_{\mathbf{Tversky}}(D_{C,i,1}, D_{C,i,2})$ and $Sim_{\mathbf{Tversky}}(R_{C,i,1}, R_{C,i,2})$ and the actual semantic similarities of ($C_{i,1}$, $C_{i,2}$) (i.e. $A_i(C_{i,1}, C_{i,2})$). This includes the following cases:

1) If the residual sum of squares rss($\boldsymbol{w}_1^{\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) obtains the least value when $w_{1,1} = 1$ and $w_{1,2} = 0$, then **(f)** rss($\boldsymbol{w}_1^{\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) = rss($\boldsymbol{X}_1$, $\boldsymbol{Z}$) = rss($\boldsymbol{Y}_1$, $\boldsymbol{Z}$);

2) If the residual sum of squares rss($\boldsymbol{w}_1^{\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) obtains the least value when $w_{1,1} = 0$ and $w_{1,2} = 1$, then **(g)** rss($\boldsymbol{w}_1^{\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) = rss($\boldsymbol{X}_1$, $\boldsymbol{Z}$) = rss($\boldsymbol{Y}_4$, $\boldsymbol{Z}$);

3) If the residual sum of squares rss($\boldsymbol{w}_1^{\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) obtains the least value when $0 < w_{1,1} < 1$ and $0 < w_{1,2} < 1$, then **(h)** rss($\boldsymbol{w}_1^{\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) = rss($\boldsymbol{X}_1$, $\boldsymbol{Z}$) < rss($\boldsymbol{Y}_1$, $\boldsymbol{Z}$) and rss($\boldsymbol{w}_1^{\mathrm{T}}\boldsymbol{A}_1$, $\boldsymbol{Z}$) = rss($\boldsymbol{X}_1$, $\boldsymbol{Z}$) < rss($\boldsymbol{Y}_4$, $\boldsymbol{Z}$).

Based on (**e**), (**f**), (**g**), and (**h**), rss($\boldsymbol{S}$, $\boldsymbol{Z}$) ≤ rss($\boldsymbol{X}_1$, $\boldsymbol{Z}$) ≤ {rss($\boldsymbol{Y}_1$, $\boldsymbol{Z}$), rss($\boldsymbol{Y}_4$, $\boldsymbol{Z}$)} holds.

Similarly, for $Sim_k(C_1, C_2)$ ($k = 2, 3, \ldots, 9$) in Expression (28), it can be proved that:

$\text{rss}(\boldsymbol{S}, \boldsymbol{Z}) \le \text{rss}(\boldsymbol{X_2}, \boldsymbol{Z}) \le \{\text{rss}(\boldsymbol{Y_1}, \boldsymbol{Z}), \text{rss}(\boldsymbol{Y_5}, \boldsymbol{Z})\}$, $\text{rss}(\boldsymbol{S}, \boldsymbol{Z}) \le \text{rss}(\boldsymbol{X_3}, \boldsymbol{Z}) \le \{\text{rss}(\boldsymbol{Y_1}, \boldsymbol{Z}), \text{rss}(\boldsymbol{Y_6}, \boldsymbol{Z})\}$,

$\text{rss}(\boldsymbol{S}, \boldsymbol{Z}) \le \text{rss}(\boldsymbol{X_4}, \boldsymbol{Z}) \le \{\text{rss}(\boldsymbol{Y_2}, \boldsymbol{Z}), \text{rss}(\boldsymbol{Y_4}, \boldsymbol{Z})\}$, $\text{rss}(\boldsymbol{S}, \boldsymbol{Z}) \le \text{rss}(\boldsymbol{X_5}, \boldsymbol{Z}) \le \{\text{rss}(\boldsymbol{Y_2}, \boldsymbol{Z}), \text{rss}(\boldsymbol{Y_5}, \boldsymbol{Z})\}$,

$\text{rss}(\boldsymbol{S}, \boldsymbol{Z}) \le \text{rss}(\boldsymbol{X_6}, \boldsymbol{Z}) \le \{\text{rss}(\boldsymbol{Y_2}, \boldsymbol{Z}), \text{rss}(\boldsymbol{Y_6}, \boldsymbol{Z})\}$, $\text{rss}(\boldsymbol{S}, \boldsymbol{Z}) \le \text{rss}(\boldsymbol{X_7}, \boldsymbol{Z}) \le \{\text{rss}(\boldsymbol{Y_3}, \boldsymbol{Z}), \text{rss}(\boldsymbol{Y_4}, \boldsymbol{Z})\}$,

$\text{rss}(\boldsymbol{S}, \boldsymbol{Z}) \le \text{rss}(\boldsymbol{X_8}, \boldsymbol{Z}) \le \{\text{rss}(\boldsymbol{Y_3}, \boldsymbol{Z}), \text{rss}(\boldsymbol{Y_5}, \boldsymbol{Z})\}$, $\text{rss}(\boldsymbol{S}, \boldsymbol{Z}) \le \text{rss}(\boldsymbol{X_9}, \boldsymbol{Z}) \le \{\text{rss}(\boldsymbol{Y_3}, \boldsymbol{Z}), \text{rss}(\boldsymbol{Y_6}, \boldsymbol{Z})\}$. □

References

[1] K. Lee, Principles of CAD/CAM/CAE systems, Addison Wesley Longman, Inc., Boston, 1999.

[2] ISO 10303-1, Industrial automation systems and integration — Product data representation and exchange — Part 1: Overview and fundamental principles, International Organization for Standardization, Geneva, 1994.

[3] ISO 10303-11, Industrial automation systems and integration — Product data representation and exchange — Part 11: Description methods: The EXPRESS language reference manual, International Organization for Standardization, Geneva, 2004.

[4] M.I. Sarigecili, U. Roy, S. Rachuri, Interpreting the semantics of GD&T specifications of a product for tolerance analysis, Comput.-Aided Des. 47(2) (2014) 72-84.

[5] J. Kim, M.J. Pratt, R.G. Iyer, R.D. Sriram, Standardized data exchange of CAD models with design intent, Comput.-Aided Des. 40(7) (2008) 760-777.

[6] V. Fortineau, T. Paviot, S. Lamouri, Improving the interoperability of industrial information systems with description logic-based models — The state of the art, Comput. Ind. 64(4) (2013) 363-375.

[7] T. Berners-Lee, J. Hendler, O. Lassila, The Semantic Web, Sci. Am. 284(5) (2001) 28-37.

[8] C. Dartigues, P. Ghodous, Product data exchange using ontologies, in: Artificial Intelligence in Design'02, Springer Netherlands, 2002, pp. 617-637.

[9] L. Patil, D. Dutta, R. Sriram, Ontology-based exchange of product data semantics, IEEE Trans. Autom. Sci. Eng. 2(3) (2005) 213-225.

[10] K.Y. Kim, D.G. Manley, H. Yang, Ontology-based assembly design and information sharing for collaborative product development, Comput.-Aided Des. 38(12) (2006) 1233-1250.

[11] Q.Z. Yang, C.Y. Miao, Semantic enhancement and ontology for interoperability of design information systems, in: IEEE Conference on Emerging Technologies and Factory Automation, IEEE, 2007, pp. 169-176.

[12] R.K. Gupta, B. Gurumoorthy, A feature-based framework for semantic interoperability of product models, Strojniski Vestn.-J. Mech. Eng. 54(6) (2008) 446-457.

[13] R. Barbau, S. Krima, S. Rachuri, A. Narayanan, X. Fiorentini, S. Foufou, R.D. Sriram, OntoSTEP: Enriching product model data using ontologies, Comput.-Aided Des. 44(6) (2012) 575-590.

[14] H. Panetto, M. Dassisti, A. Tursi, ONTO-PDM: Product-driven ONTOlogy for Product Data Management interoperability within manufacturing process environment, Adv. Eng. Inform. 26(2) (2012) 334-348.

[15] S. Tessier, Y. Wang, Ontology-based feature mapping and verification between CAD systems, Adv. Eng. Inform. 27(1) (2013) 76-92.

[16] L. Patil, Interoperability of formal semantics of product data across product development systems, Ph.D. dissertation, University of Michigan, Ann Arbor, 2005.

[17] M.J. Lee, M. Jung, H.W. Suh, Semantic mapping based on ontology and a Bayesian Network and its application to CAD and PDM integration, in: Proceedings of the ASME 2006 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, American

Society of Mechanical Engineers, 2006, pp. 591-602.
[18] P. Zhan, U. Jayaram, O. Kim, L. Zhu, Knowledge representation and ontology mapping methods for product data in engineering applications, J. Comput. Inf. Sci. Eng. 10(2) (2010) 021004-021004-11.
[19] S. Abdul-Ghafour, P. Ghodous, B. Shariat, E. Perna, F. Khosrowshahi, Semantic interoperability of knowledge in feature-based CAD models, Comput.-Aided Des. 56(11) (2014) 45-57.
[20] D.L. McGuinness, F.V. Harmelen, OWL Web Ontology Language Overview W3C Recommendation, 2004. <http://www.w3.org/TR/owl-features/>.
[21] I. Horrocks, P.F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, M. Dean, SWRL: A Semantic Web rule language combining OWL and RuleML, 2004. <http://www.w3.org/Submission/SWRL/>.
[22] S. Harispe, D. Sánchez, S. Ranwez, S. Janaqi, J. Montmain, A framework for unifying ontology-based semantic similarity measures: A study in the biomedical domain, J. Biomed. Inform. 48(4) (2014) 38-53.
[23] A Tversky, Features of similarity, Psychol. Rev. 84(4) (1977) 327-352.
[24] E.G.M. Petrakis, G. Varelas, A. Hliaoutakis, P. Raftopoulou, X-similarity: Computing semantic similarity between concepts from different ontologies, J. Digit. Inf. Manag. 4(4) (2006) 233-237.
[25] D. Sánchez, M. Batet, D. Isern, A. Valls, Ontology-based semantic similarity: A new feature-based approach, Expert Syst. Appl. 39(9) (2012) 7718-7728.
[26] M.A. Rodríguez, M.J. Egenhofer, Determining semantic similarity among entity classes from different ontologies, IEEE Trans. Knowl. Data Eng. 15(2) (2003) 442-456.
[27] Y. Jiang, X. Zhang, Y. Tang, R. Nie, Feature-based approaches to semantic similarity assessment of concepts using Wikipedia, Inf. Process Manage. 51(3) (2015) 215-234.
[28] L. Zhu, U. Jayaram, S. Jayaram, O. Kim, Ontology-driven integration of CAD/CAE applications: Strategies and comparisons, in: Proceedings of the ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, American Society of Mechanical Engineers, 2009, pp. 1461-1472.
[29] F. Baader, D. Calvanese, D.L. McGuinness, D. Nardi, P.F. Patel-Schneider, The description logic handbook: Theory, implementation and applications, 2nd ed, Cambridge University Press, Cambridge, 2010.
[30] D. Weenink, Canonical correlation analysis, in: Proceedings of the Institute of Phonetic Sciences of the University of Amsterdam, University of Amsterdam, 2003, pp. 81-99.
[31] I. Horrocks, U. Sattler, S. Tobies, Practical reasoning for expressive description logics, in: Proceedings of the 6th International Conference on Logic for Programming and Automated Reasoning, Springer Berlin Heidelberg, 1999, pp. 161-180.
[32] E. Zolin, Complexity of reasoning in description logics, School of Computer Science, The University of Manchester, 2013. <http://www.cs.man.ac.uk/~ezolin/dl/>.
[33] M. Ortiz, S. Rudolph, M. Simkus, Worst-case optimal reasoning for the horn-DL fragments of OWL 1 and 2, in: Proceedings of 12th International Conference on the Principles of Knowledge Representation and Reasoning, AAAI Press, 2010, pp. 269-279.
[34] Protégé 3.5, Stanford Center for Biomedical Informatics Research, 2013. <http://protege.stanford.edu/>.
[35] E. Friedman-Hill, Jess in action: Java rule-based systems, Manning Publications, Greenwich, 2003.
[36] H. Rubenstein, J.B. Goodenough, Contextual correlates of synonymy, Commun. ACM 8(10) (1965) 627-633.
[37] G.A. Miller, W.G. Charles, Contextual correlates of semantic similarity, Lang. Cogn. Process 6(1) (1991) 1-28.
[38] L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, E. Ruppin, Placing Search in Context: The Concept Revisited, ACM Trans. Inf. Syst. 20(1) (2002) 116-131.