Advanced Engineering Informatics 30 (2016) 449–466

Contents lists available at ScienceDirect

Advanced Engineering Informatics

journal homepage: www.elsevier.com/locate/aei

Full length article

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, School of Computing and Engineering, University of Huddersfield, Huddersfield, HD1 3DH, UK
[c] Guangxi Colleges and Universities Key Laboratory of Intelligent Processing of Computer Images and Graphics, School of Computer Science and Information Security, Guilin University of Electronic Technology, Guilin 541004, PR China

ARTICLE INFO

Article history:
Received 10 November 2015
Received in revised form 20 March 2016
Accepted 7 June 2016

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

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.

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 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].

---

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

http://dx.doi.org/10.1016/j.aei.2016.06.001
1474-0346/

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]:

$$\begin{aligned}\mathit{PROE\text{-}Extrude} \equiv\ & \mathit{PROE\text{-}Feature} \sqcap \\ & \exists \mathit{proe\text{-}hasParent}.\mathit{PROE\text{-}Sketch}\end{aligned}$$

$$\begin{aligned}\mathit{UGNX\text{-}Extrude} \equiv\ & \mathit{UGNX\text{-}Feature} \sqcap \\ & \exists \mathit{ugnx\text{-}hasParent}.\mathit{UGNX\text{-}Sketch}\end{aligned}$$

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:

$$\begin{aligned}&\mathit{PROE\text{-}Feature} \equiv \mathit{UGNX\text{-}Feature},\\ &\mathit{PROE\text{-}Sketch} \equiv \mathit{UGNX\text{-}Sketch},\\ &\mathit{proe\text{-}hasParent} \equiv \mathit{ugnx\text{-}hasParent}\end{aligned}$$

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:

$$\begin{aligned}\mathrm{PROE}:\ & \mathit{PROE\text{-}Extrude}(\mathrm{EXTRUDE1}),\\ & \mathit{PROE}-\mathit{Sketch}(\mathrm{SKETCH1}),\\ & \mathit{proe\text{-}hasParent}(\mathrm{EXTRUDE1},\mathrm{SKETCH1})\end{aligned}$$

$$\begin{aligned}\mathrm{UGNX}:\ & \mathit{UGNX\text{-}Extrude}(\mathrm{EXTRUDE1}),\\ & \mathit{UGNX\text{-}Sketch}(\mathrm{SKETCH1}),\\ & \mathit{ugnx\text{-}hasParent}(\mathrm{EXTRUDE1},\mathrm{SKETCH1})\end{aligned}$$

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 "$\mathit{PROE\text{-}Extrude} \equiv \mathit{PROE\text{-}Feature} \sqcap \exists \mathit{proe\text{-}hasParent}.\ \mathit{PROE\text{-}Sketch}$"), 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 technology. 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:

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

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

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