Computers in Industry 65 (2014) 91–107

Contents lists available at ScienceDirect

Computers in Industry

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

Ontology-based similarity for product information retrieval

Suriati Akmal [a,1], Li-Hsing Shih [b], Rafael Batres [a,*]

[a] Department of Mechanical Engineering, Toyohashi University of Technology, Hibarigaoka 1-1, Tempaku-cho, Toyohashi 441-8580 Japan
[b] Department of Resources Engineering, National Cheng Kung University, No. 1, University Road, Tainan 701, Taiwan, ROC

ARTICLE INFO

Article history:
Received 4 September 2012
Received in revised form 10 June 2013
Accepted 31 July 2013
Available online 21 September 2013

Keywords:
Semantic similarity
Ontology
Product information retrieval
Formal concept analysis

ABSTRACT

Product development of today is becoming increasingly knowledge intensive. Specifically, design teams face considerable challenges in making effective use of increasing amounts of information. In order to support product information retrieval and reuse, one approach is to use case-based reasoning (CBR) in which problems are solved "by using or adapting solutions to old problems." In CBR, a case includes both a representation of the problem and a solution to that problem. Case-based reasoning uses similarity measures to identify cases which are more relevant to the problem to be solved. However, most non-numeric similarity measures are based on syntactic grounds, which often fail to produce good matches when confronted with the meaning associated to the words they compare. To overcome this limitation, ontologies can be used to produce similarity measures that are based on semantics. This paper presents an ontology-based approach that can determine the similarity between two classes using feature-based similarity measures that replace features with attributes. The proposed approach is evaluated against other existing similarities. Finally, the effectiveness of the proposed approach is illustrated with a case study on product–service–system design problems.

1. Introduction

Due to the complexity of products and drastic technological changes, product development is becoming increasingly knowledge intensive. Design is also multi-disciplinary in nature requiring a variety of product life-cycle knowledge [1]. Specifically, design teams face a considerable challenge in making effective use of increasing amounts of information that often accumulate and remain in individual information systems. Also, it is often the case that product designers can reuse past designs rather than designing from scratch [2].

Information retrieval consists of translating and matching a query against a set of information objects. Translation of the query is necessary for converting the user requirements into the language provided by the information retrieval system. The information retrieval system responds to the query using a given algorithm and a similarity measure. Particularly, information retrieval plays an important role in areas such as product family design [3], product embodiment, and detailed design [4]. Shah et al. [5] present a combination framework that consists of software engineering, data engineering and knowledge engineering and design theory.

In order to support product information retrieval and reuse, some authors suggest the use of case-based reasoning (CBR) in which design problems are solved by using or adapting previous design solutions [4,6].

A CBR system is composed of domain knowledge, a case base, and a search mechanism based on a similarity measure. Domain knowledge refers to knowledge about the features of the different objects or entities that a case is about. A case base contains a set of cases, each of which describes a problem and a solution to the problem. The problem is typically defined in terms of specific features of objects. Finally, a similarity measure quantifies the differences that exist between objects [7]. CBR uses similarity measures to identify cases which are more relevant to the problem to be solved.

Most similarity measures evaluate differences between values of numeric properties such as in the numerical difference between two given diameter values. However, many applications also require non-numeric similarities. For example, case-based reasoning systems for the conceptual design of products must be developed to work with a limited knowledge about the product.

Nearly all of non-numeric similarity measures are based on syntactic grounds. For example, the Levenshtein distance [8,9] can be used to calculate the similarity between two words, in terms of the minimum number of operations that are needed to transform one of the words into the other. However, from the point of view of the meaning of the words that are compared, existing syntactic similarity-measures often result in incorrect matches.

* Corresponding author. Tel.: +81 532 44 6716; fax: +81 532 44 6716.
E-mail address: rafaelbatres@hotmail.com (R. Batres).
[1] On leave from the Universiti Teknikal Malaysia Melaka, 76109 Durian Tunggal, Melaka, Malaysia.

0166-3615/$ – see front matter
http://dx.doi.org/10.1016/j.compind.2013.07.011

Semantic similarity measures can be used in order to overcome the limitations of syntactic approaches. A semantic similarity is a function that assigns a numeric value to the similarity between two classes of objects based on the meaning associated to each of the objects [10]. For a review of semantic similarity metrics, the reader is referred to the paper of Cross and Hu [11].

Recently, the use of ontologies for evaluating similarity has been reported in the literature [12,13]. Ontologies are formal models that use mathematical logic to disambiguate and define classes of things [14]. Specifically, ontologies describe a shared and common understanding of a domain in terms of classes, possible relations between things, and axioms that constrain the meaning of classes and relations [15]. A class represents a set of things that share the same attributes. A relation is used to represent a relationship among two or more things. Examples of relations are less than, connected to, and part of. Class taxonomies are defined by means of the subclass relation. A class is a subclass of another class if every member of the subclass is also a member of the super class. Axioms are typically represented as logic constructions that formally define a given class or relation.

Combined with automated reasoning applications, ontologies can be used for several purposes such as knowledge extraction and information retrieval. Unfortunately, ontologies are typically created in an ad-hoc manner, which may influence the accuracy of the similarity calculations.

Formal concept analysis (FCA) is a data processing method that can be used to design ontologies [16,17]. FCA is based on a set of objects and a set of attributes. In this paper, we use FCA along with a theoretical framework for developing product and process ontologies.

Most semantic similarities are defined in terms of the number of edges between the classes that they compare (edge-counting similarity measures). Other semantic similarities are defined in terms of features but use synsets for the comparison between words rather than classes.

The underlying thesis in this paper is that if a class represents a set of things that share the same attributes (such as a class in an ontology), we can state that a class is equivalent to another class if both classes have exactly the same attributes. This implies that the more common attributes that are shared by two classes the more similar they are. In this paper, we show how an ontology-based approach can determine the similarity between two classes using feature-based similarity measures that replace features with attributes.

The paper is organized as follows. Section 2 describes the theoretical framework for product representation used in this paper. Section 3 provides details on the ontology development. Section 4 describes the proposed ontology-based similarity measures. Sections 5 and 6 describe the evaluation of the semantic measures proposed in this paper. In Section 7, the effectiveness of the proposed approach is illustrated with a case study on product-service-system design problems. Section 8 discusses some related work and Section 9 presents conclusions and suggestions.

2. Theoretical framework for product representation

Theoretical frameworks for product representation refer to the world view with which product information models or ontologies can be developed in order to represent a product. In this paper, the theoretical framework for representing a product is based on the ISO 15926 standard which specifies an upper ontology for long-term data integration, access and exchange [18]. It was developed in ISO TC184/SC4-Industrial Data by the EPISTLE consortium (1993–2003) and designed to support the evolution of data through time. The upper ontology was developed as a conceptual data model for the representation of technical information of process plants including oil and gas production facilities but it was designed to be generic enough for any engineering domain [19].

In this theoretical framework, the device is represented in terms of its physical aspects as well as in terms of its relation to some process (*activity* in ISO 15926). These aspects are illustrated in the models of Figs. 1 and 2.

A device is represented as a physical object that is defined in terms of a distribution of matter, energy, or both. The device is also described in terms of its parts. This is possible through a mereological relation that refers to the relationship that a part

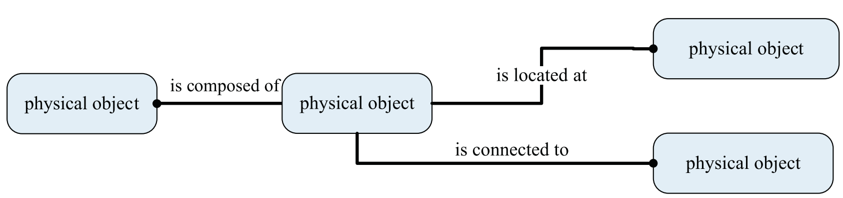

Fig. 1. Composition of device.

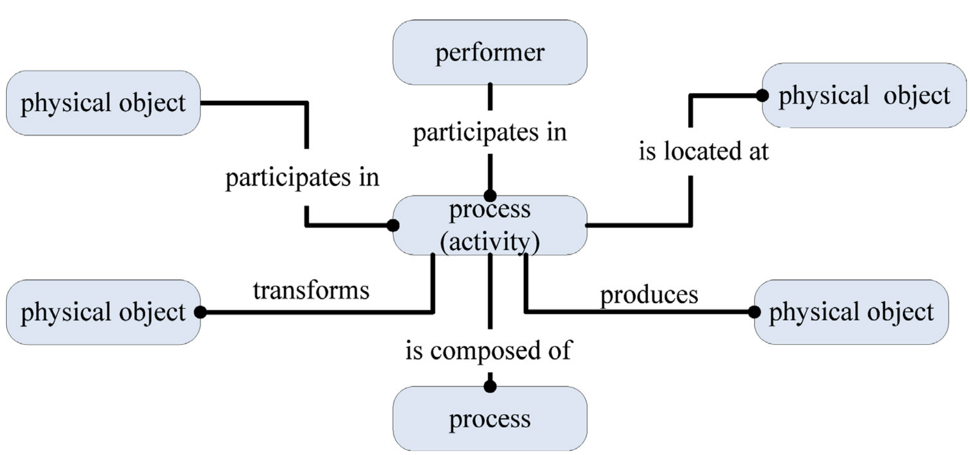

Fig. 2. Relations between device and process.