Journal of King Saud University – Computer and Information Sciences (2014) **26**, 247–257

King Saud University

Journal of King Saud University – Computer and Information Sciences

www.ksu.edu.sa
www.sciencedirect.com

Semantic matching in hierarchical ontologies

Sharifullah Khan *, **Muhammad Safyan**

School of Electrical Engineering and Computer Sciences, Pakistan
National University of Sciences and Technology, Islamabad, Pakistan

Received 11 May 2013; revised 5 November 2013; accepted 20 March 2014
Available online 24 May 2014

KEYWORDS

Hierarchical ontology;
Interoperability;
Ontology matching;
Ontology mapping

Abstract Hierarchical ontologies play a key role in organizing documents in a repository. While matching the ontologies, the relationships among the concepts are considered to be a major aspect. In hierarchical ontologies, the concepts are associated with one another only through the "*is-a*" relation. In this paper, we discuss an approach for matching heterogeneous hierarchical ontologies that are related to the same domain through the semantic interpretation and implicit context of the concepts. We have designed rules that can handle heterogeneities and inconsistencies that are found in hierarchical ontologies. These rules can be embedded to complement the existing matching systems, to resolve the matching complexities in the hierarchical ontologies.

© 2014 King Saud University. Production and hosting by Elsevier B.V. All rights reserved.

1. Introduction

An ontology provides a shared understanding of common domains and contributes to resolving interoperability issues among software applications across different autonomous organizations. However, the semantic web community agrees on the fact that a single ontology cannot be built due to the large variety of information sources on the web. Ontology matching is a solution to the semantic heterogeneity problem (Shvaiko and Euzenat, 2013; Kalfoglou and Schorlemmer, 2003; Ehrig and Sure, 2004). Ontology matching takes two ontologies as inputs and finds correspondences between semantically related entities in the ontologies to enable interoperability among them. These correspondences can be used for a variety of tasks, such as ontology merging, query answering, or data translation. Ontology matching is an important operation in applications such as information retrieval, natural language processing (NLP), health informatics, bio-informatics and ecommerce. Semantic matching of ontologies is a labor-intensive and error-prone process in integrating autonomous data sources, and it uses more than half of the integration efforts (Halevy, 2005; Halevy et al., 2006). Various ontology matching systems and algorithms have been proposed since the last decade (Pivovarov et al., 2012; Shvaiko et al., 2010; Hu et al., 2008; Giunchiglia et al., 2004, 2005, 2012; Jian et al., 2005; Ehrig and Sure, 2005; Ehrig and Staab, 2004; Do and Rahm, 2002). Several surveys (Shvaiko and Euzenat, 2005, 2013; Ehrig and Sure, 2004; Giunchiglia and Shvaiko, 2003; Kalfoglou and Schorlemmer, 2003) and books (Bellahsene et al., 2011; Euzenat et al., 2007) have been published on the topic as well.

* Corresponding author. Tel.: +92 51 9085 2150.
E-mail addresses: sharifullah.khan@seecs.edu.pk, skhan_phd@yahoo.co.uk (S. Khan).

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

1319-1578 © 2014 King Saud University. Production and hosting by Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.jksuci.2014.03.010

A hierarchical ontology is represented in the form of a directed acyclic graph in which a node models a concept and its label codifies the meaning of the concept. Relationships among the nodes are usually represented by *narrow-than* or *broader-than* relations in the graph. The hierarchical ontology classifies the concepts at each level and proceeds from generalized to specialized concepts. In the same subject domain, different hierarchical ontologies can have different classifications of concepts. In other words, similar concepts in different ontologies can be classified in different ways and are placed at different hierarchical levels. *Data type* properties, *object type* properties, relationships among concepts and their respective axioms usually define the context of an ontology (Shavaiko and Euzenat, 2005; Giunchiglia and Shvaiko, 2003; Giunchiglia et al., 2005). A hierarchical ontology is light-weight (Zuber and Faltings, 2007; Gomez-Perez et al., 2004), i.e., not rich in terms of its context. In other words, data type properties, object type property relationships among concepts and their respective axioms are missing in a hierarchical ontology. Web directories such as *Dmoz* (i.e., a Google directory)[1] and the Yahoo directory[2] and subject classification schemes such as the ACM Computing Classification System (ACM CCS)[3] and the Mathematics Subject Classification (MSC)[4] are examples of hierarchical ontologies. These systems are also known as taxonomies.

Hierarchical ontologies are usually categorized into *formal* and *informal* hierarchical ontologies (Gomez-Perez et al., 2004). Formal hierarchical ontologies strictly implement inheritance in sub-classes. An instance of a sub-class must be an instance of its super-class. For example, sub-classes of the concept *Travel* could be *Flight travel*, *Train travel*. However, informal hierarchical ontologies do not strictly follow inheritance in sub-classes. For example, *Car rental* and *Hotel* are sub-concepts of *Travel* in an informal hierarchical ontology, but they do not inherit *Travel* characteristics. This paper focuses on informal hierarchical ontologies.

In the same subject domain, heterogeneities exist in the structure of hierarchical ontologies. For example, the concepts are labeled and classified differently and placed at various levels in different hierarchies. We call these structural heterogeneities. Reasons behind the heterogeneities are asfollows: (i) ontologies evolve over a long period of time, (ii) users work in isolation and autonomously, and (iii) ontologies grow according to organizational requirements. Heterogeneities make semantic matching difficult if not impossible (Giunchiglia and Yatskevich, 2004; Ontology, 2004). To match/map the hierarchical ontologies, the context of concepts in ontologies is required. Because hierarchical ontologies are light-weight, it is therefore essential to explore and identify the context of the concepts in the ontologies.

Existing ontology-matching techniques are usually classified into two categories: (i) element level and (ii) structure level (Shavaiko and Euzenat, 2005; Kalfoglou and Schorlemmer, 2003). Element-level matching techniques handle ontology entities and their instances in isolation from their relationships with other entities or their instances. They apply very basic matching approaches, such as string-based, language-based and constraint-based. Some element-level techniques use external resources, such as WORDNET[5], to know the context of the elements; however, it might not be sufficient to capture the context with only external resources without looking into the ontology structure. These approaches are a pre-requisite of every matching technique. On the other hand, structure-level matching techniques find that mapping on the basis of relationships exist among entities and/or their instances. These techniques check hierarchical positions and child or leaf node similarities between the ontologies to determine the context (Shavaiko and Euzenat, 2005; Kalfoglou and Schorlemmer, 2003).

In this research, we found that the existing ontology-matching techniques, especially the structure-level techniques, are not sufficient to capture the context of the concepts in the matching of informal hierarchical ontologies. The main reasons behind their deficiency in matching the informal hierarchical ontologies are as follows: (i) the labels of some concepts are meaningless (i.e., undefined), (ii) hierarchical positions (i.e., levels) are not the same, (iii) a single concept can be represented with multiple concepts, and (iv) immediate parent concepts are not always predictive in these ontologies. These identified structural heterogeneities are elaborated in examples in the next section. We have designed rules to resolve the identified heterogeneities in matching informal hierarchical ontologies, and we implemented them in a prototype system. These rules can be used as an extension layer to the existing ontology-matching systems, such as (Giunchiglia et al., 2012; Jian et al., 2005; Ehrig and Sure, 2005). The proposed system was compared with existing open-source ontology matching systems: FOAM[6] (Ehrig and Sure, 2005) and Falcon[7] (Jian et al., 2005; Ehrig and Sure, 2005), in terms of the precision, recall and interpolated precision (Salton et al., 1986). Data sets of the Web directory *Dmoz* and the Yahoo directory, and the subject classification schemes, ACM Computing Classification System (ACM CCS) and Mathematics Subject Classification (MSC), were used for evaluation. The evaluation results show a significant improvement in the proposed system over the existing matching systems in the case of the identified heterogeneities.

The remainder of this paper is organized as follows: Section 2 identifies structural heterogeneities that exist in hierarchical ontologies. Related work is presented in Section 3. Section 4 discusses the proposed ontology-matching technique. Section 5 gives details on the evaluation and comparison of the results with the existing ontology-matching systems. Section 6 concludes the paper and identifies future directions.

2. Identified structural heterogeneities

Informal hierarchical ontologies do not strictly follow inheritance in their sub-concepts, which can lead to structural heterogeneities. We have identified the following structural heterogeneities in matching the hierarchical ontologies.

2.1. Meaningless labels

Each concept has a label that expresses its meaning, but the label sometimes is arbitrary and has no explicit meaning in

[1] http://googledirectory.com/ [July 22, 2009].
[2] http://dir.yahoo.com/ [July 22, 2009].
[3] http://www.acm.org/about/class/1998 [July 22, 2009].
[4] http://www.ams.org/mathscient/msc/msc.html [July 22, 2009].
[5] http://wordnet.princeton.edu/ [March28, 2013].
[6] http://www.aifb.kit.edu/ [March 28, 2013].
[7] http://ws.nju.edu.cn/falcon-ao/ [March 28, 2013].

any language, e.g., English. For example, *K-12* is a label of a sub-concept of *Education* in the Yahoo directory, but it has no explicit meaning.

2.2. Structural inconsistency

A concept in a source ontology has a different hierarchical position in the target ontology. In other words, a sub-concept of a concept can be the super-concept of the respective concept in the target ontology. For example, *News & Media* is a sub-concept of Sports in the Dmoz directory, as shown in Fig. 1. The same concept, *News & Media,* is the super-concept of *Sports* in the Yahoo directory.

2.3. Structural polysemy

Structural inconsistency makes it difficult to determine the actual facet of a concept through its immediate super-concept in hierarchical ontologies. For example, *Colleges & Universities* is a sub-concept of *News & Media* in the Dmoz directory, while it is a sub-concept of *Sports* in the Yahoo directory, as shown in Fig. 1. The concept *Colleges & Universities* in both ontologies is not the same with reference to its immediate super-concepts; however, the concepts are similar with respect to their context.

2.4. Multi-facet concepts

A concept or sub-concept of multiple concepts in a source ontology can be a sub-concept of a single concept in the target ontology. A concept that is a sub-concept of multiple concepts has more facets than the concept that is a sub-concept of a single concept. In other words, multi-facet concepts are more specialized, while single-facet concepts are more general. For example, *Baseball* is a sub-concept of *Colleges & Universities*, *Radio* and *Magazine and E-zines* in the Dmoz directory, as shown in Fig. 2. The same concept is a sub-concept of only one concept, i.e., Magazine, in the Yahoo directory. The concept *Baseball* in the Dmoz directory is more specialized than the concept *Baseball* in the Yahoo directory.

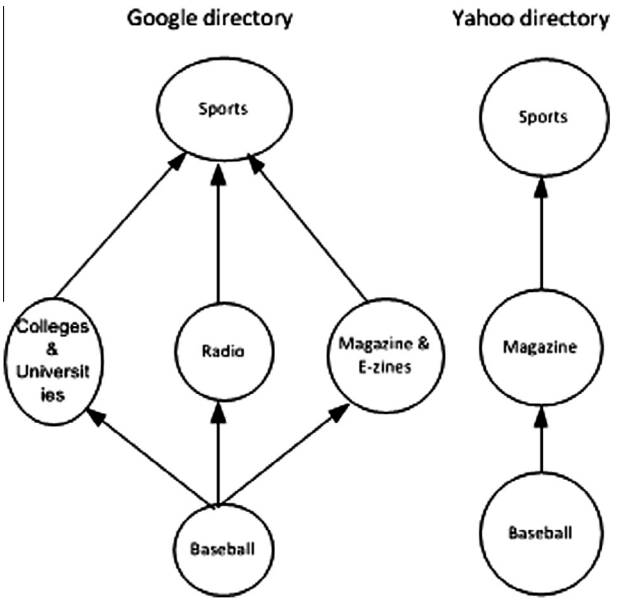

Figure 2 Multi-facet concepts in Web directories.

2.5. Synonym

The same concepts are labeled linguistically with different words (Khan and Mustafa, 2013). For example, linguistically, *Sports* and *Games* are two different things, but they are similar concepts with different labels.

2.6. Splitting context

The knowledge held in a concept of a source ontology can be scattered in multiple concepts in the target ontology. The

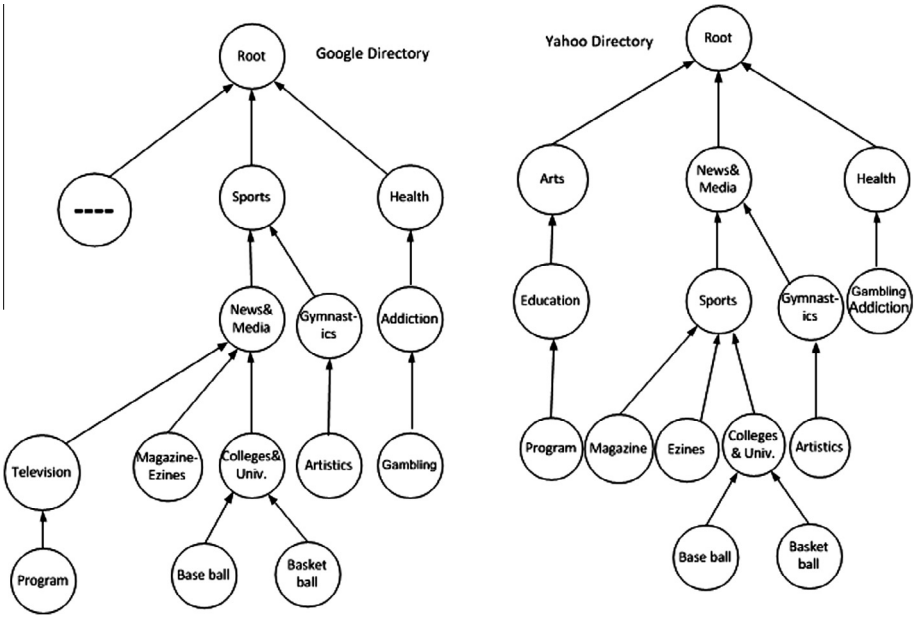

Figure 1 Snippets of Web directories.

multiple concepts can be located at a single level or at multiple levels. In other words, the concept of a source ontology can match to more than one (many) concept in the target ontology. For example, the *Gambling addiction* in the Yahoo directory is broken into two concepts at two levels, *Gambling* and *Addiction,* in the Dmoz directory, as shown in Fig. 1. Similarly, the concept *Magazine & Ezine* in the Dmoz directory is broken into two concepts: *Magazine* and *Ezine,* in the Yahoo directory at the same level, as shown in Fig. 1.

2.7. Implicit labels

Some similar concepts are labeled differently, and it is not possible to match them through their synonyms. However, they can be matched through their respective super-concepts and sub- concepts.

For example, the concepts *Gymnastics* and *Acrobats* are not synonym concepts, but we can depict their similarity on the basis of their super-concepts and sub-concepts context, as shown in Fig. 3.

In this paper, we propose rules that handle the heterogeneities and inconsistencies that are found in matching informal hierarchical ontologies that were previously mentioned.

3. Related work

Various ontology-matching systems and algorithms have been proposed since the last decade. An overview can be found in Shvaiko and Euzenat (2013, 2005), Ehrig and Sure (2004), Giunchiglia and Shvaiko (2003), Kalfoglou and Schorlemmer (2003). There is no single matcher that clearly dominates others. Often, they perform well in some cases and not very well in some of the other cases. Many systems have focused on combining and extending the known methods (Shvaiko and Euzenat, 2013). In this paper, we will discuss the identified heterogeneity issues in hierarchical ontologies with reference to the existing ontology-matching systems and algorithms.

FOAM (Ehrig and Sure, 2005), which was developed by the University of Karlsruhe, is an ontology alignment framework that is intended to fully or semi-automatically align two or more OWL ontologies. FOAM combines a rule-based approach and a machine learning approach. First, it considers the similarity of the individual entities (concepts, relations, and instances). As a result, it returns pairs of aligned entities. FOAM also provides a mechanism that allows users to set the parameters for a specific alignment task and select the alignment when doubtable alignments are produced. FOAM applies an iterative process and expands the mapping through the aggregation of previously estimated similarities.

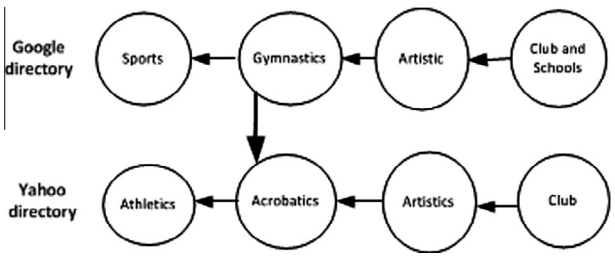

Figure 3 Super-sub context in Web directories.

Falcon-AO (Jian et al., 2005), which was developed by the South East University of China, is an automatic ontology-matching tool. There were two alignment strategies in Falcon-AO, LMO, and GMO (Hu et al., 2005). LMO is a matcher that is based on linguistic matching for ontologies, and GMO is a matcher that is based on graph matching for ontologies. Falcon-AO latest version (Hu et al., 2008) operates in three phases, to address large ontologies. It partitions entities of the input ontologies into sets of clusters and constructs blocks that are matched based on pre-calculated anchors. There are two alignment strategies in the new Falcon-AO, V-Doc (a linguistic matcher) and GMO (an iterative structural matcher).

S-Match (Giunchiglia et al., 2004, 2006, 2012; Shvaiko et al., 2010) is an algorithm and tool that was developed by the University of Trento. S-Match takes two trees as input, and for any pair of nodes from the two trees, it computes the strongest semantic relation holding between the concepts of the two nodes. To accomplish this task, it uses lexical techniques, background knowledge in the form of relations between synsets in WordNet, and the structure of the tree. S-Match is restricted to tree-like structures that are used for classification purposes.

CTXMatch (Magnini et al., 2004), QOM (Ehrig and Staab, 2004), and COMA (Do and Rahm, 2002) resolve structural polysemy of concepts only through their immediate super-concepts. However, their approach to structural polysemy is not as successful in informal hierarchical ontologies, as explained in the structural polysemy description in the previous section. Moreover, the existing matching systems do not differentiate between the matching of *multi-facet* concepts and *single-facet* concepts and handle them equally. This approach reduces the precision of the similarity of concepts in their approach in terms of concept specialization. Similarly meaningless labels that split context and super-sub context are not the focus of the existing systems.

4. Proposed matching technique

To match the concepts of the source and target ontologies, we have proposed and designed rules that recognize similarity between concepts from the structural representation of the informal hierarchical ontologies. These rules are described in detail as follows:

4.1. Identification of meaningless labels

Labels of concepts can usually be divided into two groups: (i) defined compound-word labels and (ii) meaningless (undefined) compound-word labels. We define each of these groups concisely, as follows:

Defined compound-word label: This type of label contains at least two adjacent words and has an explicit meaning in the WORDNET[8]. For example, *Academic Department* is a defined compound-word label in WORDNET, and its synonyms can be identified from WORDNET.

Undefined compound-word label: This type of label consists of two or more adjacent words and has no explicit meaning in the WORDNET. For example, *Laser Game* is not available in WORDNET; thus, we can consider it to be an undefined (i.e.,

[8] http://wordnet.princeton.edu/ [July 22, 2009].

Table 1 Rules for undefined compound word labels.

Punctuation marks	Replacement	Example	Rule
Space	Conjunction	Laser Game	Laser ∧ Game
Commas	Disjunction	Softball, Fast Pitch	Softball ∨ Fast pitch
And	Disjunction	Arts and Humanities	Arts ∨ Humanities
Or	Disjunction	Infinite group or finite group	Infinite group ∨ finite group
Preposition	Conjunction	Theory of Data	Theory ∧ Data

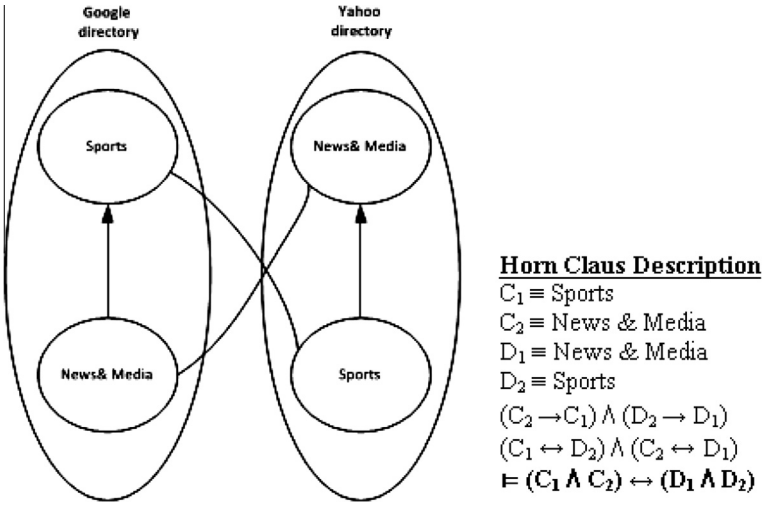

Figure 4 Structural inconsistency.

meaningless) label. These compound words could contain a space, comma, 'or', 'and', and other propositions. Our designed rules for identifying the conjectures of the concepts are shown in Table 1. For example, the label *Laser Game* must be treated as *Laser* ^ *Game*. In other words, the label *Laser Game* must be matched with a label that has both the words *Laser* and *Game*. Similarly, the label *Arts and Humanities* can be matched with either a label *Arts* ∨ *Humanities*.

4.2. Resolving structural inconsistency

The S-Match (Giunchiglia et al., 2004) approach to structural inconsistency cannot resolve this problem in informal hierarchical ontologies because sub-concepts do not essentially inherit all of the properties of their super-concepts. We propose the matching of the structure of the concepts, i.e., the composition of the concepts, instead of matching individual concepts in the case of structural inconsistency in informal hierarchical ontologies. Mathematically, we describe the rule as follows:

$$If((A \subseteq A') \in \alpha) \wedge ((A' \subseteq A) \in \beta) \ then \ (A, A') \equiv (A', A)$$

where α and β are two hierarchies, A and A' are concepts, and $\equiv$ denotes the synonym relation. The Horn clause representation of this rule is as follows:

$$((C_2(x) \rightarrow C_1(x)) \wedge (D_2(x) \rightarrow D_1(x))) \wedge ((C_1(x) \leftrightarrow D_2(x)) \wedge (C_2(x) \leftrightarrow D_1(x))) \models ((C_2(x) \wedge C_1(x)) \leftrightarrow (D_2(x) \wedge D_1(x)))$$

Example 1. In the Dmoz directory, Sport and News & Media is equivalent to News & Media and Sports in the Yahoo directory, as shown in Fig. 4.

4.3. Resolving structural polysemy

We resolve structural polysemy through an immediate *super-structure* instead of through immediate super-concepts (i.e., parents) in informal hierarchical ontologies. Here, we introduce two terminologies, *super-structure* and *sub-structure*. A superstructure of a concept is composed of broader concepts up to *great-grand-parent,* and a substructure of a concept is composed of narrower concepts up to *great-grand-child*. For example, in Fig. 5, the concept *News & Media* is the super-concept of the concept *Colleges & Universities* in the Dmoz directory, while *Sports* is the super-concept of the same concept in the Yahoo directory. If we compare the immediate super-concepts of *Colleges & Universities* in both directories, then it would be considered to be different. However, if we consider the immediate super-structure, the concept in both directories is the same according to our proposed rule that resolves structural inconsistency, as mentioned in the previous subsection. The Horn clause representation of this rule is the following:

$$((C_3(x) \rightarrow C_2(x) \rightarrow C_1(x)) \wedge (D_3(x) \rightarrow D_2(x) \rightarrow D_1(x))) \wedge ((C_1(x) \leftrightarrow D_2(x)) \wedge (C_2(x) \leftrightarrow D_1(x))) \models ((C_1(x) \wedge C_2(x)) \leftrightarrow (D_1(x) \wedge D_2(x))) \models (C_3(x) \leftrightarrow D_3(x))$$

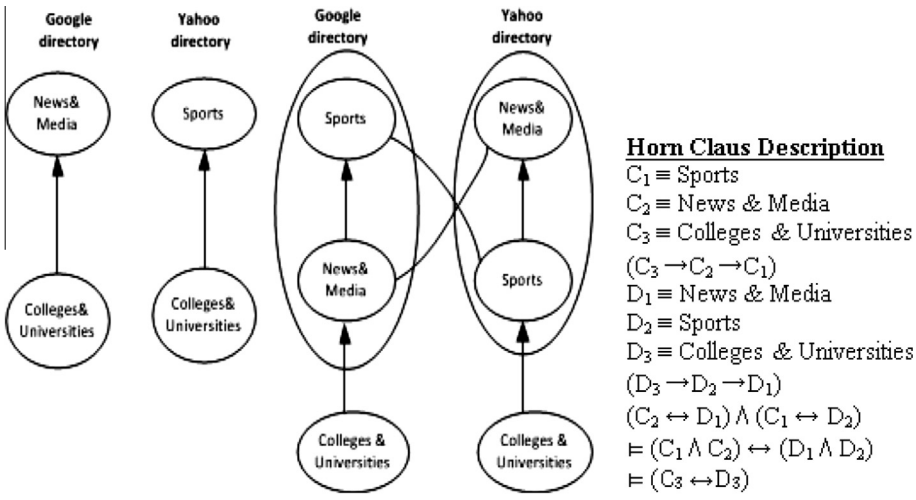

Figure 5 Structural polysemy illustration.

Example 2. This rule can be explained with the help of the Dmoz directory and the Yahoo directory, as shown in Fig. 5.

4.4. Matching multi-facet concepts

Multi-facet concepts are more specialized in comparison to single-facet concepts. In other words, multi-facet concepts are more restricted concepts. Therefore, we consider a multi-facet concept to be a sub-concept of the single-facet concept in matching. The concept *Baseball* in the Dmoz directory is a sub-concept of three parent concepts, i.e., *College and Universities*, *Radio* and *Magazine and Ezine*. On the other hand, the concept *Baseball* in the Yahoo directory has only one parent concept and is more generalized than the previous concept, as shown in Fig. 2. Here, we must define a function to assist the proposed rule.

Label: label(arg_1, [arg_2]) represents the label of a concept. The first argument represents the name of a concept, and the second argument is optional and represents the label of the first one.

The Horn clause representation of this rule is as follows:

$$(D_1(\varkappa) \rightarrow C_n(\varkappa) \wedge C_{n-1}(\varkappa) \wedge \cdots \wedge C_1(\varkappa)) \wedge (D_2(\varkappa) \rightarrow C_n(\varkappa) \vee C_{n-1}(\varkappa) \vee \cdots \vee C_1(\varkappa)) \wedge (label(D_1, y) \leftrightarrow (label(D_2, y))) \models D_2 \subseteq D_1 \models (D_1(\varkappa) \leftrightarrow D_2(\varkappa))$$

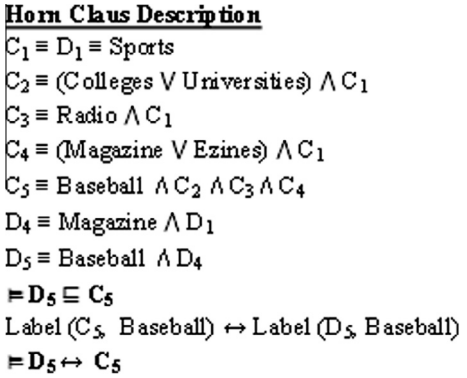

Figure 6 Multi-facet concepts matching illustration.

Example 3. This rule can be explained with help of Fig. 2 and is shown in Fig. 6.

4.5. Resolving the splitting context

In the case of a splitting context, the matching (target) concepts of a concept can be broken down either at a single level or at different levels in an informal hierarchical ontology. When the matching concepts of a concept are scattered at different levels, then those matching concepts are sub-concepts and super-concepts of one another. In this case, the conjunction (AND) of the matching concepts shall be similar to the source concept. Therefore, a source concept *Gambling Addiction* is equal to the target concepts *Gambling* $\wedge$ *Addiction,* as shown in Fig. 7.

Similarly, when the matching concepts of a concept are scattered at single levels, then these matching concepts are siblings of one another. In this case, the disjunction (OR) of the matching concepts shall be similar to the source concept. Therefore, the source concept *Magazine and Ezine* is equal to the concepts *Magazine* $\vee$ *Ezine,* as shown in Fig. 8.

The following rules in Equations 1 and 2 show resolving the splitting context on different levels and a single level, respectively, with the horn clause. We defined another function that is used in the proposed rule.

Concatenation: concate(arg_1, arg_2) represents a typical string concatenation function that is applied to the labels of the two concepts that are given as arguments.

$$(C_2(\varkappa) \rightarrow C_1(\varkappa)) \wedge label(C_1, a) \wedge label(C_2, b) \wedge label(D_1, z) \wedge label(D_1) \leftrightarrow concate(label(C_2), label(C_1)) \models ((C_1(\varkappa) \wedge C_2(\varkappa)) \leftrightarrow D_1(y)) \quad (1)$$

$$((C_1(\varkappa) \rightarrow D(z)) \wedge (C_2(y) \rightarrow D(z))) \models (C_1(\varkappa) \vee C_2(y)) \rightarrow (D(z)) \quad (2)$$

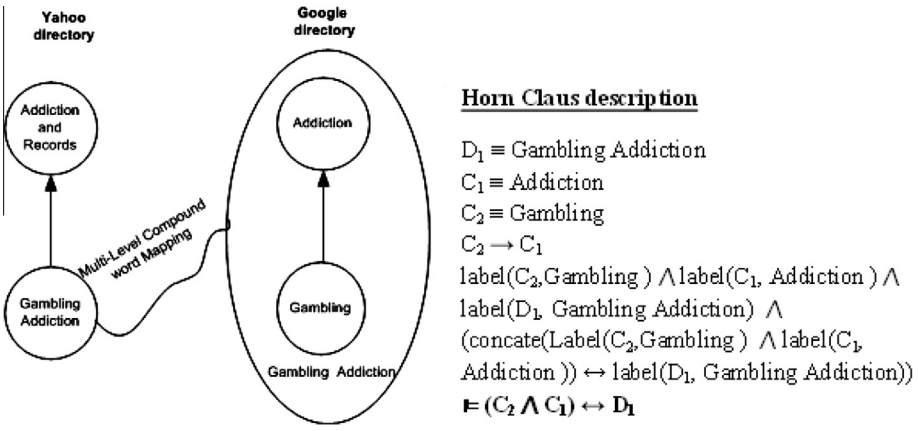

Figure 7 Splitting the context on different levels.

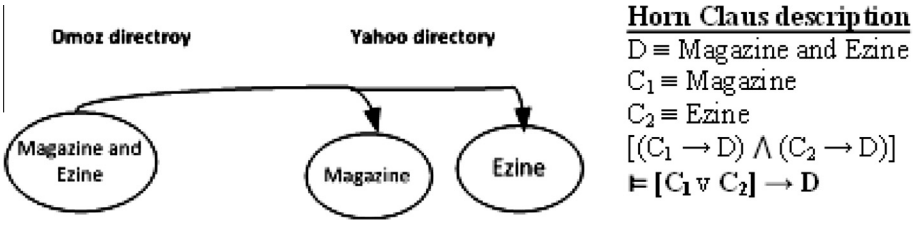

Figure 8 Splitting context on a single level.

Example 4. These rules can be explained with the help of the Dmoz directory and the Yahoo directory, as shown in Figs. 7 and 8.

4.6. Interpreting implicit labels

Implicit labels of concepts in an informal hierarchical ontology can be interpreted through their immediate super-structure and sub-structure. For example, hierarchies of the Dmoz directory *sports* \ *Gymnastics* \ *Artistic* \ *ClubandSchools* and the Yahoo directory *Athletics* \ *Acrobatics* \ *Artistic* \ *Club* are shown in Fig. 3.

According to our proposed rule, an immediate super-concept of *Gymnastics* is *Sports,* and a sub-concept is *Artistic*, whereas a super-concept of *Acrobatic* is Athletics, and a sub-concept is *Artistic*. Because the immediate super-structure and sub-structure of the concepts *Gymnastics* and *Acrobatic* are similar, the concepts are, therefore, equal. Next is the proposed rule with the horn clause, whose purpose is to handle the implicit labels.

$$\begin{aligned}((C_3(x) \rightarrow C_2(x) \rightarrow C_1(x)) \wedge (D_3(x) \rightarrow D_2(x) \\ \rightarrow D_1(x))) \wedge ((C_1(x) \rightarrow D_1(x)) \wedge (C_3(x) \\ \leftrightarrow D_3(x))) \wedge label(C_2, y) \wedge label(D_2, z) \wedge (y \equiv \neg z) \\ \models (C_2(x) \leftrightarrow D_2(x))\end{aligned}$$

Example 5. This rule can be explained with the help of the Dmoz and Yahoo directories, as shown in Fig. 9.

Figure 9 Implicit labels.

5. Implementation and evaluation

To evaluate and validate our system, ontologies are required. In the following subsections, we discuss the ontologies that were developed and their evaluation.

5.1. Data Set Specifications

We selected two different types of hierarchical ontologies: (i) web directories and (ii) subject classification schemes. In web directories, the Dmoz[9] directory is an extremely large directory that has almost 590,000 categories in its hierarchy. The Yahoo[10] directory has almost the same size. Thus, we considered only one main category, Sports, and its subsequent categories from both the Yahoo and Dmoz directories for this

[9] http://www.dmoz.org [March 28, 2013].
[10] http://dir.yahoo.com [July 22, 2009].

Table 2 Snippet of web directories mapping result.

Dmoz concepts	Yahoo concepts	Manual match	Proposed	Foam	Falcon
Health	Health	√	√	√	√
Addiction → Gabling	Gambling Addiction	√	√	X	X
Sport → Acrobat → Artistic	Sports → Gymnastics → Artistic	√	√	X	X
Sports → News & Media	News & Media → Sports	√	√	X	X
College and University	College and University	√	√	√	√
Columnist	Column and Columnists	√	√	X	√
Program	Program	X	X	√	√
Basket Ball	Archery	X	X	√	X
Disabled	Disabilities	√	X	X	√

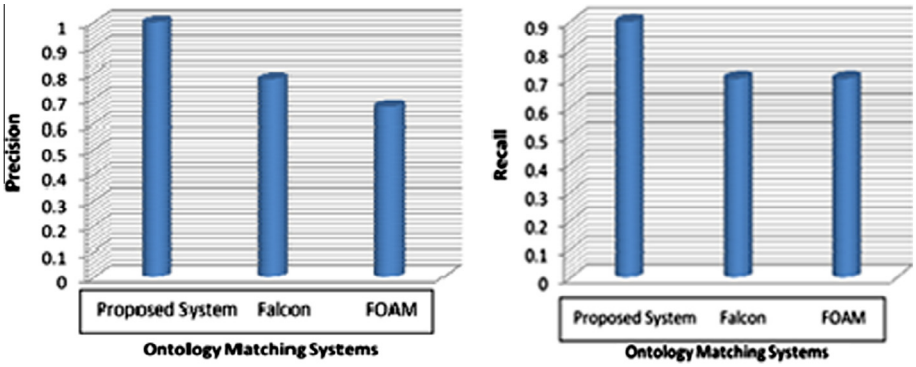

Figure 10 Web directories – precision and recall.

evaluation. On the other hand, the ACM Computing Classification System[11] and Mathematics Subject Classification (MSC) scheme[12] are both related to academia. ACM CCS classifies the Computer Science discipline, and MSC is used to classify Mathematics-related documents. Their subject domains overlap with each other; therefore, they were selected for the evaluation. Because these hierarchies are very large in size, we randomly selected their two main categories from each hierarchy, which are as follows:

- Computing Methodology & Application (MSC).
- Artificial Intelligence (MSC).
- Computing Methodology (ACM).
- Computer Application (ACM).

The reason for selecting a small portion of the available data sets for this research was that these data sets are not in ontology format but instead are either in text or XML, and they must be converted into a proper ontology language, either RDF or OWL. We developed a proper ontology in RDF for each data set in the Java language using Protege[13].

5.2. System architecture

The proposed system has been implemented in the Java language. The system architecture of the system consists of three components:

- Linguistic-analysis service uses (i) Protege-OWL and (ii) Jena-OWL-Model to parse the input ontologies. These APIs are open source Java libraries for OWL and RDF, and they provide classes and methods to load and save OWL files and to query and manipulate OWL data models to perform reasoning. This service considers concepts as standalone objects irrespective of their positions in a hierarchy. Concepts are tokenized and lemmatized first and then classified as compound words or undefined compound words according to the defined rules.
- World-knowledge service uses the WORDNET linguistic resource to find relationships among the concepts, such as synonym, hypernym and hyponym.
- Context analysis service captures the context of concepts in hierarchical ontologies using the defined rules for matching.

5.3. Evaluation and results

We compared our proposed system with two existing ontology matching systems, namely, Falcon[14] (Jian et al., 2005) and FOAM[15] (Ehrig and Sure, 2005). Both of these systems are open-source applications, and their code was available. We downloaded their APIs and deployed them on a local machine for evaluation purposes. We executed all of the three systems (i.e., the proposed system and the downloaded systems) for evaluation on the data sets that were mentioned above. The evaluation criteria that were used for the system

[11] http://www.acm.org/about/class/1998 [July 22, 2009].
[12] http://www.ams.org/mathscinet/msc/msc.html [July 22, 2009].
[13] http://protege.stanford.edu/ [Oct. 10, 2013].
[14] http://ws.nju.edu.cn/faccom-ao/ [March 28, 2013].
[15] http://www.aifb.kit.edu/ [March 28, 2013].

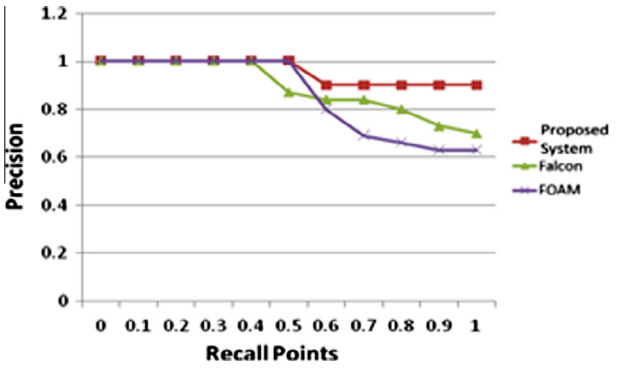

Figure 11 Web directories – interpolated precision.

were precision, recall (Hassanpour and Zahmatkesh, 2012) and interpolated precision (Salton et al., 1986). Precision can be seen to be a measure of exactness, whereas recall is a measure of completeness. Precision is the number of relevant concepts that can be retrieved by the system divided by the total number of retrieved concepts. Recall is the number of relevant mappings that can be retrieved by the system divided by the total number of relevant concepts (which should have been retrieved). To know the relevant concepts between the given ontologies, the ontologies were matched manually by domain experts. The interpolated precision combines both the precision and recall and measures the maximal precision above a certain recall level threshold.

5.3.1. Benchmark: web directories

Both the directories Dmoz and Yahoo were given as inputs to all of the three systems. A snippet of the result is shown in Table 2. In the table, the sub-concepts are represented with arrows (→), e.g., *Gambling* is a sub-concept of *Addiction*, mark (√) represents the matched concepts, and cross (X) shows the non-matched concepts. The concept *Program* in the Dmoz is not similar to *Program* in Yahoo in both the manual and the proposed matches because their super-structures are not similar (i.e., shown in Fig. 1), although their labels are the same. Fig. 10 shows the accuracy of our system (the proposed rules) in terms of the precision and recall. The precision of the proposed system is higher than Falcon and FOAM by 19% and 30%, respectively. Similarly, the recall is better by 27% than Falcon and FOAM. In Fig. 11, the graph illustrates the monotonically decreasing function of the interpolated precision at each recall point. The decrease in the interpolated precision in the case of Falcon and FOAM is sharper than in our proposed system.

5.3.2. Benchmark: subject classifications

Similarly, both classification schemes ACM CCS and MSC were inputs to all three systems. A snippet of the result is shown in Table 3. Fig. 12 shows the accuracy of our system in terms of the precision and recall. The precision of the proposed system is higher than that of Falcon and FOAM by 10% and 32%, respectively. Similarly, the recall is improved by 9% and 39% compared with Falcon and FOAM, respectively. In Fig. 13, the graph illustrates the monotonically

Table 3 Snippet of the classification schemes mapping results.

ACM concepts	MCS concepts	Manual match	Proposed	FOAM	Falcon
Artificial intelligence	Artificial intelligence	√	√	X	X
Computational geometry	Computational geometry	√	√	X	√
Computer aided design	Computer aided engineering	√	X	X	X
Image processing	Image processing and computer vision	√	√	√	X
Information system	Information systems	√	√	√	√
Problem solving, control methods and search	Problem solving	√	X	√	√
Robotics	Robotics	√	√	√	√
Simulation	Simulation and modeling	√	√	√	X
Knowledge representation formalism	Knowledge representation	√	√	√	X

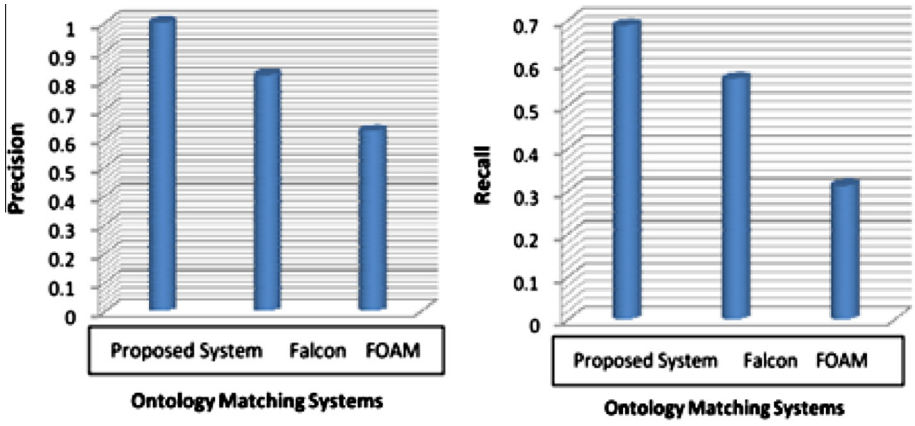

Figure 12 Subject classification – precision and recall.

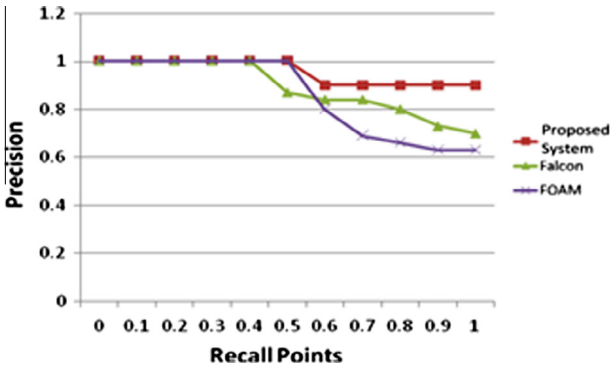

Figure 13 Subject classification – interpolated precision.

decreasing function of the interpolated precision at each recall point. The decrease in the interpolated precision in the cases of Falcon and FOAM is more severe compared with our proposed system.

6. Conclusions and future directions

Matching is vital for enabling the interoperability among different ontologies in semantic web applications. The semantic matching of ontologies is a labor-intensive and error-prone process in integrating autonomous data sources. Hierarchical ontologies are represented in the form of a directed acyclic graph, where a node models a concept and its label codifies the meaning of the concept. The hierarchical ontologies are light-weight and classify concepts at each level. They proceed from generalized to specialized concepts and are usually divided into the following categories: *formal* and *informal* hierarchical ontologies. The first category strictly implements inheritance into sub-classes, while the latter category does not strictly follow inheritance in its subclasses. In the same subject domain, different hierarchical ontologies can have different classifications of the concepts, which are called structural heterogeneities. Reasons behind the heterogeneities are as follows: (i) ontologies evolve over a long period of time, (ii) users work in isolation and autonomously, and (iii) ontologies grow according to organizational requirements. Heterogeneities make semantic matching difficult. To match hierarchical ontologies, the context of the concepts in the ontologies is required. Because hierarchical ontologies are light-weight, it is therefore essential to explore and identify the context of the concepts in the ontologies.

In this research, we found that the existing ontology-matching techniques, especially at the structure level, are not adequate to capture the context of the concepts when matching informal hierarchical ontologies. The main reasons behind their deficiency in matching informal hierarchical ontologies are as follows: (i) the labels of some of the concepts are meaningless (i.e., undefined), (ii) hierarchical positions (i.e., levels) are not the same, (iii) a single concept can be represented with multiple concepts, and (iv) immediate parent concepts are not always predictive in these ontologies. We have designed rules to resolve the identified heterogeneities in matching informal hierarchical ontologies and implemented them in a prototype system. The proposed system was compared with existing open-source ontology-matching systems, namely, FOAM and Falcon, in terms of the precision, recall and interpolated precision. Data sets of the Web directory Dmoz and the Yahoo directory and the subject classification schemes ACM Computing Classification System (ACM CCS) and Mathematics Subject Classification (MSC) were used for evaluation. The evaluation results show a significant improvement in the proposed system over the existing matching systems in the case of identified heterogeneities. A future research step can be to find the structure patterns that are equally applicable for all types of hierarchical ontologies.

References

Bellahsene, Angela, Bonifati, Erhard, Rahm (Eds.), 2011. Schema matching and mapping. Springer.

Do, H., Rahm, E., 2002. Coma – a system for flexible combination of schema matching approaches. In Proceedings of the Very Large Data Bases Conference (VLDB), Hong Kong, China, pp. 610–621.

Ehrig, M., Staab, S., 2004. Qom: quick ontology mapping. In Proceedings of the International Semantic Web Conference (ISWC), Arlington, USA, pp. 683–697.

Ehrig, M., Sure, Y., 2004. Ontology mapping – an integrated approach. In Proceedings of the European Semantic Web Symposium (ESWS), Heraklion, Crete, Greece, pp. 76–91.

Ehrig, M., Sure, Y., 2005. Foam – framework for ontology alignment and mapping; results of the ontology alignment initiative. In Proceedings of the K-CAP 2005 Workshop on Integrating Ontologies, 156, Banff, Canada, October, pp. 72–76.

Euzenat, Jerome, Euzenat, Jerome, Shvaiko, Pavel, 2007. Ontology Matching. Springer.

Giunchiglia, F., Shvaiko, P., 2003. Semantic matching. Knowl. Eng. Rev. J. (KER) 3, 265–280.

Giunchiglia, F., Yatskevich, M., 2004. Element level semantic matching. In Proceedings of meaning coordination and negotiation workshop at ISWC, Hiroshima, Japan, pp. 102–109.

Giunchiglia, F., Shvaiko, P., Yatskevich, M., 2004. S-match: an algorithm and an implementation of semantic matching. In: Bussler, Christoph, Davies, John, Fensel, Dieter, Studer, Rudi (Eds.), The Semantic Web: Research and Applications, volume 3053 of Lecture Notes in Computer Science. Springer, Berlin/Heidelberg, pp. 61–75.

Giunchiglia, F., Shvaiko, P., Yatskevich, M., 2005. Semantic schema matching. Technical report, University of Trento, Povo, Trento, Italy, 2005. Technical, Report DIT-05-014.

Giunchiglia, Fausto, Shvaiko, Pavel, Yatskevich, Mikalai, 2006. Discovering missing background knowledge in ontology matching. In: Proceedings of the 2006 conference on ECAI 2006: 17th European Conference on Artificial Intelligence August 29-September 1, 2006. IOS Press, Riva del Garda, Italy, pp. 382–386.

Giunchiglia, F., Autayeu, A., Pane, J., 2012. S-match: an open source framework for matching lightweight ontologies. Semantic Web 3 (3), 307–317.

Gomez-Perez, A., Corcho, O., Fernandez-Lopez, M., 2004. Ontological Engineering, first ed. Springer.

Halevy, A., 2005. Why your data won't mix: semantic heterogeneity. Queue 3 (8), 50–58.

Halevy, A., Rajaraman, A., Ordille, J., 2006. Data integration: the teenage years. In Proceedings of the 32nd international conference on Very large data bases, Seoul, Korea, VLDB Endowment, ACM, pp. 9–16.

Hassanpour, Hamid, Zahmatkesh, Farzaneh, 2012. An adaptive meta-search engine considering the users field of interest. J. King Saud Univ. Comput. Inf. Sci. 24 (1), 71–81.

Hu, W., Jian, N., Qu, Y., Wang, Y., 2005. Gmo: a graph matching for ontologies. In Proceedings of the K-CAP 2005 Workshop on Integrating Ontologies, Banff, Alberta, Canada, pp. 43–50.

Hu, Wei, Qu, Yuzhong, Cheng, Gong, 2008. Matching large ontologies: a divide-and-conquer approach. Data Knowl. Eng. 67 (1), 140–160.

Jian, N., Hu, W., Cheng, G., Qu, Y., 2005. Falcon-ao: aligning ontologies with falcon. In Proceedings of K-Cap 2005 Workshop on Integrating Ontologies, Banff, Alberta, Canada, pp. 85–91.

Kalfoglou, Y., Schorlemmer, M., 2003. Ontology mapping: the state of the art. Knowl. Eng. Rev. J. (KER) 18 (1), 1–31.

Khan, Sharifullah, Mustafa, Jibran, 2013. Effective semantic search using thematic similarity. J. King Saud Univ. Comput. Inf. Sci. 0.

Magnini, Speranza, M., Girardi, C., 2004. A semantic-based approach to interoperability of classification hierarchies: evaluation of linguistic techniques. In Proceedings of the 20th international conference on Computational Linguistics, Geneva, Switzerland, p. 23–27.

Ontology interoperability: state of the art report. Technical report, WP8 ST3 Deliverable, IST-508011, http://twiki.di.uniroma1.it/pub/estrinfo/material/, 2004. Date visited: July 22, 2009.

Pivovarov, Rimma, Elhadad, NoA©mie, 2012. A hybrid knowledge-based and data-driven approach to identifying semantically similar concepts. J. Biomed. Inf. 45 (3), 471–481.

Salton, G., 1986. Recent trends in automatic information retrieval. In Proceedings of the 9th annual international ACM SIGIR conference on Research and development in information retrieval, Italy, pp. 1–10.

Shavaiko, P., Euzenat, J., 2005. A survey of schema-based matching approaches. J. Data Semant. (JoDS) 4, 146–171.

Shvaiko, Pavel, Euzenat, Jerome, 2013. Ontology matching: state of the art and future challenges. IEEE Trans. Knowl. Data Eng. 25 (1), 158–176.

Shvaiko, Pavel, Giunchiglia, Fausto, Yatskevich, Mikalai, 2010. Semantic matching with s-match. In: Semantic Web Information Management. Springer, pp. 183–202.

Zuber, V.S., Faltings, B., 2007. Oss: A semantic similarity function based on hierarchical ontologies. In Proceedings of 20th International Joint Conference on Artificial Intelligence (IJCAI), Hyderabad, India, pp. 551–556.