Towards Domain Ontology Interoperability Measurement

Hussein Sseggujja[1] and Ali Selamat[1,2](✉)

[1] UTM-IRDA Digital Media Centre, UTM and Faculty of Computing, Universiti Teknologi Malaysia, 81310 Johor Bahru, Johor, Malaysia
segujah@yahoo.com

[2] Software Engineering Research Group (SERG), Faculty of Computing, Universiti Teknologi Malaysia, 81310 Johor Bahru, Johor, Malaysia
aselamat@utm.my

Abstract. Ontologies are reliable interoperability support components between information systems. However the need to make ontologies themselves interoperable to a measurable degree remains a challenge due to the semantic heteroginity problem. This paper specifically looks at domain ontologies and how to measure the interoperability degree between them to establish the extent to which they can replace each other. Different interoperability operations,semantic distance measures and lexical similarity between ontologies are dicussed. A method based on model management theory with algebraic operations such as match on the ontology models is proposed to measure lexical and structural dimensions of domain ontologies and give a value for their degree of interoperability. An example of how to compute the degree of interoperability between two domain ontologies using the proposed approach is given with an explanation of how the identified gaps can be addressed.

Keywords: Domain · Ontology · Interoperability · Similarity · Measure

1 Introduction

Current development trends in software engineering have created increasing demand for interoperability between the different systems being engineered. Ontologies have been described widely as a representation of common concepts in a given domain and their relationships. This characteristic enables ontologies to be used as an interoperability component for the integration between systems. An ontology is an "*explicit specification of a conceptualization*" [1]. Therefore an ontology is a formal representation of concepts and their relationship within a particular domain. A domain is *"An area of or field of specialization where human expertise is used, and a Knowledge-Based System application is proposed to be used within it."* [2].

Interoperability is referred to as the "capability to communicate, execute programs, or transfer data among various functional units in a manner that requires the user to have little or no knowledge of the unique characteristics of those units" [3]. The EIF (European Interoperability framework) [4] identifies three levels of interoperability: technical, semantic and organizational. This paper aims at semantic interoperability

H. Fujita and A. Selamat (Eds.): SoMeT 2014, CCIS 513, pp. 282–296, 2015.
DOI: 10.1007/978-3-319-17530-0_20

derived from use of ontologies. Among the different types of ontologies, this paper focuses on the domain ontologies which represent the meanings [5] of terminologies as they are perceived in given domain. Domain ontologies therefore offer a platform for interoperability that supports systems engineering benefits such as reusability, reliability, specification and also allow for better communication and cooperation between people and systems. It's also often that ontologies from the same domain are not interoperable due to different perceptions of the domain designers based on cultural background, ideology or because a different representation language was used to build the ontology. Therefore ontology design is subjective and parties may exploit different ontologies related to the same application domain differently thereby causing what is referred to as the semantic heterogeneity [6] problem. To achieve successful communication within heterogeneous environments where ontologies are used, it is necessary to bring them into a mutual agreement by establishing semantically related entities between the two ontologies. It is important to note here that ontology interoperability measurement sits at the intersection of three ontology interoperability areas; the interoperability algorithms or operations, the similarity measures and algebraic model operators.

The objectives of this paper are three. First is to compare and identify gaps in domain ontology interoperability approaches, algorithms and measures. The second is to propose a method for measuring the degree of interoperability between two domain ontologies and thirdly is to explain how the proposed method can be used to compute the interoperability degree. The proposed method utilises an algebraic model that represents the domain ontologies. The approach combines the model and the match operator to produce a measure of the degree of interoperability between two domain ontologies. The benefits of the proposed approach include providing knowledge about the structural and semantic heterogenity degree between two domain ontologies. This information enables system integrators and ontology designers to make better and informed choices between domain ontologies for reuse and other related purposes.

The paper is organized as follows. Section 2 presents a detailed review and analysis of the different domain ontology interoperability approaches. Section 3 covers analysis of different ontological similarity measures and model operators between ontologies. Section 4 presents the proposed improved method to measure the degree of interoperability between one domain ontology and another. An example from the geographical land coverage domain is given using the proposed method. Finally Sect. 5 gives the conclusion.

2 Approaches to Domain Ontology Interoperability

Due to the large variety of information sources, a single universal ontology cannot be built and systems will continue to use different ontologies even if they come from the same domain. Ontologies can interoperate only if correspondences between their elements have been identified through use of methodologies and tools that support the knowledge engineer in discovering semantic correspondences [7]. Therefore if the journey of making ontologies interoperable is still long, the stage of measuring the degree of their interoperability is even longer.

2.1 Operations for Domain Ontology Interoperability

This section covers different operations and approaches or algorithms used in each operation. There exists no clear standard for defining the terms matching, mapping, and alignment [8] hence what is given here includes the definitions adopted for this paper.

Ontology Alignment Operation. Given two ontologies, alignment means that for each entity (concept, relation, or instance) in the first ontology, we get an entity which has the same intended meaning, in the second ontology. The result of matching, called an alignment is a set of pairs of entities e, e' from two ontologies O and O' that are supposed to satisfy [9] a certain relation r with a certain confidence level n. Algorithms which implement the match operator can be generally classified [10] along the following dimensions; schema-based algorithms, instance-based algorithms, element-based and structure-based which compare structures of the ontology to determine the similarity. It is also possible to have combinations of different mechanisms within one algorithm.

Ontology Mapping/Matching Operation. Ontology mapping is a function between the ontologies whereas alignment merely identifies the relation between ontologies. It is a specification of the semantic overlap between two ontologies and consists of three main phases including discovery, representation and execution. Some authors consider ontology mapping as a directed version of an alignment while others like Ehrig and Staab [10] do not distinguish between mapping and alignment.

Ontology Merging/Integrating Operation. Ontology merging is the creation of one ontology from two or more source ontologies [11] and replace the original source ontologies. The outcome may be a new merged ontology that captures the original ontologies or just a 'view' (bridge ontology) that imports the original ontologies and specifies the correspondences using bridge axioms. In integration, one or more ontologies are reused for a new ontology keeping the original concepts unchanged although they can be extended. The most prominent integration approaches are union and intersection where either all entities or only those that have correspondences in both ontologies are taken.

Mediation Operation. Ontology mediation basically reconciles the differences between two or more heterogeneous ontologies. Ontology mediation enables reuse of data across applications on the semantic web and cooperation between different organizations.

Ontology Translation and Transformation Operations. Translation is restricted to data which may also include the syntax for example translating the ontology from RDF (S) to OWL. Transformation involves changing the structure of the ontology without altering its semantics (lossless) or by modifying it slightly for different new purposes.

2.2 Algorithms for Ontology Interoperability

This section outlines different solutions that address the ontology interoperability challenge. Different methods focus on various aspects of the interoperability problem. Some approaches enable interoperability at the language level to unify the specifications and then to compare the elements of the different ontologies. Table 1 gives an analysis of the

Table 1. Key features in Ontology interoperability algorithms

Algorithm	Operation	Basis	Input	Output	Solution
GLUE[12]	Alignment, Mapping	Schema concepts, instances	RDF(s) structures	Alignment proposals	Method and tool
Anchor Prompt[13]	Alignment, Merging	Schema structure	RDF(s) ontologies	Alignment proposals	Tool
QOM[14]	Alignment, Mapping	Schema Concepts	Structure ontology	Match proposals	Method and tool
MAFRA[15]	Mapping	Concepts, instances	Concepts	Alignments	Tool
PROMPT[16]	Mapping, Merging	Concept Schema	Structure	Alignment proposal	Tool
Prompt Diff[16]	Mapping, Merging	Schema concepts	Structure	Proposals for user	Tool
Chimaera[17]	Merging	Schema concepts	Structure	Interactive proposals	Tool
ONION[18]	Merging	Concepts, instances	Structure	Rules	Tool and method
Madhavan[11]	Mapping	concepts	Structures	Rules	Method
MOMIS-OIS[19]	Mapping, Merging	Schema concepts	Queries	Rules	Method, framework
Kiryakov[20]	Mapping	Concepts	Queries	Rules	Method
IFF[21]	Mapping	Instances, concepts	Ontology structures	Agreements, constraints	Method and Tool
HELIOS[22]	Merging, mapping	Schema concepts	Ontology structure	Rules	Method and Tool
ARTEMIS[23]	Mapping	Schema concepts	Schema structure	Class alignments	Tool and Method
HCONE[24]	Merging	Schema concepts	RDF(s) structure	Ontology	Tool
SHOE[12]	Merging	Schema concepts	Ontology	Tag rules	Tool
KRAFT[25]	Merging, Mapping	Schema structures	Ontology structure	Rules	Method and tool
ONTOMerge[11]	Merging, translation	Schema concepts	Ontology structure	Merged ontology	Online Tool
OLA[18]	Alignment	Concept labels	Ontology	alignments	Tool
COMA[26]	Alignment	Schema structures	schemas	Alignment proposals	Method, Tool
FCA-Merge[27]	Merging	Instances	Ontology structure	Merged ontology	Tool and method
FORM[28]	Alignment, mapping	Concepts, instances	Ontology structure	alignments	Method
SAMBO[29]	Alignment, Merging	Instances, concepts	Ontology structure	Merged ontology	Method
S-match[30]	Matching, Mapping	Concepts ,labels	Ontology structure	alignments	Method
RiMOM[31]	Matching	Concepts, instances	Ontology structure	Matches, Alignments	Mainly Framework
DELTA[32]	Mapping	Schema based	Schema	Alignments	Tool
MapOnto[33]	Mapping	Schema based	columns, concepts	rules	Method, tool
iMap[34]	Matching	Instance based	Schema elements	matches	Method
SEMINT[35]	Matching	Schema, instance	Schema structure	Rules	Method
ToMAS[36]	Mapping	Schema based	Schema structure	Mappings	Tool

key features from about 30 commonly used solutions for interoperability among ontologies in literature and the following issues are coming out clearly from the comparison;

- Alignment and merging operations are mainly interactive with the user and produce proposals for the user to adopt or neglect according to his or her experience in the domain concepts.
- Although some algorithms operate on instances in the ontology, the majority base their operations on concepts alone. This allows the operations to keep at the syntax level and metadata level which enables modelling functions for ontology management as mathematical models.
- Some mapping approaches produce mapping rules discovered during the process and these rules can be used for mapping between concepts of different ontologies.
- Although the algorithms cover different aspects of domain ontology interoperability, some of these algorithms are fully automated for their purpose while others are semi-automated or manual.
- Some algorithms take ontology structures as input as opposed to just concepts or instances. Transformations from one representation format to another in order to carry out the processing may affect the semantic level of the original ontologies.
- We have not found clear attempts to measure the degree of domain ontology interoperability, hence a gap that this paper responds to.

3 Measures for Domain Ontological Similarity

3.1 The Nature of Semantic Measures

Defining a measure involves [37] defining information sources, theoretical principles and the semantic class including semantic distance, semantic similarity and semantic relatedness. Mathematical analysis, domain-specific applications, and comparison by human judgments of similarity have been used for measures. The discussion in this section assumes an ontology in Fig. 1 that contains two concepts C1 and C2 for which the distance and corresponding similarity is to be assessed. The concept C3 represents a Least Common Subsumer (LCS) or ancestor of C1 and C2 in the ontology.

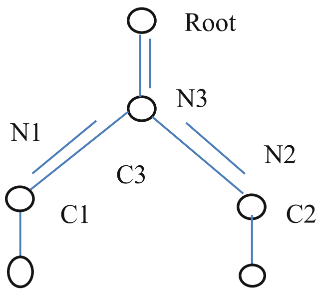

Fig. 1. Basic hierarchical ontology structure

Ontology measuring techniques are mainly classified into structure-based measures, feature-based measures, information content (IC) based measures and hybrid measures. Examples of these measures include; Tversky that looks at a concept in an ontology as an object with a set of features. A concept's similarity is determined by a respective set of ancestors X and Y. The measure is given by the formula below:

$$\frac{f(X \cap Y)}{f(X \cap Y) + \alpha f(X - Y) + \beta f(Y - X)} \tag{1}$$

The determination of α and β is based on the observation that similarity is not necessarily symmetric and gives different measure variations as shown in Table 2.

Table 2. Variations of Tversky similarity measure

α–value	β- value	Measure	Similarity function
1	1	Jaccard index	$\frac{f(X \cap Y)}{f(X \cup Y)}$
1/2	1/2	Dice coefficient	$\frac{2 * f(X \cap Y)}{f(X) + f(Y)}$
1	0	Degree of inclusion of X in Y	$\frac{f(X \cap Y)}{f(X)}$
0	1	Degree of inclusion of Y in X	$\frac{f(Y \cap X)}{f(Y)}$

The Table 3 gives an analytical comparison of the different common measures in literature.

The availability of many measures for semantic similarity raises a fundamental question: How well does a given measure capture the similarity between two concepts, set of concepts or ontology [49]. For information retrieval metrics, it is difficult to determine the evaluation value and measuring process quality independently. Hence need for compliance evaluations such as Precision, Recall, F-measure and performance measures that focus on speed, memory usage and the processing environment.

3.2 Model Management Operators for Ontology Interoperability

The common approach for interoperability computations [50] is to use ontology representation models identified by the root concept, set of reachable concepts through is-a relationships and built-in types of constraints such as the min and max cardinality. The model structure supports algebraic operations to create or delete a concept, read or write a property, and add or remove a relationship. Examples of the operators include;

1. Match – takes two models as input and returns a mapping between them. Figure 2 shows the details.
2. Compose – takes a mapping between models A and B and a mapping between models B and C, and returns a mapping between A and C (Fig. 3).

Table 3. Common similarity measures

Measure	Class / Type	Source	Equation	Description
Rada [38]	Structure-Edge counting	Ontology		Uses shortest distance between two concepts.
Wu and Palmer[39]	Structure –Edge counting	Ontology	$\frac{2*N3}{N1+N2+2*N3}$	Path-based distances to measure similarity.
Leacock and Chodorow [40]	Structure- Path - based	Ontology	$Max\ [\ -log(\frac{Np(c1,c2)}{2D})]$	Focuses on node counting
Resnik[41]	IC - Corpus	Corpus and ontology	$-\log p(c)$	Leaves out individual concepts information.
Lin[42]	IC- Corpus	Corpus	$\frac{2*IC(c3)}{IC(c1)+IC(c2)}$	Addresses the critism in Resnik
Jiang and Conrath[43]	IC	Ontology	IC(c1) +IC(c3) – 2 * IC(c3)	Integrates IC and path based measures.
Tversky [44]	Feature based	Ontology	$\frac{f(X\cap Y)}{f(X\cap Y)+\alpha f(X-Y)+\beta f(Y-X)}$	Uses ancestors and descendant
Al-Mubaid and Nguyen[45]	Node and path counting	Ontology	$log_2((len(c1,c2)-1)*(D-Depth(LCS(c1,c2)))+2)$	Uses distance and depth of LCS and overall ontology depth
Hirst and St Onge[46]	Edge counting and relations	Ontology relations	C-path length - k * number of changes in direction.	It categorises relations into 4 levels of strength.
Sussna[47]	Path and relations	Ontology relations	$max_r - \frac{max_r - min_r}{n_r(c_i)}$	Attaches to each possible link with concept.
Knappe[48]	Hybrid	Ontology	$p*\frac{\lvert Ans(c1)\cap Ans(c2)\rvert}{Ans(c1)} + (1-p)*\frac{\lvert Ans(c1)\cap Ans(c2)\rvert}{Ans(c2)}$	Considers multiple paths between concept generalisations.
Resnik[48]	IC	Corpus, Ontology	$-\log p(c)$	Focuses on content of shared parents
Shortest path[38]	Structure- Edge counting	ontology	2 * Max(C1,C2) – Shortest_Path	Mainly counts edges

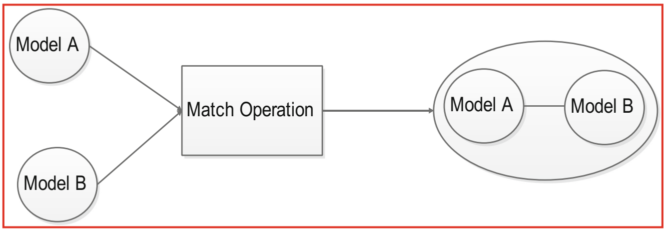

Fig. 2. Basic match operator

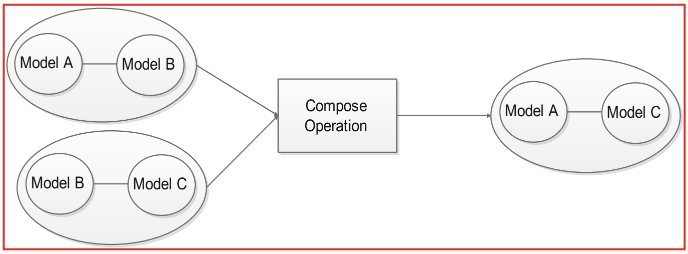

Fig. 3. Compose mapping operator

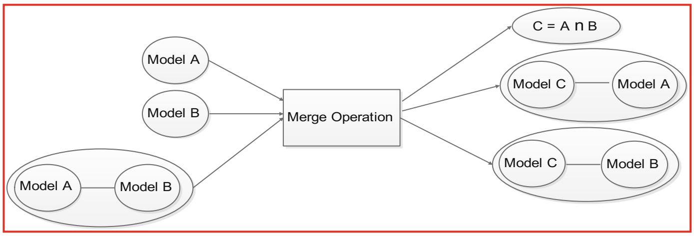

Fig. 4. Merge operator

3. Merge – takes two models A and B and a mapping between them, and returns C as a union of A and B along with mappings between C and A, and C and B (Fig. 4).

Although substantial work on Match has been done, Merge, Compose, and ModelGen are less developed.

4 A Domain Ontology Interoperability Degree Measurement Method

The *Match* operator is widely covered in model management literature. It takes two models and returns two sets of tuples for the similarity and generalization relationships that exist between the concepts of the two ontologies. The operator proposed by Bernstein [50] offers a relatively clearer approach to measure the degree of interoperability between ontologies.

4.1 Proposed Approach Methodology

In order to use the match operator to measure the degree of interoperability between domain ontologies we constrain the proposed approach within the following properties of the ontologies.

1. The ontologies measured share a common domain and are light weight.
2. The ontology model can be manipulated using model management operations.
3. One of the ontologies is taken as the base ontology for the comparison.
4. The two ontologies are made up of a set of concepts that have relations in between them and can allow for similarity, generalisations and subsumptions.
5. The ontology concepts are organised in a single hierarchy anchored at the root down to specialisations.

To compare the ontologies using the match operator we take the most specialized concepts (all leaf-nodes of the ontology tree) from the first or base ontology and attempt to find similar concepts in the second ontology. Given a concept in the base ontology, the Match operator follows these steps:

1. The *Sim* (similarity) process identifies similar concepts from both ontologies based on their parts, attributes or relationships. Given concept c1 of ontology O1, *Sim* finds a similar concept c2 from O2. This is done using the different similarity measures such as Rada or Wu and Palmer outlined in Sect. 3.
2. If step (1) fails to produce a satisfactory result, the *Gen* process identifies the concept in the second ontology that is most closely related to c1 by generalization. The *Gen* process follows three steps;
   (i) Given a concept c1 of O1 that has no similar match in O2, locate concept d1 that is the generalization of c1.
   (ii) Find a similar concept to d1 in O2, d2. Use *Sim* process to match.
   (iii) If a similar concept (d2) is found, then d2 is a generalization of c1 in O2.
3. The Match function produces two subsets as a result:
   - *Sim* (O1,O2): The similarity subset *Sim* (O1,O2) of O1 in relation to O2 is the set of all tuples <c1, c2> such that the concept c2 in O2 is similar to the concept c1 in O1.
   - *Gen* (O1,O2): The generalization subset *Gen*(O1,O2) of O1 in relation to O2 is the set of all tuples <c1, c2> such that the concept c2 in O2 is a generalization of the concept c1 in O1. Two main forms of the degree of the interoperability between two domain ontologies O1 and O2 are:-

   1. **Full interoperability:** If and only if the similarity set *Sim*(O1,O2) contains all concepts in O1.
   2. **Partial Interoperability:** Where the generalisation set *Gen*(O1,O2) is not an empty set such that atleast some concepts in O1 have been matched with concepts in O2. The degree of interoperability IntDeg is given formula (2).

$$\mathrm{IntDeg} = \left[f + \frac{\sum_{i=1}^{n} \min\left(l_2^i, l_1^i\right)}{\sum_{i=1}^{n} l_1^i} \right] \Big/ 2 \qquad (2)$$

   Where

   - *f* is the fraction of concepts of O1 that are contained in *Sim*(O1,O2).
   - the second part is the degree of generalisation derived from generalisation.

In formula (2) above l_1^i is the depth of the ith concept of ontology O1 and l_2^i is the depth of the corresponding concept in ontology O2 as given by the tuples in *Gen*(O1, O2). The degree of generalisation is obtained by comparing the depth of the tree associated with the concepts in the generalization subset *Gen*(O1,O2).The formula is based on the idea that the greater the difference between the depth levels of the two concepts, the smaller the degree of interoperability between the two domain ontologies. Figure 5 shows an outline of the algorithm for the method.

```
Preprocessing:
    Get domain ontology O1 and O2
    Compute depth of O1(depth_O1) and depth of O2(depth_O2)
    Compute generalisation limit (Gen_limit) { depth_O1/α}
    Similarity_Number_found = 0, Initialise L = 0, Number_of _leafNodes_i = 0, Gen_Hops_i = 1
Begin   {Compute Interoperability}:
     Read all leaf nodes (Leaf_Node_i) from O1 into process table (Table _Interoperability).
     Update Number_of Nodes_i
       For each Leaf_Node_i in Table _Interoperability
        Similarity_Number_found = Similarity(Leaf_Node_i , Node_j from O2)
        if Similarity_Number_found = True then
            Similarity_Number_found = Similarity_Number_found + 1
            Update status in Table _Interoperability
        else
            While Gen_Hops_i less than Gen_limit
                Get generalisation Gen_d of concept Leaf_Node_i
                Gen_Number_found = Similarity(Gen_d, Node_j from O2)
                Update Table_Interoperability with depth_O1 and depth_O2
                Compute L = min(depth_O1 , depth_O2)
                Update Table _Interoperability with L value
                Gen_Hops_i = Gen_Hops_i + 1
            endwhile;
          endif
     Similarity_Deg(f)   = Similarity_Number_found / Number_of leafNodes_i
     Generalisation_Deg(m) = ∑L / ∑depth_O1
          Degree_Interoperability = [Similarity_Deg(f) + Generalisation_Deg(m)] / 2
end.
```

Fig. 5. Ontology Interoperability measurement method algorithm

4.2 Method Application Example from the Geographical Domain

To illustrate the method outlined above, we take a look at two domain ontologies from the geographical land coverage domain. The two sample ontologies are LandClimatology (O1) and Landcover (O2). The protégé based structure of the first ontology is given in Fig. 6 and for the second ontology structure in Fig. 7.

From the ontology in Fig. 6, the most specialized classes are descendants of Forest class. The proposed method application uses these 5 leaf nodes for comparison.

Using *Sim* match, only 3 leaf nodes as seen in Table 4 have been matched giving an *f* value of 0.6 from Eq. (2). Therefore we don't have full interoperability hence the

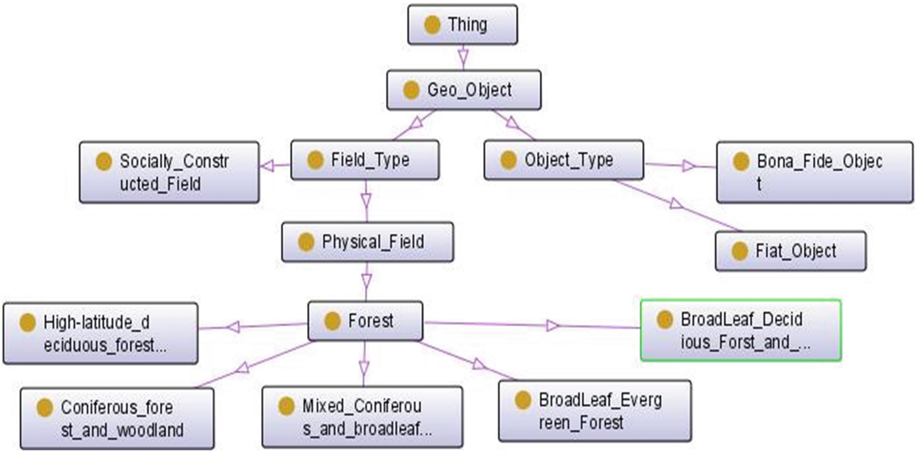

Fig. 6. LandClimatology ontology structure

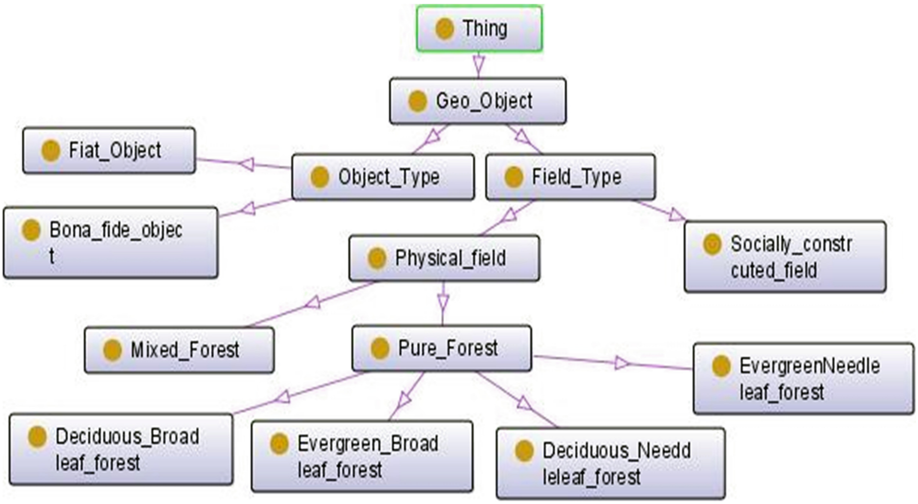

Fig. 7. LandCover ontology structure

method invokes the generalization (*Gen*) process to match the remaining classes as the seen in the Table 5.

Using Eq. (2) the degree of generalisation between the two ontologies is 0.75. Therefore the degree of interoperability between the two sample domain ontologies LandClimatology and Landcover is 0.68. Therefore that 68 % of ontology O1 is replaceable by ontology O2. However the method does not explain how the degree of interoperability computed affects the instance values in dataset likewise the performance reduces as the ontology size increases. We address this challenge by limiting the generalizable number of nodes to a quarter of the length of the base ontology.

Table 4. Similarity table between O1 and O2

#	Concepts from Ontology O1	Concepts from Ontology O2
1	`BroadLeaf_Evergreen_Forest`	`Evergreen_Broadleaf_forest`
2	`BroadLeaf_Decidious_Forst_and_Woodland`	`Deciduous_Broadleaf_forest`
3	`Mixed_Coniferous_and_broadleaf_decidious_forest`	`Evergreen_Broadleaf_forest`

Table 5. Generalisation table between ontology O1 and ontology O2

#	Ontology O1 Concepts	Ontology O2 Concepts	Depth l_1^i	Depth l_2^i	$\min(l_2^i, l_1^i)$
1	Coniferous_forest_and_woodland	Pure_Forest	4	3	3
2	High_latitude_deciduous_forest_and_woodland	Pure_Forest	4	3	3
			$\sum = 8$		$\sum = 6$

5 Conclusion

The paper outlined how to measure the degree of interoperability between two domain ontologies and provide a value of the extent to which they can replace each other. The approach is based on a model management operator to define different degrees of interoperability. The method can enable domain ontology designers and system integrators to make quicker and better informed selections between ontologies for adoption but it falls short in explaining the integration impact on the ontology instances. The method performance speed tends to decrease as the depth of the ontology becomes higher

Acknowledgements. The Universiti Teknologi Malaysia (UTM) and Ministry of Education (MOE) Malaysia under Research University grant Vots 02G31 and 00M19 are hereby acknowledged for some of the facilities utilized during the course of this research work and for supporting the related research.

References

1. Gruber, T.R.: A translation approach to portable ontology specifications. Knowl. Acquis. **5**, 199–220 (1993)
2. Harsu, M.: A survey on domain engineering (2002)
3. Seremeti, L., Kougias, I.: Computation of ontology resemblance coefficients for improving semantic interoperability. Eng. Math. Lett. **2**, 1–19 (2013)
4. Zutshi, A.: Framework for a business interoperability quotient measurement model. Master thesis dissertations, Departamento de Engenharia Mecânica e Industrial, Universidade Nova de Lisboa, Portugal (2010). http://run.unl.pt/bitstream/10362/2646/1/Zutshi_2010.pdf

5. Sanchez Ruenes, D.: Domain Ontology learning from the Web. Ph.D. thesis, Departamento de Lenguajes y Sistemas Informaticos, Universidad Politecnica de Catalufia (2007). http://www.tdx.cat/bitstream/10803/6650/1/01Dsr01de02.pdf
6. Acampora, G., Vitiello, A.: Improving agent interoperability through a memetic ontology alignment: a comparative study. In: Fuzzy-IEEE International Conference on Systems, pp. 1–8 (2012)
7. Interop, N.: State of Art Report Ontology Interoperability, (n.d.)
8. Kalfoglou, Y., Schorlemmer, M.: Ontology mapping: the state of the art. Knowl. Eng. Rev. **18**, 1–31 (2003)
9. Euzenat, J.: Semantic precision and recall for ontology alignment evaluation. In: IJCAI, pp. 348–353 (2007)
10. Shvaiko, P., Euzenat, J.: A survey of schema-based matching approaches. In: Spaccapietra, S. (ed.) Journal on Data Semantics IV. LNCS, vol. 3730, pp. 146–171. Springer, Heidelberg (2005)
11. Euzenat, J., Shvaiko, P.: Ontology Matching. Springer, Heidelberg (2007)
12. Bruijn, J., Ehrig, M., Feier, C., Martins-Recuerda, F., Scharffe, F., Weiten, M.: Ontology mediation, merging, and aligning. In: Davies, J., Studer, R., Warren, P. (eds.) Semantic Web Technologies, pp. 95–113. Wiley, Chichester (2006)
13. Noy, N.F., Musen, M.A.: Anchor-PROMPT: using non-local context for semantic matching. Framework **39**, 63–70 (2001)
14. Ehrig, M., Staab, S.: QOM – quick ontology mapping. In: McIlraith, S.A., Plexousakis, D., van Harmelen, F. (eds.) ISWC 2004. LNCS, vol. 3298, pp. 683–697. Springer, Heidelberg (2004)
15. Maedche, A., Motik, B., Silva, N., Volz, R.: {MAFRA} - an ontology mapping framework in the semantic web. In: Proceedings of the ECAI Workshop on Transformation, Lyon, France (2002)
16. Noy, N.F., Musen, M.A.: The PROMPT suite: interactive tools for ontology merging and mapping. Int. J. Hum. Comput. Stud. **59**, 983–1024 (2003)
17. Amrouch, S., Mostefai, S.: Survey on the literature of ontology mapping, alignment and merging. In: 2012 International Conference on Information Technology and E-Services, ICITeS (2012)
18. Ehrig, M.: Ontology alignment - bridging the semantic gap. In: Management, p. 250 (2005)
19. Beneventano, D., Orsini, M., Po, L., Sorrentino, S.: The MOMIS - STASIS approach for ontology-based data integration. In: Proceedings of the ISDSI (2009)
20. Klein, M., Fensel, D., Kiryakov, A., Ognyanov, D.: Ontology versioning and change detection on the web. In: Gómez-Pérez, A., Benjamins, V. (eds.) EKAW 2002. LNCS (LNAI), vol. 2473, pp. 197–212. Springer, Heidelberg (2002)
21. Kent, R.E.: The IFF foundation for ontological knowledge organization. Cat. Classif. Q. **37**, 187–203 (2003)
22. Castano, S., Ferrara, A., Montanelli, S., Zucchelli, D.: HELIOS: a general framework for ontology-based knowledge sharing and evolution in P2P systems. In: Proceedings of the 14th International Workshop on Database and Expert Systems Applications (2003)
23. Choi, N., Song, I.-Y., Han, H.: A survey on ontology mapping. ACM SIGMOD Rec. **35**, 34–41 (2006)
24. Kotis, K., Vouros, G.A: The HCONE approach to ontology merging. In: Web. pp. 1–15 (2008)
25. Preece, A., Hui, K., Gray, A., Marti, P., Bench-Capon, T., Jones, D., et al.: KRAFT architecture for knowledge fusion and transformation. Knowl. Based Syst. **13**, 113–120 (2000)

26. Do, H.-H., Rahm, E.: COMA: a system for flexible combination of schema matching approaches. In: Proceedings of the 28th International Conference on Very Large Data Bases, pp. 610–621 (2002)
27. Stumme, G.: FCA-merge: bottom-up merging of ontologies. In: International Joint Conference on Artificial Intelligence, pp. 225–230 (2001)
28. Ehrig, M.: Foam - framework for ontology alignment and mapping; results of the ontology alignment initiative. In: Proceedings of the Work Integrated Ontology, vol. 156, pp. 72–76. CEUR-WS.org (2005)
29. Lambrix, P., Tan, H.: SAMBO-A system for aligning and merging biomedical ontologies. Web Semant. **4**, 196–206 (2006)
30. Giunchiglia, F., Shvaiko, P., Yatskevich, M.: S-match: an algorithm and an implementation of semantic matching. In: Bussler, C.J., Davies, J., Fensel, D., Studer, R. (eds.) ESWS 2004. LNCS, vol. 3053, pp. 61–75. Springer, Heidelberg (2004)
31. Li, J., Tang, J., Li, Y., Luo, Q.: RiMOM : a dynamic multistrategy ontology alignment. Framework **21**, 1–15 (2009)
32. Clifton, C.: Experience with a combined approach to attribute-matching across heterogeneous databases. In: Techniques. pp. 1–17 (1997)
33. An, Y., Borgida, A., Mylopoulos, J.: Inferring complex semantic mappings between relational tables and ontologies from simple correspondences. In: Meersman, R. (ed.) OTM 2005. LNCS, vol. 3761, pp. 1152–1169. Springer, Heidelberg (2005)
34. Qian, Y., Li, Y., Song, J., Yue, L.: Discovering complex semantic matches between database schemas. In: International Conference on Web Information Systems and Mining, WISM, pp. 756–760 (2009)
35. Li, W.S., Clifton, C.: SEMINT: a tool for identifying attribute correspondences in heterogeneous databases using neural networks. Data Knowl. Eng. **33**, 49–84 (2000)
36. Velegrakis, Y., Miller, R.J., Popa, L., Mylopoulos, J.: ToMAS: a system for adapting mappings while schemas evolve. In: Proceedings of the International Conference on Data Engineering, p. 862 (2004)
37. Blanchard, E., Harzallah, M.: A typology of ontology-based semantic measures. In: EMOI (2005)
38. Rada, R., Mili, H., Bicknell, E., Blettner, M.: Development and application of a metric on semantic nets. IEEE Trans. Syst. Man. Cybern. **19**, 17–30 (1989)
39. Zhibiao Wu, P.M.: Verb semantics and lexical selection. In: Proceedings of the 32nd Annual Meeting on Association for Computational Linguistics, pp. 133–138 (1994)
40. Leacock, C., Chodorow, M.: Combining local context and WordNet similarity for word sense identification. In: Fellbaum, C. (ed.) WordNet An Electronic Lexical Database, pp. 265–283. MIT Press, Cambridge (1998)
41. Resnik, P.: Using information content to evaluate semantic similarity in a taxonomy. In: Proceedings of the 14th International Joint Conference on Artificial Intelligence (1995)
42. Lin, D.: Principle-based parsing without overgeneralization. In: Meeting of the Association for Computational Linguistics, pp. 112–120 (1993)
43. Gan, M., Dou, X., Jiang, R.: From ontology to semantic similarity: calculation of ontology-based semantic similarity. Sci. World J. **2013**, 793091 (2013)
44. Sánchez, D., Batet, M., Isern, D., Valls, A.: Ontology-based semantic similarity: a new feature-based approach. Expert Syst. Appl. **39**, 7718–7728 (2012)
45. Al-Mubaid, H., Nguyen, H.A.: Measuring semantic similarity between biomedical concepts within multiple ontologies. IEEE Trans. Syst. Man, Cybern. Part C (Appl. Rev.) **39**, 389–398 (2009)

46. Hirst, G., St-Onge, D.: Lexical chains as representations of context for the detection and correction of malapropisms. In: WordNet - An Electronic Lexical Database, pp. 305–332 (1998)
47. Luong, H.P., Gauch, S., Wang, Q.: Ontology learning through focused crawling and information extraction. In: International Conference on Knowledge and Systems Engineering 2009, pp. 106–112. IEEE (2009)
48. Knappe, R.: Measures of Semantic Similarity and Relatedness for Use in Ontology-based Information Retrieval (2005)
49. Pesquita, C., Faria, D., Falcão, A.O., Lord, P., Couto, F.M.: Semantic similarity in biomedical ontologies. PLoS Comput. Biol. **5**, e1000443 (2009)
50. Bernstein, P.: Applying model management to classical meta data problems. In: Proceedings of the CIDR (2003)