

MAK 00329.



0000097974

Positioning coverage mismatch in ontology integration for
OWL-based ontology / Emaliana Kasmuri, Peter Woods.

POSITIONING COVERAGE MISMATCH IN ONTOLOGY INTEGRATION FOR OWL-BASED ONTOLOGY

**EMALIANA BINTI KASMURI
PETER WOODS**

(11th International Conference on Hybrid Intelligent Systems 2011 (HIS 2011), 5-8 December 2011, Melaka)

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Positioning Coverage Mismatch in Ontology Integration for OWL-Based Ontology

Emaliana Kasmuri

Fakulti Teknologi Maklumat dan Komunikasi,
Universiti Teknikal Malaysia Melaka, Durian Tunggal,
Melaka, Malaysia
emaliana@utem.edu.my

Peter Woods

Knowledge of Management Centre,
Faculty of Creative Multimedia, Multimedia University,
Cyberjaya, Selangor, MALAYSIA
p.woods@mmu.edu.my

Abstract- Many challenges are to be addressed when it comes to integrating ontologies. The common challenges are ontology mismatches. It is important for the integrated ontology to model the intended meaning and resolve all the conflicts so that it will not form a false commitment to the system. Much work has paid attention to finding similarities between ontologies. Little work has considered overcoming coverage mismatch, to solve non-existent concepts, the same concepts being defined differently, and differences in hierarchical structure. This paper identifies, analyzes and models coverage mismatches, filling that gap using OWL ontology. This work is an enhancement to the current findings of ontology mismatch.

Keywords-Ontology, integration, mismatch

I. INTRODUCTION

One of the key reasons using ontologies is to facilitate interoperability for distributed systems [1]. Today, ontologies have played major roles in various areas such as knowledge management [2], [3], software engineering [4], [5], [6] and information retrieval [7] [8]. Ontology describes the facts of domain using a collection of terms. These terms are represented using classes, properties and individuals processable to the machine and understandable to humans. Ontologies facilitate the machine to understand the underlying meaning of the domain thus enabling information to be exchanged among software agents.

Many ontologies are develop independently and made available to the public to comply to various specifications, needs, and tasks. However, in a distributed environment, it is very unlikely a system will commit to a single ontology. Many have overlapping parts and some are not complete to the point they require the ontology to be integrate with others. Integrating ontologies is part of reusing knowledge and adding value to the domain. There are many challenges to integrate ontologies and the most anticipated problem is ontology mismatch. Extensive studies on ontology mismatches have been conducted [9]. [10] has developed the mismatch framework. Later [11] and [12] have refined the framework. Even though much work has been done to identify and overcome the mismatches, there are some parts that received less attention. This paper is intended to fill that gap.

This paper is to enhance the current findings of ontology mismatch. The purpose is to highlight and

position the importance of addressing coverage mismatch in ontology integration. This paper is organized as follows. Section 2 describes ontology mismatch. Section 3 defines coverage mismatch. Section 4 discusses the importance of addressing coverage mismatch and Section 5 the future works from this paper.

II. ONTOLOGY MISMATCH

Ontology integration is one of the ways to overcome heterogeneous ontology. One of the challenges integrating the ontologies is to solve ontology mismatch. This challenge has been addressed in [13], [14], [15], [16], [17], [18], and [19].

Mismatches occur in ontology when two or more ontologies are badly matched. This happens due to inconformity from one ontology to the other. Ontology mismatch happens at two level – language and ontology level [10]. Language mismatch happens when two ontologies are using different languages to represent the same concepts. For example, Ontology A is using OWL and the other is using XML. This requires one of the ontology to be translated into the preferred language.

Mismatch at ontology level happens when concepts are overlapped. This mismatch can occur for ontologies that are written using the same language. Overlaps happens when similarities exist in the way the ontologies are used and interpreted which known as conceptualization or the way it is presented which known as explication. This paper is positioning mismatches at conceptualization level. It is found that less attention has been paid in conceptualization mismatch and this paper is intend to complement the existing research findings by refining the coverage mismatch into greater detail as shown in Figure 1.

This paper is using domain ontology to classify the coverage mismatch. Domain ontology is selected due to the increasing number of available ontology developed and made available to the public. The shaded boxes in Figure 1 are the type of coverage mismatches found in this study. The framework from [10] is refined with more coverage mismatches. The details of the coverage mismatch are elaborated in the next section.

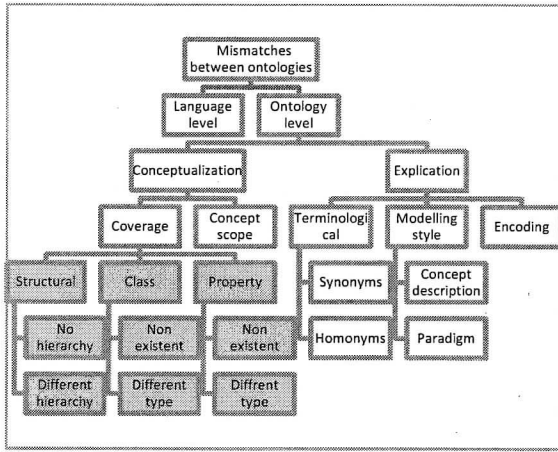


Figure 1 : A Refined framework of ontology mismatches adapted from [10]

III. COVERAGE MISMATCH

[20] has highlighted that task context has a significant influence on the coverage, content and structure of the developed ontology. Coverage of an ontology refers to the amount of classes or properties modeled compared to the domain corpus [21]. Coverage mismatch exists in between two ontologies when there are differences in scope or area that are formally represented. It covers different subsets of the domain.

The differences can be detected in the level of detail of the ontology. Coverage mismatch can be detected with the presence or the absence of classes or property in between two ontologies [22] [11]. This mismatch could occur even if the ontologies are using the same language. This paper has found that coverage mismatch happens at three levels which are class level, property level and ontology level.

A. Coverage Mismatch at Class Level

A class represents a concept of from the selected domain. OWL has four different types of classes which are simple naming class, defined class, enumerated class and disjoint class.

A class is defined as a structure of

$$c \in C(D, R) \quad (1)$$

where

C is conceptualization

D is domain

R is a set of relation that is relevant to D

There are two cases where coverage mismatch could happen at class level, 1) Non-existent class 2) Different

types of class. The first case occurs where a class exists in one ontology but not in the other. This is applies to classes that have no matches in the ontology. It is known as unmatched class. For example, Ontology A and Ontology B are both model for tourism domain. Class `CarRental` exists in Ontology A but not the other.

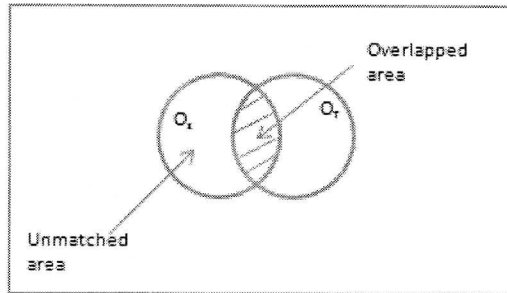


Figure 2 : Illustration of unmatched area indicating entities that exist in one ontology but not the other.

Figure 2 illustrate the conceptual unmatched and overlapped area to indicate the coverage mismatch. Classes can be syntactically or semantically overlapped. Thus, the overlapping classes can be measured using lexical similarities [23] and semantic matching [24]. However, the classes from the same domain are very likely to have the high lexical similarities.

The first case of class level mismatch can be computed as

$$mis_{(c_s, c_t)} = \frac{c_s \cap c_t}{c_s \cup c_t} \quad (2)$$

where

c_s is class of source ontology

c_t is class of target ontology

$mis_{(c_s, c_t)}$ is a mismatch value (0 | 1) which indicates 0 has coverage mismatch and 1 no coverage mismatch.

The second case occurs where the same concepts are defined using different types of classes. This might exist in overlapped area as shown in Figure 2. For example Ontology A models `Accomodation` into several types of information context which are `BedAndBreakfast`, `BudgetAccomodation`, `Campground` and `Hotel`. Ontology B defines `Accomodation` using descriptive logic if an instance `hasRoom` either `GuestRoom` or `ConferenceRoom` type.

B. Coverage Mismatch at Property Level

A property defines relationship of a class and an individual. In OWL, a property can be defined using two types either object property or data property. Object property members are using defined classes for domain and range whereas data property is using primitive values to specify the value of the range.

A property is defined as a structure of

$$p = \langle (d, r), rt, n \rangle, \quad (3)$$

where

d is domain instance of a class
 r is range of class instance or primitive data type,
 rt is restriction and
 n is cardinality

There are two cases where coverage mismatch could happen at property level, 1) Non-existent property, 2) Different types of properties. The first case occurs where a property exists in one ontology but not in the other. This is happens when the property in one ontology has no match in the other. For example property `hasBooking` exists in Ontology A but not in Ontology B. The similarity of properties can be measured using the same approach as the class similarities in the previous section. Therefore this first case of property coverage mismatch can be model as follows

$$mis_{(p_s, p_t)} = \frac{p_s \cap \overline{p_t}}{p_s \cup p_t} \quad (4)$$

where

p_s is property of source ontology
 p_t is property of target ontology
 $mis(p_s, p_t)$ is a mismatch value (0 | 1) which indicates 0 has coverage mismatch and 1 no coverage mismatch.

The second case occurs when similar properties are defined using different types. For example, class `Accommodation` in Ontology A has a domain for a property named `hasRating`. `hasRating` domain is `AccommodationRating` and range is `OneStarRating`, `TwoStarRating` or `ThreeStarRating`. In Ontology B, `Accommodation` is a domain for a property named `hasStarRating` which the range is an integer value. Here `hasRating` and `hasStarRating` are deemed to be similar. However, the range is different. One is using multiple values of classes and the other is using primitive value.

Therefore the coverage of two properties can be determine with the function of

$$mis_{(p_s, p_t)} = \frac{(d_s \cap \overline{d_t}) \cdot (r_s \cap \overline{r_t}) \cdot (rt_s \cap \overline{rt_t}) \cdot (n_s \cap \overline{n_t})}{(d_s \cup \overline{d_t}) \cdot (r_s \cup \overline{r_t}) \cdot (rt_s \cup \overline{rt_t}) \cdot (n_s \cup \overline{n_t})} \quad (5)$$

where

d is domain for a property
 r is range for a property
from s , source and t , target ontologies
 $mis_{(p_s, p_t)}$ is the mismatch value

C. Coverage Mismatch at Structural Level

Classes are organized using hierarchical structure in ontology. A class can have one or more siblings. This type of class is known as super class. A class can belong to one or more parents. This class is known as sub class.

The coverage mismatch at structural level can happen in one of these two cases 1) No hierarchy, 2) Different hierarchy. The first case happen when two classes are deemed as similar where by one class has superclass(es) or subclass(es) but the other does not have either subclass(es) or super class(es). For example, Ontology A models `Accommodation` into several sub classes which are `BedAndBreakfast`, `BudgetAccommodation`, `Campground` and `Hotel`. However in Ontology B, `Accommodation` do not have super class(es) or sub class(es).

The second case happens when two classes that are identified as similar are located in different hierarchy in the ontology. A class in one ontology can be a super class but not in the other and vice versa. For example `Destination` and `TravelDestination` are deemed to be the same concepts, sub classes such as `City`, `Country` and `Region` are also deemed to be the same concept. However, `City` in Ontology A has different structure from `City` in Ontology B by which `City` in Ontology A is sub class of `UrbanArea`.

IV. THE IMPORTANCE OF ADDRESSING COVERAGE MISMATCH

In general coverage mismatch happens because the ontology is model to different sub domains. The sub domains represents different parts of a domain. [25] address coverage issue in Figure 3. When the needs to use these ontologies together, coverage mismatch is important to be address because the ontologies need to be used as a coherent source, to get higher precision and to maximize the ontology coverage. Differences of the classes and properties in the way it is defined and structured must be solve to avoid unintended conflicts such as concept or property redundancies. The new ontology should reflect the complexity of all concepts captured from the sub domains.

Addressing the coverage should reduce the gap between the ontology model and the domain concepts. In a way, solving coverage mismatch will cover the intended portions that will not form a false commitment between the system and the ontology.

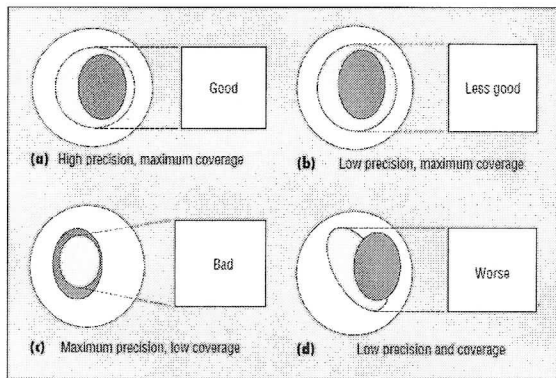


Figure 3 : Comparison of different ontologies with respect to precision and coverage [25]

V. CONCLUSION AND FUTURE WORKS

Integrating two ontologies is a difficult task which has become more and more important with the expected advancement in Semantic Web. This paper has categorized and analyzed types of coverage mismatch. This paper has selected OWL based ontology due to its prominence. This work has complements the existing findings of ontology mismatches. It is hope that by categorizing and analyzing these coverage mismatches would provide a good basis to select the right strategy for integrating the ontologies.

REFERENCES

- [1] Gruber, T.R., *A translation approach to portable ontology specifications*. Knowl. Acquis., 1993. **5**(2): p. 199-220.
- [2] Doherty, M., et al., *Using ontologies to improve knowledge management in technology transfer*. International Journal of Technology Transfer and Commercialisation 2005. Vol. **4**(No.1): p. 105 - 125.
- [3] Jurisica, I., J. Mylopoulos, and E. Yu, *Ontologies for Knowledge Management: An Information Systems Perspective*. Knowledge and Information Systems, 2004. **6**(4): p. 380-401.
- [4] Wongthongtham, P., E. Chang, and T.S. Dillon, *Enhancing Software Engineering Project Information through Software Engineering Ontology Instantiations*, in *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence*. 2006, IEEE Computer Society.
- [5] Henderson-Sellers, B., *Bridging metamodels and ontologies in software engineering*. J. Syst. Softw. **84**(2): p. 301-313.
- [6] Wongthongtham, P., E. Chang, and A.A. Aseeri, *Towards social network based approach for software engineering ontology sharing and evolution*, in *Proceedings of the 2007 OTM Confederated international conference on On the move to meaningful internet systems - Volume Part II*. 2007, Springer-Verlag: Vilamoura, Portugal.
- [7] Zhuhadar, L. and O. Nasraoui, *Augmented Ontology-Based Information Retrieval System with External Open Source Resources*, in *Proceedings of the 2010 Seventh International Conference on Information Technology: New Generations*, IEEE Computer Society.
- [8] Gomez-Perez, A., F. Ortiz-Rodriguez, and B. Villazon-Terrazas, *Ontology-based legal information retrieval to improve the information access in e-government*, in *Proceedings of the 15th international conference on World Wide Web*. 2006, ACM: Edinburgh, Scotland.
- [9] Visser, P.R.S., et al., *An Analysis of Ontology Mismatches : Heterogeneity vs interoperability*, in *AAAI 1997 Spring Symposium on Ontological Engineering*. 1997: Stanford, USA.
- [10] Klein, M., *Combining and relating ontologies: an analysis of problems and solutions*, in *IJCAI'01*. 2001: Seattle, USA.
- [11] Hameed, A., A. Preece, and D. Sleeman, *Ontology Reconciliation*, in *Handbook on Ontologies*, S. Staab and R. Studer, Editors. 2004, Springer. p. 231-250.
- [12] Qadir, M.A., M. Fahad, and M.W. Noshairwan. *On Conceptualization Mismatches Between Ontologies*. in *2007 IEEE International Conference on Granular Computing (GRC 2007)*. 2007.
- [13] Ceusters, W., *Towards A Realism-Based Metric for Quality Assurance in Ontology Matching*, in *Proceeding of the 2006 conference on Formal Ontology in Information Systems: Proceedings of the Fourth International Conference (FOIS 2006)*. 2006, IOS Press.
- [14] Fahad, M., et al. *Disjoint-Knowledge Analysis and Preservation in Ontology Merging Process*. in *Software Engineering Advances (ICSEA), 2010 Fifth International Conference on*. 2010.
- [15] Guanyu, L., L. Zhenghai, and S. Jianshuang. *Multi-mapping based ontology merging system design*. in *Advanced Computer Control (ICACC), 2010 2nd International Conference on*. 2010.
- [16] Noy, N.F. and M.A. Musen. *SMART: Automated Support for Ontology Merging and Alignment*. in *Proceedings of the 12th Workshop on Knowledge Acquisition, Modeling and Management*. 1999. Banff, Canada.
- [17] McGuinness, D.L., et al. *The Chimaera Ontology Environment*. in *Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI 2000)*. 2000. Austin, Texas.
- [18] Kim, J., et al. *MoA: OWL Ontology Merging and Alignment Tool for the Semantic Web*. 2005.
- [19] Pinto, H.S. and J.P. Martins, *A Methodology for Ontology Integration*, in *K-CAP'01*. 2001: Victoria, British Columbia.
- [20] Smart, P.R. and P.C. Engelbrecht, *An Analysis of the Origins of Ontology Mismatches on the Semantic Web*, in *16th International Conference on Knowledge Engineering and Knowledge Management (EKAW 2008)*. 2008: Acitrezza, Catania, Italy.
- [21] Demir, E., et al., *An ontology for collaborative construction and analysis of cellular pathways*. Bioinformatics, 2004. **20**(3): p. 349-356.
- [22] Missikoff, M., F. Schiappelli, and F. Taglino. *A Controlled Language for Semantic Annotation and Interoperability in e-Business Application*. 2003.
- [23] Maedche, A. and S. Staab, *Measuring Similarity between Ontologies*, in *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web*. 2002, Springer-Verlag.
- [24] Giunchiglia, F., M. Yatskevich, and P. Shvaiko, *Semantic matching: algorithms and implementation*, in *Journal on data semantics IX*, S. Stefano, et al., Editors. 2007, Springer-Verlag. p. 1-38.
- [25] Staab, S., et al., *Why evaluate ontology technologies? Because it works!* Intelligent Systems, IEEE, 2004. **19**(4): p. 74-81.