Ontologies and similarity

Steffen Staab
staab@uni-koblenz.de
http://west.uni-koblenz.de

Institute for Web Science and Technologies, Universität Koblenz-Landau, Germany

1 Introduction

Ontologies [9] comprise a definition of concepts describing their commonalities (genus proximum) as well as their differences (differentia specifica). One might think that with the definition of commonalities and differences, the definition of similarities in and for ontologies should follow immediately. Traditionally, however, the contrary is true, because the method background of ontologies, i.e. logics-based representations, and similarity, i.e. geometry-based representations, have been explored in disjoint communities that have mixed only to a limited extent. In this short paper we survey how our own work touches on the intersection between ontologies and similarity. While this cannot be a comprehensive account of the interrelationship between ontologies and similarity, we aim it to be a stepping stone for inspiration and for indicating entry points for future investigations.

2 Similarity

When analyzing the interplay of ontologies and similarities, we have encountered two issues that need to be clarified first:

1. For which entities should similarity be assessed?
   - Objects: In many applications, the eventual target is the assessment of similarity between objects. For instance, in information retrieval one may want to search for the document vector neighboring most closely to a given query vector whereby the assessment of similarity should take the ontology-based semantics of query and document representations into account. Here, the ontology comes as an auxiliary means of influencing the geometric space in a desired manner, e.g. [10].
   - Concepts: Frequently the entities to be compared are the concepts defined in one or several ontologies. For instance, the integration of two information systems may be pursued by aligning the two ontologies that conceptualize the underlying information systems. Correspondencies between concepts from the two ontologies may be explored by taking different types of similarities between concepts into account, e.g. [7].
   - Ontologies: In knowledge engineering the task of comparing ontologies is sometimes required in order to answer questions about the match between two ontologies, e.g. if they have been learned by automatic means,

e.g. [6]. Not all of the measures used here are similarity measures in the mathematical sense, but similarities are fed into precision/recall-like measure in order to judge the container-containee relationship between two ontologies.

2. What is the objective of this similarity assessment?
   – Numeric Similarity Assessment: The most common type of similarity assessment is based on the mathematical notion of similarity measure that fulfills the following core properties for any given entities e, e_1, e_2:

$$
\begin{gathered}
sim(e_1, e_2) \in [0, 1] \\
sim(e, e) = 1 \\
sim(e_1, e_2) = sim(e_2, e_1)
\end{gathered}
$$

   – Preference Ordering: While one may find quite some efforts in the literature for aligning the numerical assessment of similarity with judgements found in user experiments, in many application cases numerical measures are in fact not needed. Application cases like clustering of objects primarily need information about which pair of objects is most similar to each other — regardless of a numerical value. This is particularly relevant when ontological knowledge is so incomplete that deciding about such a preference ordering is still possible, while further measurements cannot be reasonably predicted.

3 Ontological Foundations for Similarity Assessments

Even within computer science, the word "ontology" is used in two senses. In its proper sense [9] an ontology is *a formal specification of a shared conceptualization of a domain of interest*. Thereby, the conceptualization abstracts from the particularities of a particular situation, e.g. a household ontology would typically include the definition that a desk is a table, while it would not contain statements about whether a desk is actually found in a particular household, which rather constitutes a situational and possibly changing aspect:

$$desk \sqsubseteq table$$

The word "formal" refers to the use of a mathematical mechanism for describing an ontology. In practice, the Web Ontology Language OWL, which is derived from the paradigm of description logics [14], is most frequently used as a notation for writing down ontologies, as it is an standardized, expressive language with a clear formal semantics. In description logics, one may distinguish definitional knowledge about concepts and relationships found in the T(erminological)-Box, e.g. every desk being a table, and assertional knowledge about objects, e.g. a specific object being a desk and occurring in a given household, such as found in the following A(ssertional)-Box:

$$Desk(object1). occur(object1, household2).$$

Thus, the Web Ontology Language provides convenient language constructs for specifying an ontology as well as for specifying the facts found in a particular situation. This convenience led to the use of the term "ontology" in a second meaning, namely as refering to a complete knowledge base consisting of an ontology in the proper sense and factual knowledge about a situation.

3.1 Logics-based Ontology Representations

The full advantage of description logics is derived from the expressiveness of the language allowing for m the possibility to use concepts and relations to define new concepts. The following definition describes that a desk that has a Minibar which only contains FancyDrink$s$ is a FancyDesk:

$$FancyDesk \sqsubseteq Desk \sqcap \exists hasPart.(MiniBar \sqcap \forall contains.FancyDrink)$$

This advantage, however, is problematic when, e.g., concept expressions are to be compared. For instance, the following definition specifies that a FavoriteDesk is a Desk with a Compartment having at least some SingleMalt:

$$FavoriteDesk \sqsubseteq Desk \sqcap \exists hasPart.(Compartment \sqcap \exists contains.SingleMalt)$$

When assessing the similarity between FancyDesk and FavoriteDesk, the intricate logic expressions lead to many non-trivial problems that we have analysed in [4]. The core result of this analysis was that existing similarity measures were not adequately reflecting the richness of description logics with some of them being unsound. We then suggested a new measure that uses the richness of description logics in a sound way, however, it is very expensive to implement.

3.2 Lattice-based Ontology Representations

Historically, researchers did not approach the problems of similarity in ontology-based representations, but they rather relied on simpler representations. Some of them cannot be called ontology representations anymore, because they fail to reflect the richness of interactions between concepts and relationships between concepts.

A very elegant mechanism is formal concept analysis [8]. Here objects (table1 ... seat4) are represented by whether they have a property (hasLegs ... hasCompartment) or not. An example is given in Table 1.

This table of binary decision is analysed revealing what Ganter and Wille call 'formal concepts'. Each formal concept has an extension consisting of a subset of objects and an intension consisting of a subset of properties, e.g. ({chair3,seat4},{hasLegs, offersSeating, hasSurface}) is a formal concept — obviously representing the class of all chairs. Also subclass relationships between formal concepts can be derived, for this particular example there will be a class of desks (containing desk1) and a class of tables (containinng table1 and desk2), with the former being more specific than the latter.

	hasLegs	offersSeating	hasSurface	hasCompartment
table1	X		X	
desk2	X		X	X
chair3	X	X	X	
seat4	X	X	X	

Table 1. Formal context representing the relations between objects and properties

This representation is very interesting during ontology engineering and for data mining (and many extensions towards non-binary table entries and higher-arity relationships exist), but it is not close to common ontology representations. The structures are still very useful as approximations of ontological structures and the binary vectors lend themselves as a basic means for set-based assessments of similarity between objects, properties and formal concepts.

Formal concept analysis has been used to induce a taxonomy from words and their correlations to other words in texts [2]. While we are not aware of direct applications of formal concept analysis starting with an ontology, we see fruitful possibilities for such future use — especially because the given method constitutes a well-researched mathematical method with corresponding well-defined operators.

3.3 Graph-based Ontology Representations

Graph-based representations of ontologies are the most frequently used means to assess similarities in or between ontologies. The advantage of using graphs is that the definitions are easy to implement and to adjust to specific needs. The disadvantage is that graphs also cannot capture the richness of descriptions found in OWL ontologies.

Most of the work on determining similarity between concepts in ontologies built on the core idea of Resnik [13] that such similarity is defined on the basis of a. a stable taxonomy, b. the counting of taxonomic links between two concepts that are to be compared and c. some normalization to take into account the height of a concept in this taxonomy, i.e. some version of measuring the extension of two concepts.

4 Applying Similarities and Ontologies

A very comprehensive survey of similarity measures in ontologies can be found in the dissertation by d'Amato [3]. Interestingly, these assumptions by Resniks (and 'followers') did not match very well the actual situation found in description logics ontologies. But independently from which representation is chosen to define which exact similarity measure, there is a wide range of applications for which this combination has been used. We simply give here an indicative list without much further explanation (and without claiming completeness): Information Retrieval [10]; Machine Learning [1]; Web Service Discovery [11]; Ontology Alignment [7]; Ontology Learning: Evaluation [12, 6]; Ontology Learning: Induction [2]; Indexing of description logics ontologies [5]

5 Conclusion

We have sketched hier the possible space of how different ontology representations (or approximations of ontologies) may be used as a foundation for similarity measures. Depending on which entities are to be compared and what the purpose of comparison is a corresponding similarity measure must be chosen. It is intriguing to note that many of the existing similarity methods do not match with the assumptions underlying description logics ontologies — making existing methods invalid and requiring further exploration of this space of ontologies and similarities.

Acknowledgements This research has been co-funded by DFG in the project "Multipla".

References

1. S. Bloehdorn and Y. Sure. Kernel methods for mining instance data in ontologies. In *Proc. of ISWC/ASWC 2007*, pages 58–71, 2007.
2. P. Cimiano, A. Hotho, and S. Staab. Learning concept hierarchies from text corpora using formal concept analysis. *JAIR — Journal of AI Research*, 24:305–339, 2005.
3. C. d'Amato. *Similarity-based Learning Methods for the Semantic Web*. PhD thesis, University of Bari, 2007.
4. C. d'Amato, S. Staab, and N. Fanizzi. On the influence of description logics ontologies on conceptual similarity. In *Proc. of the European Conference on Knowledge Acquisition and Management EKAW-2008. Sicily, Italy*, LNCS. Springer, 2008.
5. C. D'Amato, S. Staab, N. Fanizzi, and F. Esposito. Dl-link: A conceptual clustering algorithm for indexing description logics knowledge bases. *International Journal of Semantic Computing*, 4(4):453–486, 2011.
6. K. Dellschaft and S. Staab. On how to perform a gold standard based evaluation of ontology learning. In *Proc. of ISWC-2006 International Semantic Web Conference, Athens, GA, USA*, LNCS. Springer, November 2006.
7. M. Ehrig, S. Staab, and Y. Sure. Bootstrapping ontology alignment methods with apfel. In *Proc. of ISWC-2005 International Semantic Web Conference, Galway, Ireland*, 2005.
8. B. Ganter and R. Wille. *Formal Concept Analysis Mathematical Foundations*. Springer, 1999.
9. N. Guarino, D. Oberle, and S. Staab. What is an ontology? In *Handbook on Ontologies*, 2009. 2nd revised edition.
10. A. Hotho, S. Staab, and G. Stumme. Ontologies improve text document clustering. In *Proceedings of the International Conference on Data Mining ICDM-2003*. IEEE Press, 2003.

11. Matthias Klusch, Benedikt Fries, and Katia P. Sycara. Owls-mx: A hybrid semantic web service matchmaker for owl-s services. *Journal of Web Semantics*, 7(2):121–133, 2009.
12. A. Maedche and S. Staab. Measuring similarity between ontologies. In *Proc. Of the European Conference on Knowledge Acquisition and Management EKAW-2002. Madrid, Spain, October 1-4, 2002*, LNCS/LNAI 2473, pages 251–263, 2002.
13. P. Resnik. Using information content to evaluate semantic similarity in a taxanomy. In *Proc. of Int. Joint Conf. for Artificial Intelligence (IJCAI-95)*, pages 448–453, 1995.
14. S. Staab and R. Studer, editors. *Handbook on Ontologies*. 2009. 2nd revised edition.