Modularity and Ontology Change
Defence Date: January, 2019

Ricardo F. Guimarães[1], **Supervisor: Renata Wassermann**[1]

[1]Institute of Mathematics and Statistics – University of São Paulo (USP)
São Paulo – SP – Brazil

`{ricardof,renata}@ime.usp.br`

Abstract. *We consider, in this work, ontologies as documents describing the conceptualization of a domain, in a computer-processable language. Specifically, we address the problem of repairing and evolving ontologies consistently. These non-trivial tasks become even harder for ontologies with hundreds of thousands of axioms. Since the standard language set by the W3C for writing ontologies in the Semantic Web, OWL, is founded in the family of the description logics, we adopt the view of ontologies as finite sets of formulae in a logic of this family. This view allows us to combine the theory of Belief Revision, which formalises rational change in logical objects, and strategies of modularisation to devise and improve procedures for ontology change.*

1. About the Author

I am currently a PhD student in Computer Science at the Institute of Mathematics and Statistics of the University of São Paulo, under supervision of Renata Wassermann. I started my MSc in February of 2015, and after my qualification exam, given the nature of the research project, I moved to the PhD program. My PhD qualification exam should be around August, 2017 and my thesis defence around January, 2019.

2. Introduction

Nowadays, ontologies written in languages such as OWL have become the standard method to share information in the Semantic Web (according to the W3C). These documents provide the formal description of a domain in such way that properties can be verified and extracted with the aid of computer programs called reasoners.

To aid the maintenance and evolution of the ontologies, there have been studies in the area of ontology revision (which we discuss in Section 3) that formalized how these tasks should be done. However, even this theory is not able to completely define these operations. In particular, some of them can have more than one outcome considered “correct” and the comparison among them may not be clear.

Additionally, the size of some ontologies (some with hundreds of thousands of axioms) can also complicate the change management. However, modularisation strategies were devised to allow engineers, users and programs to handle only a part of the ontology at a time. Thus, in this work we intend to improve the methods devised in the theory of ontology revision, using techniques from the field of ontology modularisation.

In the rest of this paper, we consider that an ontology $\mathcal{O}$ is a set of formulas (or axioms) in a description logic. Also, if X is a formula or set of formulas, we denominate the *signature* of X the set of all non-logical symbols (concepts, roles and individuals) that X employs.

3. Ontology Revision

The subject matter of Belief Revision (or Belief Change) is how rational agents change their beliefs in consequence of the acquisition. The field was born with the seminal work from Alchourrón, Gärdenfors and Makinson [Alchourrón et al. 1985], in the currently called "AGM theory". In their framework, the beliefs are represented by a set of logical formulas closed under logical consequence.

With subsequent works, such as [Hansson 1991, Hansson and Wassermann 2002, Flouris 2006, Ribeiro 2013], we are now able to apply the fundamentals of Belief Change to ontologies in description logics, which are incompatible with the original AGM model. In this work, we call the resulting theory by "Ontology Revision" (or "Ontology Change").

Most papers in the literature focus in one change operation, contraction, which consists in removing a formula from the consequences of the agent's beliefs. There are two major constructive methods to obtain ontology contraction operations: the partial meet approach and the kernel approach.

The partial meet approach consists of obtaining maximal subsets of the ontology $\mathcal{O}$ that do not imply a formula φ (which we call remainders), selecting the best among them and taking the intersection. Dually, the kernel approach relies on minimal subsets of an ontology $\mathcal{O}$ that do imply a formula φ (called kernels) and removes the least relevant elements from each of such set.

In both constructions, there can be multiple plausible outcomes, depending on how one chooses among the remainders or between the formulas inside each kernel. And one of our objectives in this work is to use the structure of the ontology, according to a particular modularity approach, to devise criteria to rank these elements.

4. Ontology Modularisation

To deal with the size of some ontologies, there have been many approaches to modularise ontologies, which consist in extracting parts of the ontology (modules) that are relevant for a given set of terms (signature). Some of these techniques even tried to use the modules to induce a structure in the ontology to aid both algorithms and engineers.

Our intent is to employ one of such approaches to solve two particular issues that arise when trying to revise ontologies, especially when they are too big: the multiple plausible outcomes of the ontology revision operations and the lack of computational resources to compute execute the procedures. Thus, we need a modularisation scheme that can impose an ordering of the axioms in the ontology.

Del Vescovo [2013] evaluates different modularity techniques in the literature with respect to their aptitude to identify structure in ontologies. Since all the approaches considered were inept to the task, the author devises a technique that fulfils the requirements proposed. This technique is called atomic decomposition and is founded on the notion of syntactic locality-based modules, which we present first.

4.1. Syntactic Locality-Based Modules

One of the most successful modularisation strategies is the syntactic variant of locality-based modules (syntactic LBMs) proposed in [Grau et al. 2008]. They are used nowadays in practice and are implemented thoroughly in the OWL API[1], mostly due to its logical properties and efficiency.

The general idea is that a module of an ontology $\mathcal{O}$ for a signature Σ, will consist of all axioms of $\mathcal{O}$ that are relevant (non-local) with respect to the signature Σ. The locality check is based on syntactic matching rules that reduce each axiom to a form where deciding whether relevant or not is simple (for further details of these rules, refer to [Grau et al. 2008, Del Vescovo 2013]).

4.2. Atomic Decomposition

While the syntactic LBMs may solve the problem of module extraction, they do not induce a structure in the ontology by themselves. Besides, there can be exponentially many subsets of the ontology that are modules for some signature.

These issues on representing ontology structure were addressed with a technique called atomic decomposition (AD) that is formalized by the following definitions:

Definition 4.1 (Co-occurrence Equivalence Relation [Del Vescovo 2013]). Let $\mathfrak{F}(\mathcal{O})$ be the set of all locality-based modules of an ontology $\mathcal{O}$. The relation $\approx$ is the binary relation over $\mathcal{O}$ defined to hold between two axioms $\alpha, \beta \in \mathcal{O}$ if, for all $\mathcal{M} \in \mathfrak{F}(\mathcal{O})$, $\alpha \in \mathcal{M}$ if and only if $\beta \in \mathcal{M}$.

Definition 4.2 (Atom [Del Vescovo 2013]). Let $\mathcal{O}$ be an ontology and $\approx$ the co-occurrence relation between atoms. We define an *atom* $\mathfrak{a}$ of $\mathcal{O}$ to be an equivalence class $[\alpha]_{\approx}$ for an axiom $\alpha \in \mathcal{O}$. The *set of atoms* of $\mathcal{O}$ is denoted by $\mathcal{A}(\mathcal{O})$.

---

[1] `http://owlapi.sourceforge.net/`

Definition 4.3 (Dependency Between Atoms [Del Vescovo 2013]). Let $\mathfrak{a}$ and $\mathfrak{b}$ be two atoms induced by $\approx$ over an ontology $\mathcal{O}$. We say that $\mathfrak{a}$ is *dependent* on $\mathfrak{b}$ (denoted as $\mathfrak{a} \succeq \mathfrak{b}$) if, for every module $\mathcal{M} \in \mathfrak{F}(\mathcal{O})$ such that $\mathfrak{a} \subseteq \mathcal{M}$, we have $\mathfrak{b} \subseteq \mathcal{M}$.

It is also important to remark that this relation establishes a partial order of the atoms in $\mathcal{A}(\mathcal{O})$. Given this fact, the strict part of this relation, denoted by $\succ$ is a strict partial ordered set (i.e., a poset). Finally, we define the atomic decomposition of an ontology $\mathcal{O}$ as the pair: $(\mathcal{A}(\mathcal{O}), \succ)$. Also, there are results which sustain the atomic decomposition as a succinct representation of all modules in an ontology [Del Vescovo 2013].

5. Proposal and Objectives

In this work, we aim to devise a solution to the problem of repairing and evolving big ontologies by combining the theory of ontology revision and modularisation techniques. In particular, we address both the computational problems associated with the size of the ontologies and that of the choice among the possible outcomes of a revision operation.

The principal contributions are: the software framework unifying an state-of-art modularisation strategies (the atomic decomposition) and the algorithms for ontology revision, the new criteria to automatically decide which of the revision outcomes is the best (without resorting to extralogical features) and the formal characterization of operations that follow the criteria devised.

6. Methodology

Following the previous implementations that will be extended in this work, the framework will also be written in Java and rely upon the OWL API. This framework will be used to run experiments that will sustain our studies and will be equipped with tools to record the processing time and memory spent during the most critical tasks.

The framework will consist of three loosely coupled components. One of them is the management of the input ontologies which will be responsible for generating use cases from real world ontologies (such as those obtained in the NCBO Bioportal[2]) and artificial ontologies which will be produced by the algorithm used in [Resina et al. 2014].

The second part of the framework will handle the structure of the ontology. In short, it will execute the following tasks: compute, store, retrieve and maintain the resulting atomic decompositions and provide syntactic LBMs. The principle is to use the same techniques discussed in [Klinov et al. 2012] for persistence and implement some of the extensions for the atomic decomposition in the literature [Del Vescovo 2013, Turlapati and Puligundla 2013, Martín-Recuerda and Walther 2014].

---

[2] `http://www.bioontology.org/`

The last part handles the ontology change operations, extending the framework devised by Cóbe [Cóbe 2014]. Besides using the algorithms already implemented, we also provide the implementation of the new operations devised.

6.1. Definition of Relevance Criteria

We use the atomic decomposition to devise valuation functions over the axioms of the ontology to decide among the remainder and kernel sets discussed in Section 3. These criteria will try to encode the entrenchment (the importance) of each axiom in the ontology. Afterwards, operators that employ these criteria will be implemented in the aforementioned framework.

6.2. Experiments

With the framework established, we generate a corpus of revision test cases and execute the experiments described below a sufficient number of times to obtain proper statistics for comparative purposes.

1. Measure the execution time and memory of the existing ontology revision algorithms (those implemented in [Cóbe 2014]) over the full ontologies and after extracting modules.
2. Repeat the previous experiments, but using the new criteria devised in Subsection 6.1.

6.3. Analysis

1. Compare the computational time and memory among the algorithms in the use-case corpus and evaluate the gains.
2. Study the differences in the results of the revision procedures between the existing approaches and the new operations devised (for instance verifying which axioms are kept or removed).
3. Observe how revision algorithms modify the atomic decomposition of ontologies. Such changes include: agglomeration of atoms and removal of dependencies.

6.4. Logical Characterization

From the analysis, we aim to characterize the operators that employ our relevance criteria with rationality postulates, that is, describe them using logical properties about their results, as is common in the area of Belief Revision [Alchourrón et al. 1985, Hansson 1991, Ribeiro 2013].

7. Conclusions

This document presents the PhD proposal to be developed during the next three semesters. We hope this will bring light for those interested in repairing and debugging ontologies in a way that is both computationally efficient and theoretically well-founded.

References

Alchourrón, C. E., Gärdenfors, P., and Makinson, D. (1985). On the logic of theory change: Partial meet contraction and revision functions. *The Journal of Symbolic Logic*, 50(2):510–530.

Cóbe, R. (2014). *Integração entre múltiplas ontologias: reúso e gerência de conflitos*. PhD thesis, Universidade de São Paulo.

Del Vescovo, C. (2013). *The modular structure of an ontology: Atomic decomposition*. PhD thesis, University of Manchester.

Flouris, G. (2006). *On Belief Change and Ontology Evolution*. PhD thesis, University of Crete.

Grau, B. C., Horrocks, I., Kazakov, Y., and Sattler, U. (2008). Modular Reuse of Ontologies: Theory and Practice. *Journal of Artificial Intelligence Research*, 31:273–318.

Hansson, S. O. (1991). Belief contraction without recovery. *Studia Logica*, 50(2):251–260.

Hansson, S. O. and Wassermann, R. (2002). Local Change. *Studia Logica*, 70(1):49–76.

Klinov, P., Del Vescovo, C., and Schneider, T. (2012). Incrementally Updateable and Persistent Decomposition of OWL Ontologies. In *Proceedings of the 9th OWL: Experiences and Directions Workshop, OWLED 2012*, volume 849 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Martín-Recuerda, F. and Walther, D. (2014). Axiom Dependency Hypergraphs for Fast Atomic Decomposition of Ontologies. In *Informal Proceedings of the 27th International Workshop on Description Logics, DL 2014*, volume 1193 of *CEUR Workshop Proceedings*, pages 299–310. CEUR-WS.org.

Resina, F., Ribeiro, M. M., and Wassermann, R. (2014). Algorithms for Multiple Contraction and an Application to OWL Ontologies. In *Proceedings of the 3rd Brazilian Conference on Intelligent Systems, BRACIS 2014*, pages 366–371. IEEE.

Ribeiro, M. M. (2013). *Belief Revision in Non-Classical Logics*. Springer Briefs in Computer Science. Springer, London.

Turlapati, V. K. C. and Puligundla, S. K. (2013). Efficient Module Extraction for Large Ontologies. In *Proceedings of the 4th International Conference on Knowledge Engineering and the Semantic Web, KESW 2013*, volume 394 of *Communications in Computer and Information Science*, pages 162–176. Springer.