Künstl Intell (2017) 31:107–109
DOI 10.1007/s13218-016-0460-x

Quantitative Methods for Similarity in Description Logics

Andreas Ecke[1]

Published online: 28 November 2016

Abstract Description logics (DLs) are a family of logic-based knowledge representation languages used to describe the knowledge of an application domain and reason about it in a formally well-defined way. However, all classical DLs have in common that they can only express exact knowledge, and correspondingly only allow exact inferences. In practice though, knowledge is rarely exact. Many definitions have exceptions or are vaguely formulated in the first place, and people might not only be interested in exact answers, but also in alternatives that are "close enough". We are interested in tackling how to express that something is "close enough", and how to integrate this notion into the formalism of DLs. To this end we employ the notion of similarity and dissimilarity measures, we will look at how useful measures can be defined in the context of DLs and two particular applications: Relaxed instance queries will use a similarity measure in order to not just give the exact answer to some query, but all answers that are reasonably similar. Prototypical definitions on the other hand use a measure of dissimilarity or distance between concepts in order to allow the definitions of and reasoning with concepts that capture not just those individuals that satisfy exactly the stated properties, but also those that are "close enough".

Keywords Description logics · Similarity measures · Query relaxation · Prototypes

✉ Andreas Ecke
andreas.ecke@tu-dresden.de

[1] Institute for Theoretical Computer Science, TU Dresden, Dresden, Germany

1 Introduction

This is a report on the doctoral dissertation [1], and briefly explains the questions and obtained results.

Description logics (DLs) are a family of logic-based knowledge representation languages used to describe the knowledge of an application domain and reason about it in a formally well-defined way [2]. DLs allow to describe the important classes of the knowledge domain as concepts, which formalize the necessary and sufficient conditions for individual objects to belong to that concept, expressed as a combination of atomic properties (concept names) and properties that refer to the relationship with other elements (role restrictions). In order to encode the conceptual knowledge, the user can then state how these concepts relate to each other, for example by giving superconcept–subconcept relationships. Additionally, DLs allow to express knowledge about individual objects, to which concepts they belong and how they relate to each other.

A variety of different DLs exist, from the inexpressive but efficient DL $\mathcal{EL}$, to the propositional complete DL $\mathcal{ALC}$, to very expressive DLs; differing in the set of properties one can use to express concepts, as well as the types of axioms available to describe the relations between concepts or individuals. However, all classical DLs have in common that they can only express exact knowledge, and correspondingly only allow exact inferences. Either we can infer that some individual belongs to a concept, or we can't, there is no in-between. In practice though, knowledge is rarely exact. Many definitions have exceptions or are vaguely formulated in the first place, and people might not only be interested in exact answers, but also in alternatives that are "close enough". In order to formally talk about how close different alternatives are, the notion of

semantic similarity and dissimilarity measures between DL concepts is used.

2 Similarity Measures

Fundamentally, similarity measures quantify how close two things are from a conceptual point of view [3]. They are thought of as one of the fundamental concepts of human reasoning. Many different approaches have evolved on how to measure similarity, but most rely on the same intuitions: The similarity between two objects increases with the commonalities that they share, while it decreases with the differences between them. Additionally, many similarity measures also have a notion of a maximal and minimal similarity, which intuitively occur when the objects have no differences or no commonalities, respectively.

We are interested in *semantic concept similarity measures*, which compare the meaning of two DL concepts, based on the background knowledge defined in the DL knowledge base. For concept similarity measures, the above intuitions can be formalized into a set of formal properties: a measure should be symmetric, invariant under equivalent concepts, and similarities 0 and 1 occur exactly if the concepts have no common subsumers or are equivalent, respectively. While many different similarity measures for DL concepts have been defined before, they all have drawbacks: In particular, no measure was able to completely use general (i.e., possibly cyclic) knowledge, while at the same time satisfying all formal properties states above.

In [4] we introduce the similarity measure $\sim_c$, which is a parameterizable concept similarity measure that works w.r.t. general $\mathcal{EL}$ knowledge bases. This measure is based on the similarity between the elements of the canonical interpretations of the the two concepts, which is computed by the interpretation similarity measure $\sim_i$. We show that $\sim_i$ (and thus $\sim_c$) is well-defined and computable in polynomial time, while satisfying all of the formal properties stated above [5].

3 Query Relaxation

Knowledge about individuals, the categories they belong to, and the relations between them, is usually stored in some kind of relational database, an XML file, an RDF triple store, a DL ABox, or similar storage formats. In order to access this data, one can formulate a query that describes which of the individuals one is interested in, by restricting for instance the categories or the relations to other individuals. A query answering system then selects all those individuals that satisfy the query and returns them as answers.

However, when specifying the query, one may not only be interested in the exact answers, which satisfy every single restriction that is part of the query; alternatives that do not completely satisfy the query, but most of it, may give interesting insights as well. The process of broadening the set of answers to include similar alternatives is often called query expansion or query relaxation, and has attracted a great deal of research.

Classical query relaxation approaches usually only work in presence of a very simple background ontology, like a concept hierarchy. Also, often the process of query relaxation can not be influenced. However, a way to specify which aspects of the query are less important and may be relaxed further can be exceedingly useful to control the query relaxation process based on user- or query-dependent preferences. In order to allow for parameterizable query relaxation working with general knowledge, we investigate the problem of instance queries relaxed by concept similarity measures. Formally, an individual is called a *relaxed instance* of concept Q w.r.t. a similarity measure, a DL knowledge base, and a threshold t, iff it is instance of a concept Q' that is similar to Q with a degree of at least t.

The first case we consider are arbitrary concept similarity measures and unfoldable TBoxes, which do not allow the definition of cyclic knowledge [6]. We show that the problem of computing all relaxed instances is decidable as long as the similarity measure used for the relaxation has certain properties (equivalence invariant, and the role-depth of the concepts can be bounded), but is also highly inefficient: It has non-elementary complexity. Afterwards we restrict to a single family of similarity measures, namely $\sim_c$, but allow for general $\mathcal{EL}$ TBoxes. In this setting we derive an NP algorithm for both checking whether an individual is a relaxed instance of the query concept, and for finding all answers to the relaxed instance query [4, 5].

In [7] we present an implementation for the case of general TBoxes: the ELASTIQ system. In order to show the usefulness of relaxed instance queries and the $\sim_c$ measure, we evaluate ELASTIQ on different ontologies. The results indicate that the answers that ELASTIQ returns are generally quite intuitive and the ability to tweak the results using the parameters of $\sim_c$ is very useful; the performance of ELASTIQ also seems to scale quite well with the size of the ontology. However, choosing a suitable threshold value and finer control over the parameters is often not quite as clear.

4 Prototypical Definitions

In practical applications one often cannot define all relevant concepts exactly by giving necessary and sufficient conditions. In fact, it has been argued that humans generally recognize categories by prototypes rather than

concepts. For example, it is impossible to define an abstract concept like "games" using just a set of necessary and sufficient conditions, such that the definition includes various things like video games, Olympic games, and jigsaw puzzles, while excluding all non-games [8]. Instead, other formalisms like prototypical definitions, where we can define games as things close to one or more prototypical objects, might be more useful.

In order to be used within a formal knowledge representation language with automated reasoning capabilities, such prototypes need to be equipped with a formal semantics. For this, we use ideas underlying Gärdenfors' conceptual spaces [9], where categories are explained in terms of convex regions defined using the distance from a focal point. To obtain a concrete representation language, we define prototype distance functions [10], which return for each element of an interpretation the distance of this element to a focal point; if the focal point is given as a specific individual, these functions could be seen as dissimilarity measures between elements. Prototype distance functions then allow the introduction of a new concept constructor for specifying prototypes: $P_{\leq t}(d)$ selects all elements of an interpretation for which the prototype distance function d returns at most distance t.

We give a concrete formalism for prototypical definitions for the DL $\mathcal{ALC}$, which uses weighted alternating parity tree automata (wapta) to specify prototype distance functions. In order to show that $\mathcal{ALCP}(\text{wapta})$, i.e., the DL $\mathcal{ALC}$ extended with prototypes defined by wapta, is decidable, we first show how unweighted automata can be used to decide concept satisfiability in $\mathcal{ALC}$. Afterwards, we present a cut-point construction that computes an unweighted automata $\mathcal{A}_{\leq n}$ which recognize exactly the cut-point language of a wapta $\mathcal{A}$ with threshold n, i.e., the language of all trees that have a distance of at most n. Finally, we show that one can combine the automaton used to decide concept satisfiability in $\mathcal{ALC}$ with the cut-point version of the prototype distance automata in order to decide the concept satisfiability problem in $\mathcal{ALCP}$. We show that, if numbers are encoded in unary, then reasoning in $\mathcal{ALCP}(\text{wapta})$ is ExpTime-complete, the same as classical $\mathcal{ALC}$ without prototypes [10].

5 Conclusions and Outlook

The three main contributions of our research are the concept similarity measure $\sim_c$, the definition of relaxed instances, and an approach to define and reason with prototypes using weighted automata. The thesis [1] explains all contributions in more depth, and provides an in-depth discussion and comparison to related approaches. There are many directions in which this work can be expanded. While the use of more expressive DLs is always useful to investigate, we believe that an extension of the query language for relaxed instance queries, and an investigation of alternative semantics for the weighted automata in the prototype approach would be particularly worthwhile.

Acknowledgements This work was supported by the DFG research training group "Graduiertenkolleg 1763 (QuantLA)".

References

1. Ecke A (2016) Quantitative methods for similarity in description logics. Dissertation, TU Dresden
2. Baader F, Calvanese D, McGuinness DL, Nardi D, Patel-Schneider PF (eds) (2003) The description logic handbook: theory, implementation, and applications. Cambridge University Press, NY, USA
3. Harispe S, Ranwez S, Janaqi S, Montmain J (2013) Semantic measures for the comparison of units of language, concepts or entities from text and knowledge base analysis. Computing research repository. arXiv:1310.1285
4. Ecke A, Peñaloza R, Turhan AY (2014) Answering instance queries relaxed by concept similarity. In: Proceedings of the fourteenth international conference on principles of knowledge representation and reasoning (KR'14). AAAI Press, Vienna, pp 248–257
5. Ecke A, Peñaloza R, Turhan AY (2015) Similarity-based relaxed instance queries. J Appl Logic 13(4, Part 1):480–508
6. Ecke A, Peñaloza R, Turhan AY (2013) Towards instance query answering for concepts relaxed by similarity measures. In: Workshop on weighted logics for AI (in conjunction with IJCAI'13). Beijing, China
7. Ecke A, Pensel M, Turhan AY (2015) ELASTIQ: Answering similarity-threshold instance queries in EL. In: Proceedings of the 28th international workshop on description logics (DL-2015). In: Calvanese D, Konev B (eds) CEUR workshop proceedings, vol 1350 (CEUR-WS.org)
8. Wittgenstein L (1953) Philosophical investigations. Basil Blackwell, Oxford (translated by GEM Anscombe)
9. Gärdenfors P (2000) Conceptual spaces: the geometry of thought. MIT Press, Cambridge, MA
10. Baader F, Ecke A (2016) Reasoning with prototypes in the description logic ALC using weighted tree automata. In: Proceedings of the 10th international conference on language and automata theory and applications (LATA 2016). Lecture notes in computer science. Springer-Verlag, pp 63–65

Andreas Ecke was born in Germany in 1987. He received his diploma in computer science from TU Dresden in 2012. During this time he spent a semester at the University of Auckland. From 2012 to 2015 he received a scholarship for the DFG graduate school "Quantitative Logics and Automata" at TU Dresden. His work focused on the use of similarity measures in Description Logics. He finished his doctoral degree in 2016.