Explaining Reasoning Results for OWL Ontologies with Evee

Christian Alrabbaa[1], Stefan Borgwardt[1], Tom Friese[1], Anke Hirsch[2], Nina Knieriemen[2], Patrick Koopmann[3], Alisa Kovtunova[1], Antonio Krüger[2], Alexej Popovič, and Ida Siahaan[1]

[1] Technische Universität Dresden, Germany
[2] Saarland University, Germany
[3] Vrije Universiteit Amsterdam, Netherlands

One of the advantages of logic-based artificial intelligence systems is that they are in principle *explainable by design*. However, to obtain also *explainability in practice*, appropriate tool support is needed. This paper focusses on reasoning with *OWL ontologies*. Such ontologies are used in many areas such as artificial intelligence, biology, medicine and semantic technologies. An ontology specifies terminological knowledge by providing definitions and logical axioms about *concepts*. A concept can be a *name* referring to a set of objects or a formal description of a set of objects. OWL (the Web Ontology Language) is the de-facto standard language for specifying ontologies [5], and is itself based on *description logics*, a family of first-order logics that are designed for specifying ontologies [3]. Using such a formalism based on logics has the advantage of a clearly defined semantics, which in turn allows the use of automated reasoning systems. These can be used to infer implicit information from the ontology itself (for instance, whether one concept describes a special case of another concept), and to infer new information from datasets about concrete objects.

Due to the expressivity of OWL and the size of realistic ontologies, inferences performed by a reasoner are not always transparent to end-users, which can make developing and maintaining, but also just understanding ontologies challenging. In particular, they may not understand why something *was* inferred (positive entailments), or why something was *not* inferred (missing entailments). The widely used ontology editor Protégé [11] comes with a simple explanation service that explains positive entailments using *justifications* [12,6], which are sets of statements from the ontology that are sufficient for producing the entailment. However, justifications do not explain *why* these statements produce the entailment, and they cannot explain missing entailments. To solve this issue, we developed Evee, a Java library that comes with a set of plugins for Protégé, which integrates various recently developed methods to fill this gap. Evee uses

1. *proofs* of different granularity to provide more detailed explanations of positive entailments, and which also take into account *user-specific vocabularies*;
2. *counterexamples* that describe situations that explain missing entailments, which can also be used to detect and add missing disjointness axioms; and
3. *abduction* to explain and repair *missing entailments* by providing sets of axioms that could be added to the ontology to produce the desired entailment.

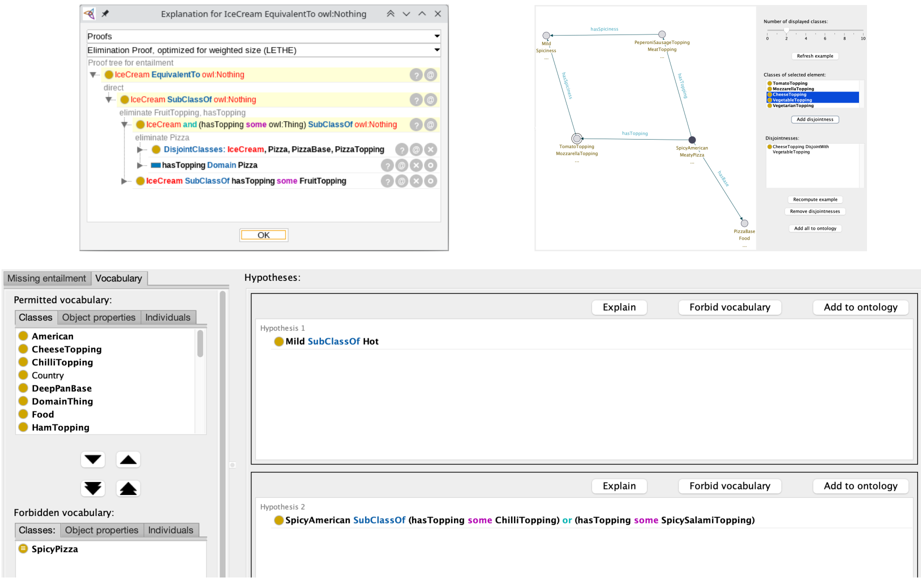

Fig. 1. Explaining IceCream EquivalentTo owl:Nothing with an elimination proof (top left) and missing entailment SpicyAmerican SubClassOf SpicyPizza with counterexamples (top right) and signature-based abduction (bottom).

Figure 1 illustrates the different types of explanation on bugs in a modified version of the pizza ontology,an ontology that is often used for educational purposes. For each of these explanation services, we implemented different methods: in the case of proofs, we have the more high-level *elimination proofs* [1], the more detailed proofs extracted from the reasoning system Elk [8,7], and for more expressive ontologies, detailed proofs extracted from the uniform interpolation tool Lethe [9]. Proofs can be optimized towards different criteria such as (weighted) size or depth. Our *counterexamples* may be complete, minimized models of a concept in question, focus on relevant parts of the model, or contrast with positive examples. For abduction, we rely on the recently developed systems Capi and Lethe-Abduction, which produce explanations that respectively satisfy the criteria *connection-minimalilty* [4] and *signature-based completeness* [10]. We evaluated the explanation services for positive and missing entailments in different user studies with students, ontology engineers and researchers. Our studies confirm that, while still in a prototypical state, our explanations can help understanding inferences and solve debugging tasks. Moreover, we found that there is no one-size-fits-all explanation service, but that rather different explanation services are preferred by different users and for different tasks.

Evee is available online,[4] and the full paper is accepted at the *KR in the Wild* track at KR 2024 [2].

[4] https://github.com/de-tu-dresden-inf-lat/evee

References

1. Alrabbaa, C., Baader, F., Borgwardt, S., Koopmann, P., Kovtunova, A.: Finding small proofs for description logic entailments: Theory and practice. In: Albert, E., Kovács, L. (eds.) LPAR 2020: 23rd International Conference on Logic for Programming, Artificial Intelligence and Reasoning. EPiC Series in Computing, vol. 73, pp. 32–67. EasyChair (2020). https://doi.org/10.29007/nhpp
2. Alrabbaa, C., Borgwardt, S., Friese, T., Hirsch, A., Knieriemen, N., Koopmann, P., Kovtunova, A., Krüger, A., Popovič, A., Siahaan, I.: Explaining reasoning results for owl ontologies with evee. In: Proceedings of KR (2024), to appear
3. Baader, F., Horrocks, I., Lutz, C., Sattler, U.: An Introduction to Description Logic. Cambridge University Press (2017). https://doi.org/10.1017/9781139025355, http://dltextbook.org/
4. Haifani, F., Koopmann, P., Tourret, S., Weidenbach, C.: Connection-minimal abduction in $\mathcal{EL}$ via translation to FOL. In: Blanchette, J., Kovács, L., Pattinson, D. (eds.) Automated Reasoning. pp. 188–207. Springer International Publishing, Cham (2022). https://doi.org/10.1007/978-3-031-10769-6_12
5. Hitzler, P., Krötzsch, M., Parsia, B., Patel-Schneider, P.F., Rudolph, S. (eds.): OWL 2 Web Ontology Language: Primer (Second Edition). W3C Recommendation, W3C (2012), https://www.w3.org/TR/owl2-primer/
6. Horridge, M.: Justification Based Explanation in Ontologies. Ph.D. thesis, University of Manchester, UK (2011), https://www.research.manchester.ac.uk/portal/files/54511395/FULL_TEXT.PDF
7. Kazakov, Y., Klinov, P., Stupnikov, A.: Towards reusable explanation services in protege. In: Artale, A., Glimm, B., Kontchakov, R. (eds.) Proceedings of the 30th International Workshop on Description Logics. CEUR Workshop Proceedings, vol. 1879. CEUR-WS.org (2017), http://ceur-ws.org/Vol-1879/paper31.pdf
8. Kazakov, Y., Krötzsch, M., Simancik, F.: The incredible ELK - from polynomial procedures to efficient reasoning with $\mathcal{EL}$ ontologies. J. Autom. Reason. **53**(1), 1–61 (2014). https://doi.org/10.1007/s10817-013-9296-3
9. Koopmann, P.: LETHE: Forgetting and uniform interpolation for expressive description logics. Künstliche Intell. **34**(3), 381–387 (2020). https://doi.org/10.1007/s13218-020-00655-w
10. Koopmann, P., Del-Pinto, W., Tourret, S., Schmidt, R.A.: Signature-based abduction for expressive description logics. In: Calvanese, D., Erdem, E., Thielscher, M. (eds.) Proceedings of the 17th International Conference on Principles of Knowledge Representation and Reasoning, KR. pp. 592–602 (2020). https://doi.org/10.24963/kr.2020/59
11. Musen, M.A.: The Protégé project: A look back and a look forward. AI Matters **1**(4), 4–12 (2015). https://doi.org/10.1145/2757001.2757003
12. Schlobach, S., Cornet, R.: Non-standard reasoning services for the debugging of description logic terminologies. In: Gottlob, G., Walsh, T. (eds.) IJCAI-03, Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence. pp. 355–362. Morgan Kaufmann (2003), http://ijcai.org/Proceedings/03/Papers/053.pdf