

Querying LUBM with Non-monotonic Features in Protégé using NoHR^{*}

Nuno Costa, Matthias Knorr and João Leite

NOVA LINCS, Departamento de Informática, Faculdade de Ciências e Tecnologia,
Universidade Nova de Lisboa

Abstract. The Protégé plug-in NoHR, which allows the user to combine an OWL 2 EL ontology with a set of non-monotonic (logic programming) rules – suitable e.g. to express defaults and exceptions – and query the combined knowledge base (KB), has recently been extended to OWL 2 QL in a non-trivial way. In this paper, we showcase this extension using an OWL 2 QL version of the well-known LUBM benchmark ontology that preserves all its reasoning features and, in addition, introduce meaningful non-monotonic rules. We can then query this extended knowledge base making use of the data sets that can be generated with LUBM. The demo will thus not only showcase the extension(s) of the tool NoHR, but also its usability with potentially large data sets.

1 Introduction

NoHR¹ is a plug-in for the ontology editor Protégé² that allows its users to query combinations of tractable Description Logic (DL) ontologies and non-monotonic rules in a top-down manner, and one of its crucial features is the fact that the (data) complexity of the combined approach remains tractable.

As originally introduced in [9], NoHR has been developed for $\mathcal{EL}_{\perp}^{\dagger}$ with the motivation that many current ontologies, such as the very large health care ontologies widely used in the area of medicine, e.g., SNOMED CT,³ are expressed in OWL 2 EL, one of the OWL 2 profiles [11], and its underlying description logic (DL) \mathcal{EL}^{++} [4]. Yet, due to their monotonic semantics, i.e., previously drawn conclusions persist when new additional information is adopted, DL-based ontology languages [3] are not suitable to model defaults and exceptions with a closed-world view, a frequently requested feature, e.g., when matching patient records to clinical trial criteria [12].

The tool itself builds on a formal framework [10], that is both very general and flexible for combining DL ontologies and non-monotonic rules. At the same time, tractability of reasoning for polynomial DLs is preserved, and top-down query procedures, such as $\text{SLG}(\mathcal{O})$ [1], can be applied to answer queries based only on the information relevant for the query, and without computing the entire model.

^{*} This work was partially supported by Fundação para a Ciência e a Tecnologia (FCT) under project PTDC/EIA-CCO/121823/2010, strategic project PEst/UID/CEC/04516/2013, and grant SFRH/BPD/86970/2012 (M. Knorr).

¹ <http://centria.di.fct.unl.pt/nohr/>

² <http://protege.stanford.edu>

³ <http://www.ihtsdo.org/snomed-ct/>

Other applications (see, e.g., [6,13]) require ontologies using DL constructors which are not covered by OWL 2 EL, such as concept and role negation or role inverses, as admitting these would raise its polynomial complexity [4]. Here, OWL 2 QL, which is tailored towards huge datasets, and the *DL-Lite* family [5,2] to which *DL-Lite_R*, the DL underneath OWL 2 QL belongs, is suitable, and has recently drawn a lot of attention in research and in applications. Therefore, in our ISWC 2015 Research Track paper [7], we have extended NoHR to OWL 2 QL in order to provide also such applications based on OWL 2 QL with the additional expressive power obtained from combining DL ontologies with non-monotonic rules.

In this paper, we will demonstrate the feasibility of this extension, building on an OWL 2 QL version of the well-known LUBM benchmark ontology [8] that preserves all its reasoning features and, in addition, introduce meaningful non-monotonic rules. We can then query this extended knowledge base making use of the data sets that can be generated with LUBM. The demo will thus not only showcase the extension(s) of the tool NoHR, but also its usability with potentially large data sets.

2 System Description

In this section, we briefly describe the architecture of our plug-in and point to some optimizations implemented w.r.t. the translation described in [7].

In general, the architecture of NoHR remains very similar to the one described in [9]. Namely, the input for our plug-in consists of an OWL file, which can be manipulated as usual in Protégé, and a rule file, which can be manipulated in a dedicated new tab. A second tab provides an interface for querying the combined KB and presenting the results. Whenever the first query is posed, the translator is started, and a joint (non-monotonic) rule set is created which is then loaded in XSB Prolog⁴ after which the query can be executed. The returned answers are processed into an answer table showing the variable substitutions for which we obtain true, undefined, or inconsistent valuations (or just shows the truth value for a ground query). Subsequently, further queries can be posed without repeated preprocessing, unless data in the ontology or the rules has been changed, in which case, a conditional recompilation is started for the changed component.

To allow the usage of OWL QL ontologies, changes were essentially made in the translator. First, since now two OWL profiles are supported we have introduced a switch that checks the profile of the loaded/edited ontology. If it is in OWL EL, then NoHR behaves as described in [9], i.e., the reasoner ELK is used to classify the ontology and return the inferred axioms to translator, which are then translated. Otherwise, we treat \mathcal{O} of the hybrid KB based on the direct translation described in [7] for OWL QL.

Notably, the translation only considers *DL-Lite_R* while OWL QL includes a number of additional constructs which often can be expressed in *DL-Lite_R*. To account for that, we first normalize such expressions to axioms in *DL-Lite_R*. This includes ignoring certain expressions, most of which do not contribute anything to derivations, e.g., `SubClassOf(B owl:Thing)`, while others make the ontology unsatisfiable, such

⁴ <http://xsb.sourceforge.net>

as `ClassAssertion(owl:Nothing a)`, although with no effect when querying the translated rules. Subsequently, unsatisfiable concepts and unsatisfiable and irreflexive roles are determined, after which the translation is performed, including optimizations to reduce the overall number of rules created during the translation.

The second group of optimizations is related to tabling in XSB, which contributes to help answering queries very efficiently in a top-down manner, and avoid infinite loops while querying. However, simply declaring all predicates to be tabled is very memory-consuming, so we reduced the number of tabled predicates without affecting loop detection. For example, only predicates that appear in any rule head and in any rule body need to be tabled. In addition, rules with an empty body (facts) can be ignored in the previous criterion, as these will not cause an infinite loop.

3 Demo and Dataset Details

In this section, we briefly describe the data set used in the demo. It builds on the well-known LUBM benchmark ontology [8], and we will describe the changes and additions we made.

First, as the LUBM ontology includes DL language features that are not covered in OWL QL, namely one transitive role and several concept definitions using conjunctions and general existential restrictions, we introduce a version of LUBM that is in OWL QL (plus one rule). Regarding the transitive role ‘is part of’, we replace this transitive declaration with the rule ‘is part of’(X, Z):-‘is part of’(X, Y), ‘is part of’(Y, Z). Regarding the concept definitions, we can replace, e.g., the concept definition

$$\text{Student} \equiv \text{Person} \sqcap \exists \text{takesCourse}.\text{Course}$$

by the following axioms:

$$\begin{array}{ll} \text{Student} \sqsubseteq \text{Person} & \exists \text{takesCourse}.\top \sqsubseteq \text{Student} \\ \text{Student} \sqsubseteq \exists \text{takesCourse}.\top & \exists \text{takesCourse}.\top \sqsubseteq \text{Course} \end{array}$$

Such transformation is obviously not equivalent, but unlike the standard reductions of LUBM to OWL QL, it preserves the answers to the 14 standard queries of LUBM and is thus in our opinion closer in spirit to the original LUBM than the former. This transformation does not work directly for the role head of because its range may be three (distinct) different classes, which is why we introduced two further specific roles. The original role remains associated to the class chair (as this the only usage appearing in data generated by LUBM), whereas two specific roles are associated to director and dean implicitly, which then permits the replacement as described above.

Second, we add a set of non-monotonic rules that describe a number of different use-cases of defaults and exceptions in this scenario. For example, some professor may become unavailable for teaching (in a given term), and we may determine replacement candidates from the relevant department based on the classes already given by everyone, but accounting for exceptions (e.g., sabbatical, long-term illness), but also account “interactively” for legislation changes, e.g., w.r.t. parental leaves. Such an example rule

for finding a replacement X at department Y would be:

Replacement(X, Y):-Professor(X), worksFor(X, Y), lowTeachingLoad(X),
not onSabbatical(X), not ill(X).

where lowTeachingLoad and necessary data on exceptions is also contained in the rules. Another use case is determining university staff that is performing well (or not) within the parameters given the data profile,⁵ for an evaluation process again with general regulations that include exceptions.

The demo itself proceeds in a similar fashion as shown in the video on the web page of NoHR,⁶ with the difference that it builds on the extension of the LUBM benchmark ontology described above, thus allowing to demonstrate the usability of the tool with a large dataset.

References

1. Alferes, J.J., Knorr, M., Swift, T.: Query-driven procedures for hybrid MKNF knowledge bases. *ACM Trans. Comput. Log.* 14(2), 1–43 (2013)
2. Artale, A., Calvanese, D., Kontchakov, R., Zakharyashev, M.: The *DL-Lite* family and relations. *J. Artif. Intell. Res. (JAIR)* 36, 1–69 (2009)
3. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F. (eds.): *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 3rd edn. (2010)
4. Baader, F., Brandt, S., Lutz, C.: Pushing the \mathcal{EL} envelope. In: Kaelbling, L.P., Saffioti, A. (eds.) *Proc. of IJCAI*. pp. 364–369. Professional Book Center (2005)
5. Calvanese, D., de Giacomo, G., Lembo, D., Lenzerini, M., Rosati, R.: Tractable reasoning and efficient query answering in description logics: The *DL-Lite* family. *Journal of Automated Reasoning* 39(3), 385–429 (2007)
6. Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Poggi, A., Rodriguez-Muro, M., Rosati, R., Ruzzi, M., Savo, D.F.: The MASTRO system for ontology-based data access. *Semantic Web* 2(1), 43–53 (2011)
7. Costa, N., Knorr, M., Leite, J.: Next Step for NoHR: OWL 2 QL. In: Arenas, M., Corcho, O., Simperl, E., Strohmaier, M., d’Aquin, M., Srinivas, K., Groth, P., Dumontier, M., Heflin, J., Thirunarayan, K., Staab, S. (eds.) *Proc. of ISWC*. Springer (2015)
8. Guo, Y., Pan, Z., Heflin, J.: LUBM: A benchmark for OWL knowledge base systems. *J. Web Sem.* 3(2-3), 158–182 (2005)
9. Ivanov, V., Knorr, M., Leite, J.: A query tool for \mathcal{EL} with non-monotonic rules. In: Alani, H., Kagal, L., Fokoue, A., Groth, P.T., Biemann, C., Parreira, J., Aroyo, L., Noy, N.F., Welty, C., Janowicz, K. (eds.) *Proc. of ISWC. LNCS*, vol. 8218, pp. 216–231. Springer (2013)
10. Knorr, M., Alferes, J.J., Hitzler, P.: Local closed world reasoning with description logics under the well-founded semantics. *Artif. Intell.* 175(9–10), 1528–1554 (2011)
11. Motik, B., Cuenca Grau, B., Horrocks, I., Wu, Z., Fokoue, A., Lutz, C. (eds.): *OWL 2 Web Ontology Language: Profiles*. W3C Recommendation (5 February 2013), available at <http://www.w3.org/TR/owl2-profiles/>

⁵ <http://swat.cse.lehigh.edu/projects/lubm/profile.htm>

⁶ <http://centria.di.fct.unl.pt/nohr/>

12. Patel, C., Cimino, J.J., Dolby, J., Fokoue, A., Kalyanpur, A., Kershenbaum, A., Ma, L., Schonberg, E., Srinivas, K.: Matching patient records to clinical trials using ontologies. In: Aberer, K., et al. (eds.) ISWC/ASWC. LNCS, vol. 4825, pp. 816–829. Springer (2007)
13. Savo, D.F., Lembo, D., Lenzerini, M., Poggi, A., Rodriguez-Muro, M., Romagnoli, V., Ruzzi, M., Stella, G.: Mastro at work: Experiences on ontology-based data access. In: Haarslev, V., Toman, D., Weddell, G.E. (eds.) Proceedings of (DL 2010). CEUR Workshop Proceedings, vol. 573. CEUR-WS.org (2010)