Separating Positive and Negative Data Examples by Concepts and Formulas: The Case of Restricted Signatures (Abstract)

Jean Christoph Jung[1], Carsten Lutz[1], Hadrien Pulcini[2], and Frank Wolter[2]

[1]University of Bremen, Germany [2]University of Liverpool, UK

There are several applications that fall under the broad term of supervised learning and seek to compute a logical expression that separates positive from negative examples given in the form of labeled data items in a knowledge base. A prominent example is concept learning for description logics (DLs) where the aim is to automatically construct a concept description that can then be used, for instance, in ontology engineering [4, 21, 20, 30, 8, 9, 27]. A further example is reverse engineering of database queries (also called query by example, QBE), which has a long history in database research [28, 29, 32, 31, 17, 1, 6, 18, 23] and which has also been studied in the presence of a DL ontology [12, 24]. Note that a closed world semantics is adopted for QBE in databases while an open world semantics is required when the data is assumed to be incomplete as in the presence of ontologies, but also, for example, in reverse engineering of SPARQL queries [2]. Another example is entity comparison in RDF graphs, where one aims to find meaningful descriptions that separate one entity from another [26, 25] and a final example is generating referring expressions (GRE) where the aim is to describe a single data item by a logical expression such as a DL concept, separating it from all other data items. GRE has originated in linguistics [19], but has recently received interest in DL-based ontology-mediated querying [7].

A fundamental problem common to all these applications is to decide whether a separating expression exists at all. There are several degrees of freedom in defining this problem. One concerns the negative examples: is it enough that they do not entail the separating formula (*weak separability*) or are they required to entail its negation (*strong separability*)? Another one concerns the question whether additional helper symbols are admitted in the separating formula (*projective separability*) or not (*non-projective separability*). The emerging family of problems has recently been investigated in [10, 14], concentrating on the case where the separating expression is a DL concept or formulated in a decidable fragment of first-order logic (FO) such as the guarded fragment (GF) and the guarded negation fragment (GNF). In the work reported about in this abstract [15], we add a signature Σ that is given as an additional input and require separating expressions to be formulated in Σ. This makes it possible to 'direct' separation towards expressions based on desired features and to exclude features that are not supposed to be used for separation such as gender and skin color. In the projective case, helper symbols from outside Σ are also admitted, but must be 'fresh' in that they cannot occur in the given knowledge base. The signature Σ brings the separation problem closer to the problem of deciding whether an ontology is a conservative extension of another ontology [13], also a form of separation, and to deciding the existence of uniform interpolants [22]. It turns out, in fact, that lower bounds for these problems can often be adapted to weak separability with signature. In constrast, for strong separability we observe a close connection to Craig interpolation.

We consider both weak and strong separability, generally assuming that the ontology is formulated in the same logic that is used for separation. We concentrate on combined complexity, that is, the input to the decision problems consists of the knowledge base that comprises an ABox and an ontology, the positive and negative examples in the form of lists of individuals (for DLs) or lists of tuples of individuals (for FO fragments that support more than one free variable), and the signature. In the following, we summarize our main results.

We start with weak projective separability in $\mathcal{ALCI}$, present a characterization in terms of Σ-homomorphisms that generalizes characterizations from [10, 14], and then give a decision procedure based on tree automata. This yields a 2ExpTime upper bound, and a matching lower bound is obtained by reduction from conservative extensions. In contrast, weak projective (and non-projective) separability in $\mathcal{ALCI}$ without a signature is only NExpTime-complete [10]. The non-projective case with signature remains open. We then show that weak separability is undecidable in any fragment of FO that extends GF (such as GNF) or $\mathcal{ALCFIO}$ (such as the two-variable fragment of FO with counting quantifiers). In both cases, the proof is by adaptation of undecidability proofs for conservative extensions [13, 11] and applies to both the projective and the non-projective case. This should be contrasted with the fact that weak separability is decidable and 2ExpTime-complete for GF and for GNF without a signature, both in the projective and in the non-projective case [14]. The decidability status of (any version of) separability in $\mathcal{ALCFIO}$ without a signature is open. It is known, however, that projective and non-projective weak separability without a signature are undecidable in the two-variable fragment of FO [14].

We then turn to strong separability. Here, the projective and the non-projective case coincide and will thus not be distinguished in what follows. We again start with $\mathcal{ALCI}$ for which we show 2ExpTime-completeness, and thus the increase in complexity that results from adding a signature is even more pronounced. In fact, strong separability without a signature is only ExpTime-complete in $\mathcal{ALCI}$ [14]. The proofs are rather different from those used in the weak case. The upper bound proof uses a characterization of non-separability in terms of the existence of a set of types that can be realized in a model of the ontology at elements that are all $\mathcal{ALCI}(\Sigma)$-bisimilar. A matching lower bound is proved by a reduction from the word problem of exponentially space bounded ATMs. Alternatively, the 2ExpTime upper bound can be proved by a polynomial time reduction to Craig interpolant existence in $\mathcal{ALCIO}$, the extension of $\mathcal{ALCI}$ with nominals. In fact, it has recently be shown that Craig interpolant existence in $\mathcal{ALCIO}$ is decidable in 2ExpTime [3].

For the guarded fragments GF and GNF, we show decidability in 3ExpTime and 2ExpTime-completeness, respectively. The upper bounds are proved by providing a polynomial time reduction to the respective Craig interpolant existence problems. Since GNF has the Craig interpolation property (CIP) [5], interpolant existence reduces to validity and is thus 2ExpTime-complete. GF does not enjoy the CIP and the 3ExpTime upper bound for interpolant existence has only been established recently [16].

Acknowledgments. Supported by the DFG Collaborative Research Center 1320 EASE - Everyday Activity Science and Engineering.

References

1. Arenas, M., Diaz, G.I.: The exact complexity of the first-order logic definability problem. ACM Trans. Database Syst. 41(2), 13:1–13:14 (2016)
2. Arenas, M., Diaz, G.I., Kostylev, E.V.: Reverse engineering SPARQL queries. In: Proc. of WWW. pp. 239–249 (2016)
3. Artale, A., Jung, J.C., Mazzullo, A., Ozaki, A., Wolter, F.: Living without Beth and Craig: Explicit definitions and interpolants in description logics with nominals (2020), available at https://arxiv.org/abs/2007.02736
4. Badea, L., Nienhuys-Cheng, S.: A refinement operator for description logics. In: Proc. of ILP. pp. 40–59 (2000)
5. Bárány, V., Benedikt, M., ten Cate, B.: Some model theory of guarded negation. J. Symb. Log. 83(4), 1307–1344 (2018)
6. Barceló, P., Romero, M.: The complexity of reverse engineering problems for conjunctive queries. In: Proc. of ICDT. pp. 7:1–7:17 (2017)
7. Borgida, A., Toman, D., Weddell, G.E.: On referring expressions in query answering over first order knowledge bases. In: Proc. of KR. pp. 319–328 (2016)
8. Bühmann, L., Lehmann, J., Westphal, P., Bin, S.: DL-learner - structured machine learning on semantic web data. In: Proc. of WWW. pp. 467–471 (2018)
9. Fanizzi, N., Rizzo, G., d'Amato, C., Esposito, F.: DLFoil: Class expression learning revisited. In: Proc. of EKAW. pp. 98–113 (2018)
10. Funk, M., Jung, J.C., Lutz, C., Pulcini, H., Wolter, F.: Learning description logic concepts: When can positive and negative examples be separated? In: Proc. of IJCAI. pp. 1682–1688 (2019)
11. Ghilardi, S., Lutz, C., Wolter, F.: Did I damage my ontology? A case for conservative extensions in description logics. In: Proc. of KR. pp. 187–197. AAAI Press (2006)
12. Gutiérrez-Basulto, V., Jung, J.C., Sabellek, L.: Reverse engineering queries in ontology-enriched systems: The case of expressive Horn description logic ontologies. In: Proc. of IJCAI-ECAI (2018)
13. Jung, J., Lutz, C., Martel, M., Schneider, T., Wolter, F.: Conservative extensions in guarded and two-variable fragments. In: Proc. of ICALP. pp. 108:1–108:14. Schloss Dagstuhl – LZI (2017)
14. Jung, J.C., Lutz, C., Pulcini, H., Wolter, F.: Logical separability of incomplete data under ontologies. In: Proc. of KR (2020)
15. Jung, J.C., Lutz, C., Pulcini, H., Wolter, F.: Separating positive and negative data examples by concepts and formulas: The case of restricted signature. In: https://arxiv.org/abs/2007.02736 (2020)

16. Jung, J.C., Wolter, F.: Living without Beth and Craig: Explicit definitions and interpolants in the guarded fragment (2020), available at http://arxiv.org/abs/2007.01597
17. Kalashnikov, D.V., Lakshmanan, L.V., Srivastava, D.: Fastqre: Fast query reverse engineering. In: Proc. of SIGMOD. pp. 337–350 (2018)
18. Kimelfeld, B., Ré, C.: A relational framework for classifier engineering. ACM Trans. Database Syst. 43(3), 11:1–11:36 (2018), https://doi.org/10.1145/3268931
19. Krahmer, E., van Deemter, K.: Computational generation of referring expressions: A survey. Computational Linguistics 38(1), 173–218 (2012)
20. Lehmann, J., Fanizzi, N., Bühmann, L., d'Amato, C.: Concept learning. In: Perspectives on Ontology Learning, pp. 71–91. AKA / IOS Press (2014)
21. Lehmann, J., Hitzler, P.: Concept learning in description logics using refinement operators. Machine Learning 78, 203–250 (2010)
22. Lutz, C., Wolter, F.: Foundations for uniform interpolation and forgetting in expressive description logics. In: Proc. of IJCAI. pp. 989–995. IJCAI/AAAI (2011)
23. Martins, D.M.L.: Reverse engineering database queries from examples: State-of-the-art, challenges, and research opportunities. Information Systems (2019)
24. Ortiz, M.: Ontology-mediated queries from examples: a glimpse at the DL-Lite case. In: Proc. of GCAI. pp. 1–14 (2019)
25. Petrova, A., Kostylev, E.V., Grau, B.C., Horrocks, I.: Query-based entity comparison in knowledge graphs revisited. In: Proc. of ISWC. pp. 558–575. Springer (2019)
26. Petrova, A., Sherkhonov, E., Grau, B.C., Horrocks, I.: Entity comparison in RDF graphs. In: Proc. of ISWC. pp. 526–541 (2017)
27. Sarker, M.K., Hitzler, P.: Efficient concept induction for description logics. In: Proc. of AAAI. pp. 3036–3043 (2019)
28. Tran, Q.T., Chan, C., Parthasarathy, S.: Query by output. In: Proc. of PODS. pp. 535–548. ACM (2009)
29. Tran, Q.T., Chan, C.Y., Parthasarathy, S.: Query reverse engineering. VLDB J. 23(5), 721–746 (2014)
30. Tran, T., Ha, Q., Hoang, T., Nguyen, L.A., Nguyen, H.S.: Bisimulation-based concept learning in description logics. Fundam. Inform. 133(2-3), 287–303 (2014)
31. Weiss, Y.Y., Cohen, S.: Reverse engineering spj-queries from examples. In: Proc. of PODS. pp. 151–166. ACM (2017)
32. Zhang, M., Elmeleegy, H., Procopiuc, C.M., Srivastava, D.: Reverse engineering complex join queries. In: Proc. of SIGMOD. pp. 809–820. ACM (2013)