

Learning Description Logic Concepts: When can Positive and Negative Examples be Separated? (Abstract)

Maurice Funk¹, Jean Christoph Jung¹, Carsten Lutz^{1,*}, Hadrien Pulcini², and Frank Wolter²

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

An important challenge for adopting ontologies in practical applications is the knowledge acquisition bottleneck, that is, the significant time and effort it takes to build the required ontologies. As a promising approach to help overcoming this difficulty, the varied field of ontology learning has received a lot of attention in the last two decades, see [15] for a recent overview. A prominent line of research within ontology learning is concerned with learning description logic (DL) concepts from positive and negative examples, given an already available DL ontology that contains background knowledge [12, 14, 16, 20, 7]. Applications include the support of ontology development and the construction of concept based classifiers [4, 18]. The precise formulation is as follows. Given a knowledge base (KB) $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ and designated sets of individuals P and N from \mathcal{A} that serve as positive and negative examples, find a concept C formulated in a DL \mathcal{L}_S that *separates* the positive from the negative examples, that is, $\mathcal{K} \models C(a)$ for all $a \in P$ and $\mathcal{K} \not\models C(a)$ for all $a \in N$. In addition to separation, one also wants to achieve that the learned concept C generalizes the positive examples in a meaningful way, classifying new examples accordingly.

As a prominent system for DL concept learning, we mention DL LEARNER. It encompasses several learning algorithms that support a range of DLs, including expressive ones such as \mathcal{ALC} and \mathcal{ALCQ} , Horn DLs such as \mathcal{EL} , and even full OWL 2 [5, 4]. Like competing systems such as DL-FOIL, YINYANG, and PFOIL-DL [7, 10, 19], DL LEARNER uses carefully crafted *refinement operators* [1, 13, 14] along with various heuristics to learn concepts that provide an as good as possible generalization of the examples, avoiding overfitting. If possible, refinement operators are designed so that the resulting algorithm terminates on any input and is complete in the sense that whenever there is a concept that separates the positive and negative examples in the input, then such a concept is indeed learned.

In the paper reported about in this abstract [8], we investigate the fundamental question of when a separating concept exists for a *learning instance* (\mathcal{K}, P, N) , considering the most popular choices of DLs for the TBox language \mathcal{L}_T and the separation language \mathcal{L}_S . Our main contributions are model-theoretic characterizations that give important insight into when this is the case and a precise analysis of the computational complexity of separability viewed as a decision problem, which we refer to as $(\mathcal{L}_T, \mathcal{L}_S)$ *concept separability* and as \mathcal{L} *concept separability* when $\mathcal{L}_T = \mathcal{L}_S = \mathcal{L}$. We also consider *concept definability*, the spe-

* Supported by DFG CRC 1320 Ease.

cial case of concept separability in which $P \cup N$ comprises all individuals from \mathcal{A} . All our complexity results actually hold for both separability and definability.

We believe that these problems are relevant both from a practical and from a theoretical perspective. In fact, complexity lower bounds for concept separability point to an inherent complexity that no practical system that aims for completeness can avoid. Undecidability results even mean that there can be no practical learning system that is both terminating and complete. From the viewpoint of machine learning theory, concept separability corresponds to the existence of a consistent hypothesis, the most fundamental problem for exploring the version space [9]. The associated decision problem is often called the *consistency problem*, and it is known to be closely related to PAC learnability [17, 11].

We cover the expressive DLs \mathcal{ALC} , \mathcal{ALCI} , \mathcal{ALCQ} , and \mathcal{ALCQI} as well as the Horn DLs \mathcal{EL} and \mathcal{ELI} . For the former, overfitting is a risk because the disjunction operator available in such DLs enables the construction of separating concepts that do not provide the desired generalization of the positive examples. Nevertheless, most practical systems such as DL LEARNER work with expressive DLs and avoid overfitting by using appropriate refinement operators. Horn DLs do not admit disjunction and therefore are not prone to overfitting. On the other hand, they provide less separating power and, as we show, tend to incur higher computational (worst-case) cost for learning.

For expressive DLs, we start with initial characterizations in terms of (a form of) bisimulations and then proceed to more refined characterizations based on homomorphisms. Interestingly and unexpectedly to us, these establish a tight link between concept separability and the evaluation of ontology-mediated queries (OMQs) based on unions of ‘rooted’ conjunctive queries [6, 2]. We use this link to obtain complexity upper and lower bounds. In fact, \mathcal{L} concept separability is NEXPTIME-complete for $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCI}, \mathcal{ALCQ}\}$ while \mathcal{ALCQI} concept separability is only EXPTIME-complete. This refers to combined complexity where all components of the learning instance are part of the input. We also study data complexity where the ABox is the only input while the TBox is fixed. In all expressive DLs above, concept separability is Σ_2^p -complete in data complexity.

For Horn DLs, we establish characterizations based on products of universal models and simulations. Based on these, we show that $(\mathcal{L}_T, \mathcal{EL})$ concept separability is EXPTIME-complete for $\mathcal{L}_T \in \{\mathcal{EL}, \mathcal{ELI}\}$, both in combined complexity and in data complexity. We find the high data complexity of this problem rather remarkable. We also prove that \mathcal{ELI} concept separability is undecidable, a result that came as a surprise to us.

We finally consider a mix of expressive DLs and Horn DLs, that is, $(\mathcal{L}_T, \mathcal{L}_S)$ concept separability where \mathcal{L}_T is any of the expressive DLs mentioned above and \mathcal{L}_S is \mathcal{EL} or \mathcal{ELI} . These problems again turn out to be undecidable, thus ruling out terminating and complete learning systems. The proof exploits a connection to a certain version of query based conservative extensions between \mathcal{ALC} KBs [3].

We also consider a stronger version of concept separability that is also considered in the literature requires that $\mathcal{K} \models \neg C(a)$ for all $a \in N$, rather than only $\mathcal{K} \not\models C(a)$.

References

1. Badea, L., Nienhuys-Cheng, S.: A refinement operator for description logics. In: Proceedings of ILP. pp. 40–59 (2000)
2. Bienvenu, M., ten Cate, B., Lutz, C., Wolter, F.: Ontology-based data access: A study through disjunctive datalog, CSP, and MMSNP. *ACM Trans. Database Syst.* 39(4), 33:1–33:44 (2014)
3. Botoeva, E., Kontchakov, R., Lutz, C., Ryzhikov, V., Wolter, F., Zakharyashev, M.: Query inseparability for ALC ontologies. *Artif. Intell.* 272, 1–51 (2019)
4. Bühmann, L., Lehmann, J., Westphal, P.: DL-learner - A framework for inductive learning on the semantic web. *J. Web Sem.* 39, 15–24 (2016)
5. Bühmann, L., Lehmann, J., Westphal, P., Bin, S.: DL-learner structured machine learning on semantic web data. In: Proceedings of WWW. pp. 467–471 (2018)
6. Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Rosati, R.: Data complexity of query answering in description logics. *Artificial Intelligence* 195, 335–360 (2013)
7. Fanizzi, N., Rizzo, G., d’Amato, C., Esposito, F.: Difoil: Class expression learning revisited. In: Proceedings of EKAW. pp. 98–113 (2018)
8. Funk, M., Jung, J.C., Lutz, C., Pulcini, H., Wolter, F.: Learning description logic concepts: When can positive and negative examples be separated. In: Proceedings of IJCAI (2019)
9. Hirsh, H., Mishra, N., Pitt, L.: Version spaces and the consistency problem. *Artificial Intelligence* 156(2), 115–138 (2004)
10. Iannone, L., Palmisano, I., Fanizzi, N.: An algorithm based on counterfactuals for concept learning in the semantic web. *Appl. Intell.* 26(2), 139–159 (2007)
11. Kietz, J.: Some lower bounds for the computational complexity of inductive logic programming. In: Proceedings of ECML. pp. 115–123 (1993)
12. Lehmann, J., Fanizzi, N., Bühmann, L., d’Amato, C.: Concept learning. In: Lehmann, J., Voelker, J. (eds.) *Perspectives on Ontology Learning*, pp. 71–91. AKA / IOS Press (2014)
13. Lehmann, J., Haase, C.: Ideal downward refinement in the \mathcal{EL} description logic. In: Proceedings of ILP. pp. 73–87 (2009)
14. Lehmann, J., Hitzler, P.: Concept learning in description logics using refinement operators. *Machine Learning* 78(1-2), 203–250 (2010)
15. Lehmann, J., Völker, J.: *Perspectives on Ontology Learning, Studies on the Semantic Web*, vol. 18. IOS Press (2014)
16. Lisi, F.A.: A formal characterization of concept learning in description logics. In: Proceedings of DL Workshop (2012)
17. Pitt, L., Valiant, L.G.: Computational limitations on learning from examples. *J. ACM* 35(4), 965–984 (1988), <https://doi.org/10.1145/48014.63140>
18. Sarker, M.K., Xie, N., Doran, D., Raymer, M., Hitzler, P.: Explaining trained neural networks with semantic web technologies: First steps. In: Proceedings of NeSy (2017)
19. Straccia, U., Mucci, M.: pFOIL-DL: Learning (fuzzy) \mathcal{EL} concept descriptions from crisp OWL data using a probabilistic ensemble estimation. In: Proceedings of SAC. pp. 345–352. ACM (2015)
20. 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)