Probabilistic Description Logics: Reasoning and Learning

Riccardo Zese

Dipartimento di Ingegneria – University of Ferrara
Via Saragat 1, I-44122, Ferrara, Italy
`riccardo.zese@unife.it`

The last decade has seen an exponential increase in the popularity of the Semantic Web. However, given the nature of the domains usually modeled in such scenario and the origin of available data, the interest for the development of methods for combining probability with Description Logics (DLs) has been exponentially increased as well.

A possible probabilistic semantics for DLs is DISPONTE [3, 5], which applies to them the distribution semantics, one of the most prominent semantics in probabilistic logic programming. DISPONTE allows to annotate axioms with a probability, interpreted as epistemic probability, indicating the degree of our belief in the truth of the corresponding axiom.

Prob-$\mathcal{ALC}$ considers only epistemic probabilities, while CR$\mathcal{ALC}$ extends $\mathcal{ALC}$ by allowing only statistical probabilities. In both these approaches the probability can be assigned to a limited set of axioms, differently from DISPONTE where every axiom can be probabilistic. P-$\mathcal{SHIQ}(\mathbf{D})$ uses probabilistic lexicographic entailment from probabilistic default reasoning and allows to annotate with a probabilistic interval both assertional and terminological axioms. $\mathcal{BEL}$ exploits Bayesian networks to extend the $\mathcal{EL}$ DL, while Probabilistic Datalog$^{\pm}$ uses Markov networks.

Several algorithms have been proposed for supporting the development of the Semantic Web. Efficient DL reasoners are able to extract implicit information from the modeled ontologies. Despite the availability of many DL reasoners, the number of probabilistic reasoners is quite small. BUNDLE [3, 5] is a reasoner able to compute the probability of queries w.r.t. DISPONTE DL KBs. It implements the tableau algorithms and returns the set of all explanations for the query, then represented with a Binary Decision Diagram (BDD), i.e., a tree representing a boolean formula, used for computing the probability.

However, some tableau expansion rules are non-deterministic forcing to explore all the non-deterministic choices to compute the set of all explanations for the query. This non-determinism can be managed with Prolog language. Thus, we developed TRILL [6, 5] which implements the tableau algorithm in Prolog to perform inference over DISPONTE DLs. We also developed TRILL^{P} [6, 5], which builds a monotone Boolean formula, called "pinpointing formula", instead of the set of explanations, which compactly represents them and can be directly translated into a BDD. Finally, TORNADO builds BDDs instead of pinpointing formulas during the inference process. TRILL, TRILL^{P} and TOR-

NADO are available at `http://trill.ml.unife.it` in the web service "TRILL on SWISH".

Other examples are PRONTO, which follows P-$\mathcal{SHIQ}(\mathbf{D})$ semantics and BORN following $\mathcal{BEL}$ semantics. A completely different approach addresses reasoning for Datalog$^{\pm}$ ontologies with an Abductive Logic Programming framework named $\mathcal{S}$CIFF, with existential and universal variables, and Constraint Logic Programming constraints in rule heads.

The correct values of the axioms' probabilities are unfortunately difficult to set, since they depend on many different factors. Therefore, it is necessary to develop systems able to automatically learn such values. Moreover, often KBs are incomplete or poorly structured, requiring systems able to correct erroneous information and learn new definitions. We developed EDGE [2] that learns the parameters of a DISPONTE KB from the information available in the domain. It exploits BUNDLE for building the BDDs representing explanations for the input examples and an Expectation Maximization algorithm to define probability values. We also developed LEAP [4], which combines EDGE with the learning system CELOE, in order to learn the structure of a DISPONTE KB by building new axioms. EDGE is used to learn the parameters of the KB. A different approach is used in Goldminer where association rules are exploited to define probabilistic terminological axioms.

However, nowadays most of the KBs are defined following the vision of Big Data and *Linked Open Data*. Thus, they require the implementation of algorithms exploiting parallelization and cloud computing to handle such big amount of data. Therefore, we extended EDGE and LEAP by developing EDGE^{MR} [2] and LEAP^{MR} [1], which distribute the work load.

References

1. Cota, G., Zese, R., Bellodi, E., Lamma, E., Riguzzi, F.: Structure learning with distributed parameter learning for probabilistic ontologies. In: Doctoral Consortium of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECMLPKDD 2015). pp. 75–84 (2015)
2. Cota, G., Zese, R., Bellodi, E., Riguzzi, F., Lamma, E.: Distributed parameter learning for probabilistic ontologies. In: 25th International Conference on Inductive Logic Programming (ILP 2015) (2015)
3. Riguzzi, F., Bellodi, E., Lamma, E., Zese, R.: Probabilistic description logics under the distribution semantics. Semantic Web 6(5), 447–501 (2015)
4. Riguzzi, F., Bellodi, E., Lamma, E., Zese, R., Cota, G.: Learning probabilistic description logics. In: Uncertainty Reasoning for the Semantic Web III, pp. 63–78. LNCS, Springer International Publishing, Berlng, Heidelberg (2014)
5. Zese, R.: Probabilistic Semantic Web, Studies on the Semantic Web, vol. 28. IOS Press, Amsterdam (2017)
6. Zese, R., Bellodi, E., Riguzzi, F., Cota, G., Lamma, E.: Tableau reasoning for description logics and its extension to probabilities. Ann. Math. Artif. Intell. pp. 1–30 (2016)