StaRAI or StaRDB?

A Tutorial on Statistical Relational AI

Tanya Braun[1]

Abstract: This tutorial aims at connecting databases and statical relational AI (StaRAI), demonstrating how database systems can benefit from methods developed within StaRAI, e.g., for implementing efficient systems combining databases and StaRAI. Thus, the goal of this tutorial is two-fold: (i) Present an overview of methods within StaRAI. (ii) Provide a forum to members of both communities for exchanging ideas.

Keywords: Statistical Relational AI, Probabilistic Relational Models, Probabilistic Inference

1 Introduction

In recent years, a need for compact representations of large relational databases became apparent, e.g., in natural language understanding or decision making. Using inductive logic programming (ILP), one can build a model of a database, allowing for a crisp reproduction of data. Another idea is to build a so called factor graph model of data and introduce a probability distribution to reproduce data approximately. Such a model defines an intensional representation of a probabilistic database. A factor graph model uses parameterised variables similarly to the variables in ILP to compactly represent relations and objects. Grounding such a model incurs an exponential blowup and makes inference infeasible. Instead of grounding out a model, one can answer queries on the model directly and in a scalable way.

This tutorial aims at connecting databases and statical relational AI (StaRAI), demonstrating how database systems can benefit from methods developed within StaRAI with the goal of implementing efficient systems for probabilistic inference. Thus, the goal of this tutorial is two-fold: (i) Present an overview of methods within StaRAI. (ii) Provide a forum to members of both communities for exchanging ideas.

This tutorial provides an overview of the approaches towards probabilistic relational modelling, looking at applications, semantics, and inference problems. The main part dives into algorithms developed to scale inference in probabilistic relational models, highlighting where database systems connect in our quest for efficient implementations for large data sets. Slides are available at `https://www.ifis.uni-luebeck.de/index.php?id=597`.

[1] Universität zu Lübeck, Institut für Informationssysteme, Ratzeburger Allee 160, 23562 Lübeck, braun@ifis.uni-luebeck.de

2 Probabilistic Relational Modelling

Applications of probabilistic relational modelling range from information retrieval to predicting network attacks, often requiring some form of query answering (QA). As early as 1995, Fuhr presents Probabilistic Datalog for information retrieval [Fu95]. In 2017, Muñoz-González et al. use Bayesian networks for network analysis [Mu17]. While the range of application areas is wide, the areas have all in common that QA requires proper probabilistic reasoning which brings along scalability issues, especially under grounding semantics [Sa95]. In a complementary approach, one is interested to lift ground instances into a first-order or template representation and answer queries exploiting the first-order structure during calculations [Po03]. Lifted representations allow for modelling relations between objects under uncertainty.

Of course, there is not only the question of how to model a scenario but what semantics to link with a model. The above mentioned grounding semantics defines a discrete joint distribution based on factors. Various frameworks, such as the above mentioned Probabilistic Datalog or ProbLog [RKT07] use grounding semantics, including lifted approaches [Po03, RD06]. For continuous domains, probabilistic soft logic defines a density function using a log-linear model [BMG10]. Maximum-entropy semantics allow for partially specifying discrete joints, which are then completed uniformly [Th10].

The inference problems full into two categories, static or dynamic, where dynamic refers to modelling a sequential or temporal process. In the static case, the inference tasks consist of (i) projection (margins), (ii) most-probable explanation (MPE), and (iii) maximum a posteriori (MAP). In the dynamic case, the inference problems are (i) filtering (present), (ii) prediction (future), (iii) hindsight (past), and (iv) MPE/MAP (temporal sequence). The main part, regarding scalability through lifting, focuses on solving static and dynamic inference tasks within the parfactor modelling framework first introduced by Poole [Po03].

3 Scalability by Lifting

Lifting for inference has lead to the development of a range of representation formalisms such as parfactor models [Po03] and Markov logic networks [RD06], lifting propositional representations, e.g., Bayesian networks, factor graphs, or Markov networks. There exist approaches for learning lifted representations, a prominent one being the colouring algorithm by Ahmadi et al. [Ah13]. Learning lifted representations is a subject deserving of a tutorial of its own. This tutorial concentrates on efficient inference algorithms for lifted representations.

Inference algorithms take a lifted representation as input and answer queries efficiently by exploiting the first-order structures in the representations. Similar to lifting propositional models, researchers have lifted algorithms that work for propositional models to work for lifted models. From a more logical inference perspective, there exist lifted versions [VMD14] of knowledge compilation [DM02] or theorem proving [GD11], all based on weighted model counting. Approximate lifted inference include lifted versions of belief propagation [SD08, Ah13] as well as lifted sampling approaches [GD11, FV18].

From the area of probabilistic exact inference, variable elimination (VE) is a standard algorithm for answering queries on static models [ZP94]. The junction tree algorithm (JT) builds a helper structure, called a junction tree, to efficiently answer a set of queries on a static model, incorporating VE as a subroutine [LS88]. The interface algorithm (IA) uses the junction tree idea to formalise an efficient algorithm for answering a set of queries on a dynamic model [Mu02]. For each algorithm, lifted counterparts exist, allowing for runtimes no longer depending exponentially on domain sizes, i.e., number of objects in a model. Lifted VE (LVE) avoids duplicate calculations by performing a calculation once for a representative instance and then incorporating the isomorphic instances [Po03, dSBAR05, Mi08, Ta13]. The lifted junction tree algorithm (LJT) builds a lifted junction tree representation, which enables LJT to use LVE as a subroutine [BM16]. The lifted dynamic junction tree algorithm is based on IA and LJT, combining the idea behind the interface algorithm and lifted junction trees [GBM18].

In this tutorial, we take a look at the given formalisms and algorithms through examples, investigating possible links to database systems in a quest for an efficient implementation of StaRAI algorithms in database systems.

References

[Ah13] Ahmadi, Babak; Kersting, Kristian; Mladenov, Martin; Natarajan, Sriraam: Exploiting Symmetries for Scaling Loopy Belief Propagation and Relational Training. Machine Learning, 92(1):91–132, 2013.

[BM16] Braun, Tanya; Möller, Ralf: Lifted Junction Tree Algorithm. In: Proc. of KI 2016: Advances in Artificial Intelligence. Springer, pp. 30–42, 2016.

[BMG10] Bröcheler, Matthias; Mihalkova, Lilyana; Getoor, Lise: Probabilistic Similarity Logic. In: UAI-10 Proc. of the 26th Conf. on Uncertainty in Artificial Intelligence. pp. 73–82, 2010.

[DM02] Darwiche, Adnan; Marquis, Pierre: A Knowledge Compilation Map. Journal of Artificial Intelligence Research, 17(1):229–264, 2002.

[dSBAR05] de Salvo Braz, Rodrigo; Amir, Eyal; Roth, Dan: Lifted First-order Probabilistic Inference. In: IJCAI-05 Proc. of the 19th International Joint Conf. on Artificial Intelligence. IJCAI Organization, pp. 1319–1325, 2005.

[Fu95] Fuhr, Norbert: Probabilistic Datalog - A Logic for Powerful Retrieval Methods. In: SIGIR-95 Proc. of the 18th Annual International ACM SIGIR Conf. on Research and Development in Information Retrieval. pp. 282–290, 1995.

[FV18] Friedman, Tal; Van den Broeck, Guy: Approximate Knowledge Compilation by Online Collapsed Importance Sampling. In: NIPS-18 Advances in Neural Information Processing Systems 31. Curran Associates, Inc., pp. 8034–8044, 2018.

[GBM18] Gehrke, Marcel; Braun, Tanya; Möller, Ralf: Lifted Dynamic Junction Tree Algorithm. In: Proc. of the International Conf. on Conceptual Structures. Springer, pp. 55–69, 2018.

[GD11] Gogate, Vibhav; Domingos, Pedro: Probabilistic Theorem Proving. In: UAI-11 Proc. of the 27th Conf. on Uncertainty in Artificial Intelligence. pp. 256–265, 2011.

[LS88] Lauritzen, Steffen L.; Spiegelhalter, David J.: Local Computations with Probabilities on Graphical Structures and Their Application to Expert Systems. Journal of the Royal Statistical Society. Series B: Methodological, 50:157–224, 1988.

[Mi08] Milch, Brian; Zettelmoyer, Luke S.; Kersting, Kristian; Haimes, Michael; Kaelbling, Leslie Pack: Lifted Probabilistic Inference with Counting Formulas. In: AAAI-08 Proc. of the 23rd AAAI Conf. on Artificial Intelligence. AAAI Press, pp. 1062–1068, 2008.

[Mu02] Murphy, Kevin Patrick: Dynamic Bayesian Networks: Representation, Inference and Learning. PhD thesis, University of California, Berkeley, 2002.

[Mu17] Muñoz-González, Luis; Sgandurra, Daniele; Barrère, Martín; Lupu, Emil C.: Exact Inference Techniques for the Analysis of Bayesian Attack Graphs. IEEE Transactions on Dependable and Secure Computing, PP(99):1–14, 2017.

[Po03] Poole, David: First-order Probabilistic Inference. In: IJCAI-03 Proc. of the 18th International Joint Conf. on Artificial Intelligence. IJCAI Organization, pp. 985–991, 2003.

[RD06] Richardson, Matthew; Domingos, Pedro: Markov Logic Networks. Machine Learning, 62(1-2):107–136, 2006.

[RKT07] Raedt, Luc De; Kimmig, Angelika; Toivonen, Hannu: ProbLog: A Probabilistic Prolog and its Application in Link Discovery. In: IJCAI-07 Proc. of 20th International Joint Conf. on Artificial Intelligence. IJCAI Organization, pp. 2062–2467, 2007.

[Sa95] Sato, Taisuke: A Statistical Learning Method for Logic Programs with Distribution Semantics. In: Proc. of the 12th International Conf. on Logic Programming. MIT Press, pp. 715–729, 1995.

[SD08] Singla, Parag; Domingos, Pedro: Lifted First-order Belief Propagation. In: AAAI-08 Proc. of the 23rd AAAI Conf. on Artificial Intelligence. AAAI Press, pp. 1094–1099, 2008.

[Ta13] Taghipour, Nima; Fierens, Daan; Davis, Jesse; Blockeel, Hendrik: Lifted Variable Elimination: Decoupling the Operators from the Constraint Language. Journal of Artificial Intelligence Research, 47(1):393–439, 2013.

[Th10] Thimm, Matthias; Finthammer, Marc; Loh, Sebastian; Kern-Isberner, Gebriele; Beierle, Christoph: A System for Relational Probabilistic Reasoning on Maximum Entropy. In: FLAIRS-10 Proc. of the 23rd International Florida Artificial Intelligence Research Society Conf. pp. 116–121, 2010.

[VMD14] Van den Broeck, Guy; Meert, Wannes; Darwiche, Adnan: Skolemisation for Weighted First-order Model Counting. In: KR-14 Proc. of the 14th International Conf. on Principles of Knowledge Representation and Reasoning. AAAI Press, pp. 111–120, 2014.

[ZP94] Zhang, Nevin L.; Poole, David: A Simple Approach to Bayesian Network Computations. In: Proc. of the 10th Canadian Conf. on Artificial Intelligence. Springer, pp. 171–178, 1994.