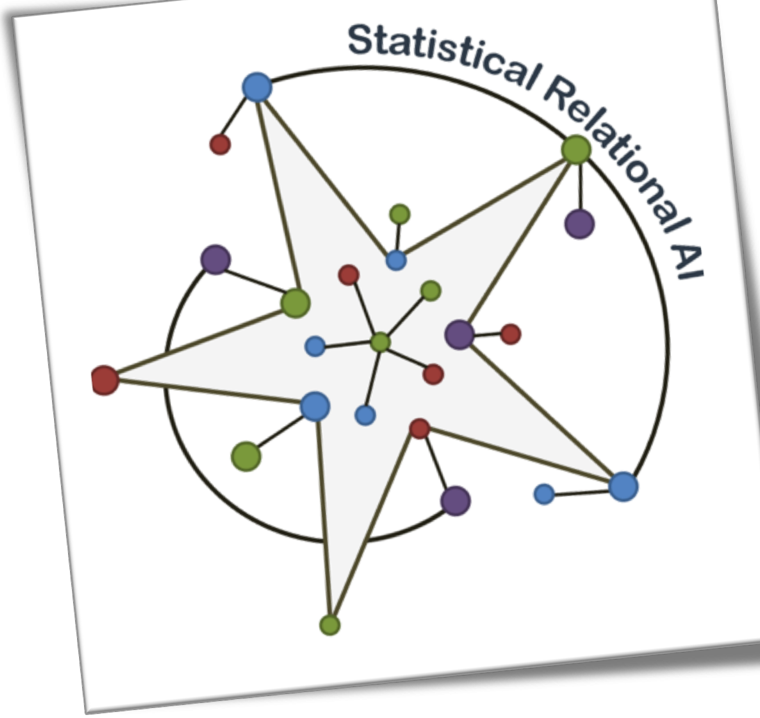
Probabilistic Relational Modeling



Statistical Relational AI

Tutorial at ICCS 2019

Marcel Gehrke, University of Lübeck

UNIVERSITÄT ZU LÜBECK

Thanks to Ralf Möller for making his slides publicly available.

Agenda: Probabilistic Relational Modeling

- Application
  - Information retrieval (IR)
  - Probabilistic Datalog
- Probabilistic relational logics
  - Overview
  - Semantics
  - Inference problems
- Scalability issues
  - Proposed solutions

Goal:
Overview of
central ideas

*We would like to thank all our colleagues for making their slides available (see some of the references to papers for respective credits). Slides are almost always modified.

Application

- Probabilistic Datalog for information retrieval[Fuhr 95]:

```
0.7 term(d1,ir).
0.8 term(d1,db).
0.5 link(d2,d1).
about(D,T):- term(D,T).
about(D,T):- link(D,D1), about(D1,T).
```

- Query/Answer

```
:- term(X,ir) & term(X,db).

X = 0.56 d1
```

N. Fuhr, Probabilistic Datalog - A Logic For Powerful Retrieval Methods. Prod. SIGIR, pp. 282-290, 1995

Application: Probabilistic IR

- Probabilistic Datalog

```
0.7 term(d1,ir).
0.8 term(d1,db).
0.5 link(d2,d1).
about(D,T):- term(D,T).
about(D,T):- link(D,D1), about(D1,T).
```

- Query/Answer

```
q(X):- term(X,ir).
q(X):- term(X,db).
:-q(X)
X = 0.94 d1
```

Application: Probabilistic IR

- Probabilistic Datalog

```
0.7 term(d1,ir).
0.8 term(d1,db).
0.5 link(d2,d1).
about(D,T):- term(D,T).
about(D,T):- link(D,D1), about(D1,T).
```

- Query/Answer

```
:- about(X,db).

X = 0.8 d1;
X = 0.4 d2
```

Application: Probabilistic IR

- Probabilistic Datalog

```
0.7 term(d1,ir).
0.8 term(d1,db).
0.5 link(d2,d1).
about(D,T):- term(D,T).
about(D,T):- link(D,D1), about(D1,T).
```

- Query/Answer

```
:- about(X,db) & about(X,ir).

X = 0.56 d1;
X = 0.28 d2 # NOT naively 0.14 = 0.8*0.5*0.7*0.5
```

Solving Inference Problems

- QA requires proper probabilistic reasoning
- Scalability issues
  - Grounding and propositional reasoning?
  - In this tutorial the focus is on lifted reasoning in the sense of [Poole 2003]
    - Lifted exact reasoning
    - Lifted approximations
- Need an overview of the field: Consider related approaches first

D. Poole. "First-order Probabilistic Inference." In: IJCAI-03 Proceedings of the 18th International Joint Conference on Artificial Intelligence. 2003

Application: Probabilistic IR

- Uncertain Datalog rules: Semantics?

```
0.7 term(d1,ir).
0.8 term(d1,db).
0.5 link(d2,d1).
0.9 about(D,T):- term(D,T).
0.7 about(D,T):- link(D,D1), about(D1,T).
```

Application: Probabilistic IR

- Uncertain Datalog rules: Semantics?

```
0.7 term(d1,ir).
0.8 term(d1,db).
0.5 link(d2,d1).
0.9 temp1.
0.7 temp2.
    about(D,T):- term(D,T), temp1.
    about(D,T):- link(D,D1), about(D1,T), temp2.
```

Probabilistic Datalog: QA

- Derivation of lineage formula
  with Boolean variables corresponding to used facts

  T. Rölleke; N. Fuhr, Information Retrieval with Probabilistic Datalog. In: Logic and Uncertainty in Information Retrieval: Advanced models for the representation and retrieval of information, 1998.

- Probabilistic relational algebra

  N. Fuhr; T. Rölleke, A Probabilistic Relational Algebra for the Integration of Information Retrieval and Database Systems. ACM Transactions on Information Systems 14(1), 1997.

- Ranking / top-k QA

  N. Fuhr. 2008. A probability ranking principle for interactive information retrieval. Inf. Retr. 11, 3, 251-265, 2008.

Probabilistic Relational Logics: Semantics

- Distribution semantics (aka grounding or Herbrand semantics) [Sato 95]
  Completely define discrete joint distribution by "factorization"
  Logical atoms treated as random variables
  - Probabilistic extensions to Datalog [Schmidt et al. 90, Dantsin 91, Ng & Subramanian 93, Poole et al. 93, Fuhr 95, Rölleke & Fuhr 97 and later]
  - Primula [Jaeger 95 and later]
  - BLP, ProbLog [De Raedt, Kersting et al. 07 and later]
  - Probabilistic Relational Models (PRMs) [Poole 03 and later]
  - Markov Logic Networks (MLNs) [Domingos et al. 06]
- Probabilistic Soft Logic (PSL) [Kimmig, Bach, Getoor et al. 12]
  Define density function using log-linear model
- Maximum entropy semantics [Kern-Isberner, Beierle, Finthammer, Thimm 10, 12]
  Partial specification of discrete joint with "uniform completion"

Sato, T., A statistical learning method for logic programs with distribution semantics, In: International Conference on Logic Programming, pages 715–729, 1995.

Inference Problems w/ and w/o Evidence

- Static case
  - Projection (margins),
  - Most-probable explanation (MPE)
  - Maximum a posteriori (MAP)
  - Query answering (QA): compute bindings
- Dynamic case
  - Filtering (current state)
  - Prediction (future states)
  - Hindsight (previous states)
  - MPE, MAP (temporal sequence)

ProbLog

```
% Intensional probabilistic facts:
0.6::heads(C) :- coin(C).

% Background information:
coin(c1).
coin(c2).
coin(c3).
coin(c4).

% Rules:
someHeads :- heads(_).

% Queries:
query(someHeads).
0.9744
```


https://dtai.cs.kuleuven.be/problog/

ProbLog

- Compute marginal probabilities of any number of ground atoms in the presence of evidence
- Learn the parameters of a ProbLog program from partial interpretations
- Sample from a ProbLog program
  - Generate random structures (use case: [Goodman & Tenenbaum 16])
- Solve decision theoretic problems:
  - Decision facts and utility statements

ProbLog: A probabilistic Prolog and its application in link discovery, L. De Raedt, A. Kimmig, and H. Toivonen, Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI-07), Hyderabad, India, pages 2462-2467, 2007

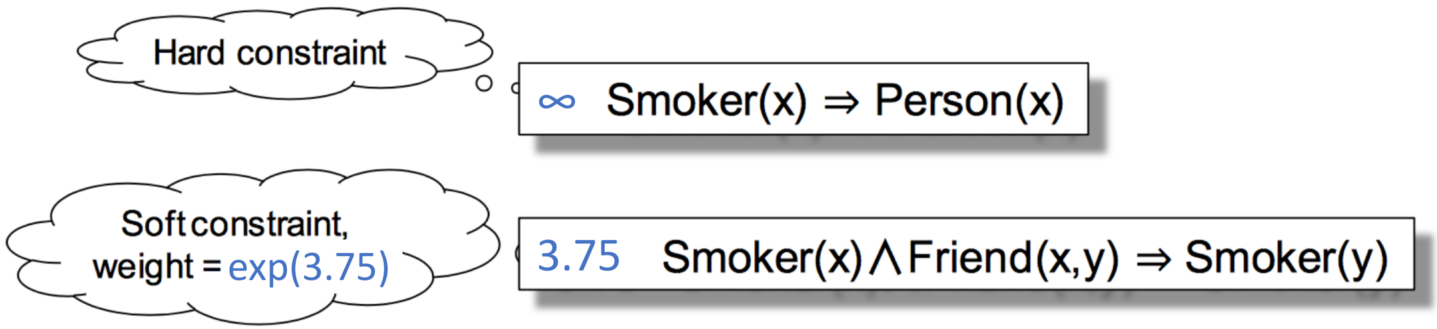
K. Kersting and L. De Raedt, Bayesian logic programming: Theory and Tool. In L. Getoor and B. Taskar, editors, An Introduction to Statistical Relational Learning. MIT Press, 2007

Daan Fierens, Guy Van den Broeck, Joris Renkens, Dimitar Shterionov, Bernd Gutmann, Ingo Thon, Gerda Janssens, and Luc De Raedt. Inference and learning in probabilistic logic programs using weighted Boolean formulas, In: Theory and Practice of Logic Programming, 2015

N. D. Goodman, J. B. Tenenbaum, and The ProbMods Contributors. Probabilistic Models of Cognition (2nd ed.), 2016. Retrieved 2018-9-23 from https://probmods.org/

Markov Logic Networks (MLNs)

- Weighted formulas for modelling constraints [Richardson & Domingos 06]



- An MLN is a set of constraints $(w, \Gamma(\boldsymbol{x}))$
  - w = weight
  - $\Gamma(\boldsymbol{x})$ = FO formula
- *weight* of a world = product of $\exp(w)$
  - for all MLN rules $(w, \Gamma(\boldsymbol{x}))$ and groundings $\Gamma(\boldsymbol{a})$ that hold in that world
- Probability of a world $= \frac{weight}{Z}$
  - Z = sum of weights of all worlds (no longer a simple expression!)

Richardson, M., & Domingos, P. Markov logic networks. Machine learning, 62(1-2), 107-136. 2006

Why exp?

- Log-linear models
- Let D be a set of constants and $\omega \in \{0,1\}^m$ a world with m atoms w.r.t. D

$$weight(\omega) = \prod_{\{(w,\Gamma(\boldsymbol{x}))\in MLN \mid \exists \boldsymbol{a}\in D^n : \omega\models\Gamma(\mathbf{a})\}} \exp(w)$$

$$\ln\big(weight(\omega)\big) = \sum_{\{(w,\Gamma(\boldsymbol{x}))\in MLN \mid \exists \boldsymbol{a}\in D^n : \omega \models \Gamma(\mathbf{a})\}} w$$

  - Sum allows for component-wise optimization during weight learning

- $Z = \sum_{\omega\in\{0,1\}^m} \ln\big(weight(\omega)\big)$
- $P(\omega) = \frac{\ln\big(weight(\omega)\big)}{Z}$

Factor graphs

- Unifying representation for specifying discrete distributions with a factored representation
  - Potentials (weights) rather than probabilities
- Also used in engineering community for defining densities w.r.t. continuous domains [Loeliger et al. 07]

H.-A. Loeliger, J. Dauwels, J. Hu, S. Korl, L. Ping, and F. R. Kschischang. "The Factor Graph Approach to Model-Based Signal Processing." In: Proc. IEEE 95.6, pp. 1295–1322, 2007.

Scalability Issues

Scalability: Proposed solutions

- Limited expressivity
  - Probabilistic databases
- Knowledge Compilation
  - Linear programming
  - Weighted first-order model counting
- Approximation
  - Grounding + belief propagation (TensorLog)

Wrap-up Statistical Relational AI

- Probabilistic relational logics
  - Overview
  - Semantics
  - Inference problems
- Dealing with scalability issues (avoiding grounding)
  - Reduce expressivity (liftable queries)
  - Knowledge compilation (WFOMC)
  - Approximation (BP)

Next: Exact Lifted Inference

Mission and Schedule of the Tutorial*

Providing an introduction into inference in StaRAI

- Introduction — 20 min ✓
  - StaR AI
- Overview: Probabilistic relational modeling — 30 min ✓
  - Semantics (grounded-distributional, maximum entropy)
  - Inference problems and their applications
  - Algorithms and systems
- Scalable static inference — 40 + 30 min
  - Exact propositional inference
  - Exact lifted inference
- Scalable dynamic inference — 50 min
  - Exact propositional inference
  - Exact lifted inference
- Summary — 10 min

*Thank you to the SRL/StaRAI crowd for all their exciting contributions! The tutorial is necessarily incomplete. Apologies to anyone whose work is not cited