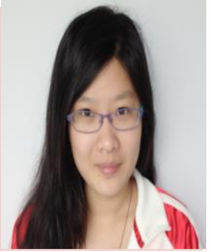
SFU

Fast Learning of Relational Dependency Networks

Oliver Schulte	Zhensong Qian	Arthur Kirkpatrick	Xiaoqian Yin	Yan Sun

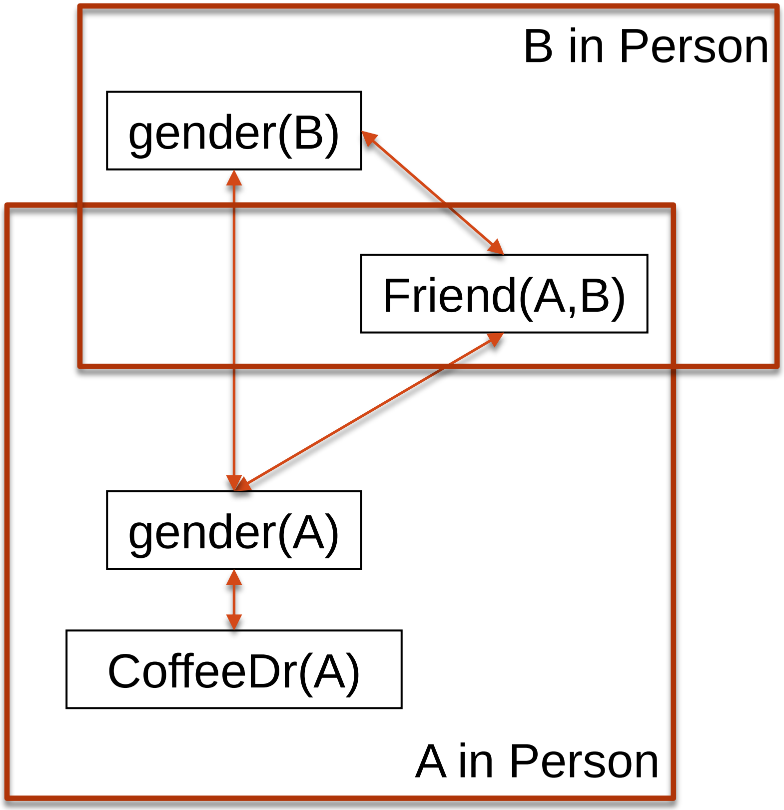






Relational Dependency Networks

- Structure: Directed graph, **cycles are allowed.**
- Parents of Node = Markov Blanket of Node.
- Parameter = distribution of child given parents.
- Accommodates relational autocorrelations.



Neville, J. & Jensen, D. (2007), 'Relational Dependency Networks', *Journal of Machine Learning Research 8, 653--692.*

Overview

Task: learn relational dependency network

structure + parameters

our new approach

single generative model

<u>fast</u> learning Bayesian network

e.g., 1 min for 1M records.

new closed-form transformation method

Convert Bayesian network to Relational Dependency Network

previous approaches

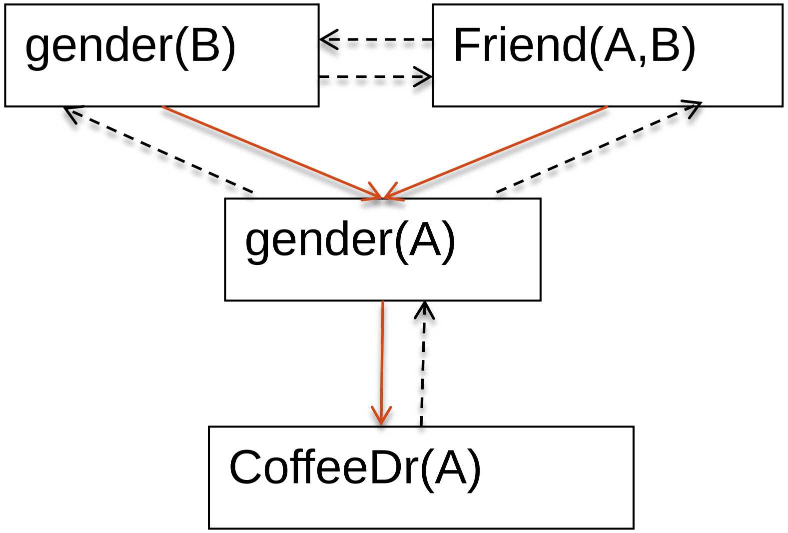
multiple discriminative models

independently learned

(one for each predicate)

From BN Structure To DN Structure

- Solid arrows = Bayesian Network
- Solid + dash arrows = Dependency Network



Heckerman, D.; Chickering, D. M.; Meek, C.; Rounthwaite, R.; Kadie, C. & Kaelbling, P. (2000), 'Dependency Networks for Inference, Collaborative Filtering, and Data Visualization', *Journal of Machine Learning Research* *1, 49—75.*

From BN Parameters to DN Parameters

- *Log-linear* model for probability of target instance given its Markov blanket.
- Example: Predict the gender of Sam, given that
  - 40% of Sam's friends are Women, and
  - Sam is a coffee drinker.

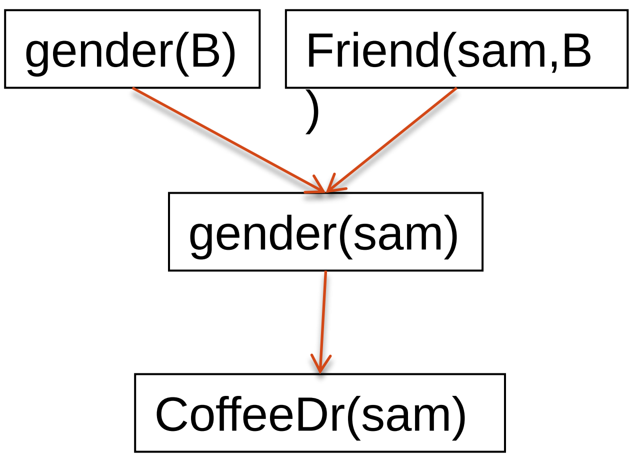
DN Parameter

$$P(\text{target} = \text{value} \mid \text{Markov blanket}) \propto \exp \{\textstyle\sum_{\text{target instance + children}} \sum_{\text{parent values PV, child values CV}} \ln(P(CV \mid PV)) \cdot \text{frequency}(CV,PV)\}$$

BN Parameter   Markov Blanket

Example

- Predict the gender of Sam, given that
  - 40% of Sam's friends are Women, and
  - Sam is a coffee drinker:



Child Value	Parent State	CP	log(CP)	Rel. Freq.	log(CP) * Freq.
g(sam) = W	g(B) = W, F(sam,B) = T	0.55	-0.60	0.40	-0.24
g(sam) = W	g(B) = M, F(sam,B) = T	0.37	-0.99	0.60	-0.60
cd(sam) = T	g(sam) = W	0.80	-0.22	1.00	-0.22
cd(sam) = F	g(sam) = W	0.20	-1.61	0.00	0.00
Sum { EXP(Sum) $\propto$ P(gender(sam)=W | MB) }					-1.06

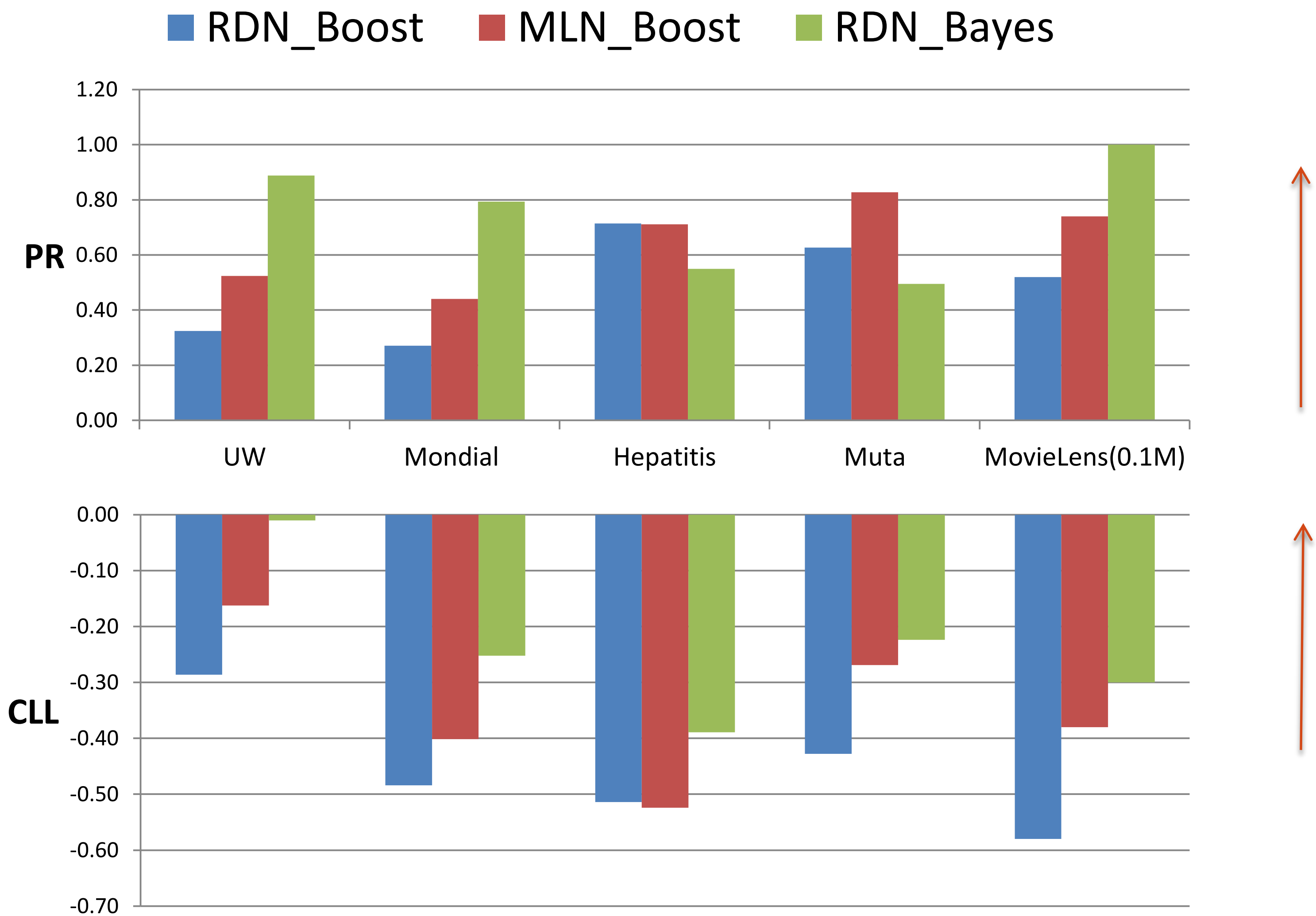
P(g(A) = W | g(B) = W, F(A,B) = T) =0.55
P(g(A) = M | g(B) = M, F(A,B) = T) = 0.63
P(cd(A) = T|g(A) = M) = 0.6
P(cd(A) = T|g(A) = W) = 0.8

Evaluation Metrics

- Running time
- Conditional Log Likelihood (CLL)
  - How confident we are with the prediction
- Area Under Precision-Recall Curve (PR)
  - For skewed distributions.
- Results are averaged over 5-fold cross-validation, over all two-class predicates in the dataset.
- Comparison Methods: RDN-Boost, MLN-Boost.

Natarajan, S.; Khot, T.; Kersting, K.; Gutmann, B. & Shavlik, J. W. (2012), 'Gradient-based boosting for statistical relational learning: The relational dependency network case', *Machine Learning* ***86(1), 25-56.***

Accuracy Comparison



Learning Time Comparison

Dataset	# Predicates	# tuples	RDN_Boost	MLN_Boost	RDN_Bayes
UW	14	612	15±0.3	19±0.7	**1±0.0**
Mondial	18	870	27±0.9	42±1.0	102±6.9
Hepatitis	19	11,316	251±5.3	230±2.0	286±2.9
Mutagenesis	11	24,326	118±6.3	49±1.3	**1±0.0**
MovieLens(0.1M)	7	83,402	44±4.5 min	31±1.87 min	**1±0.0**
MovieLens(1M)	7	1,010,051	>24 hours	>24 hours	**10±0.1**

- Standard deviations are shown.
- Units are *seconds* unless otherwise stated.

RDN-Bayes uses more relevant predicates and more first-order variables

- Our best predicate for each database:

Database	Target Predicate	# extra predicates	# extra first order variables	CLL-diff
Mondial	religion	11	1	0.58
IMDB	gender	6	2	0.30
UW-CSE	student	4	1	0.50
Hepatitis	sex	4	2	0.20
Mutagenesis	ind1	5	1	0.56
MovieLens	gender	1	1	0.26

Structure Comparison Example IMDB

UserID	Occupation	Age	gender

UserID	MovieID	Rating

RDN-Boost

Model	Target	Markov Blanket
RDN-Boost	gender(U)	Occupation(U), Age(U)
RDN-Bayes	gender(U)	Occupation(U), Age(U), **Rating(U,M), RunningTime(M), CastMember(M,X), AGender(X)**

MovieID	Time

RDN-Bayes

ActorID	MovieID

ActorID	AGender

Conclusions

- Basic Idea: convert Bayesian networks to relational dependency networks.
  - fast BN learning ⇒ fast DN learning.
  - dependency networks ⇒ inference with cyclic dependencies/autocorrelations.
- New log-linear model for converting BN parameters to DN parameters.
  - I.e., define probability of a node given Markov blanket, Bayes net model.
- Empirical evaluation
  - Scales very well with number of records.
  - Competitive accuracy with functional gradient boosting.

There's More

- Empirical Comparisons
  - counts instead of frequencies
  - weight learning
  - more on MLN-Boost
- Theorems about dependency network consistency

The End

- Any questions?