Human Allied Statistical Relational Artificial Intelligence

Sriraam Natarajan

University of Texas-Dallas

Who we are!

Research Faculty

Dr. Gautam Kunapuli

Current Students (PhD)

Mayukh Das, Srijita Das, Devendra Dhami, Alex Hayes, Navdeep Kaur, Nandini Ramanan, Kaushik Roy, Harsha Kokel, Changbing Li, Yuqiao Chen…

Alumni (PhD)

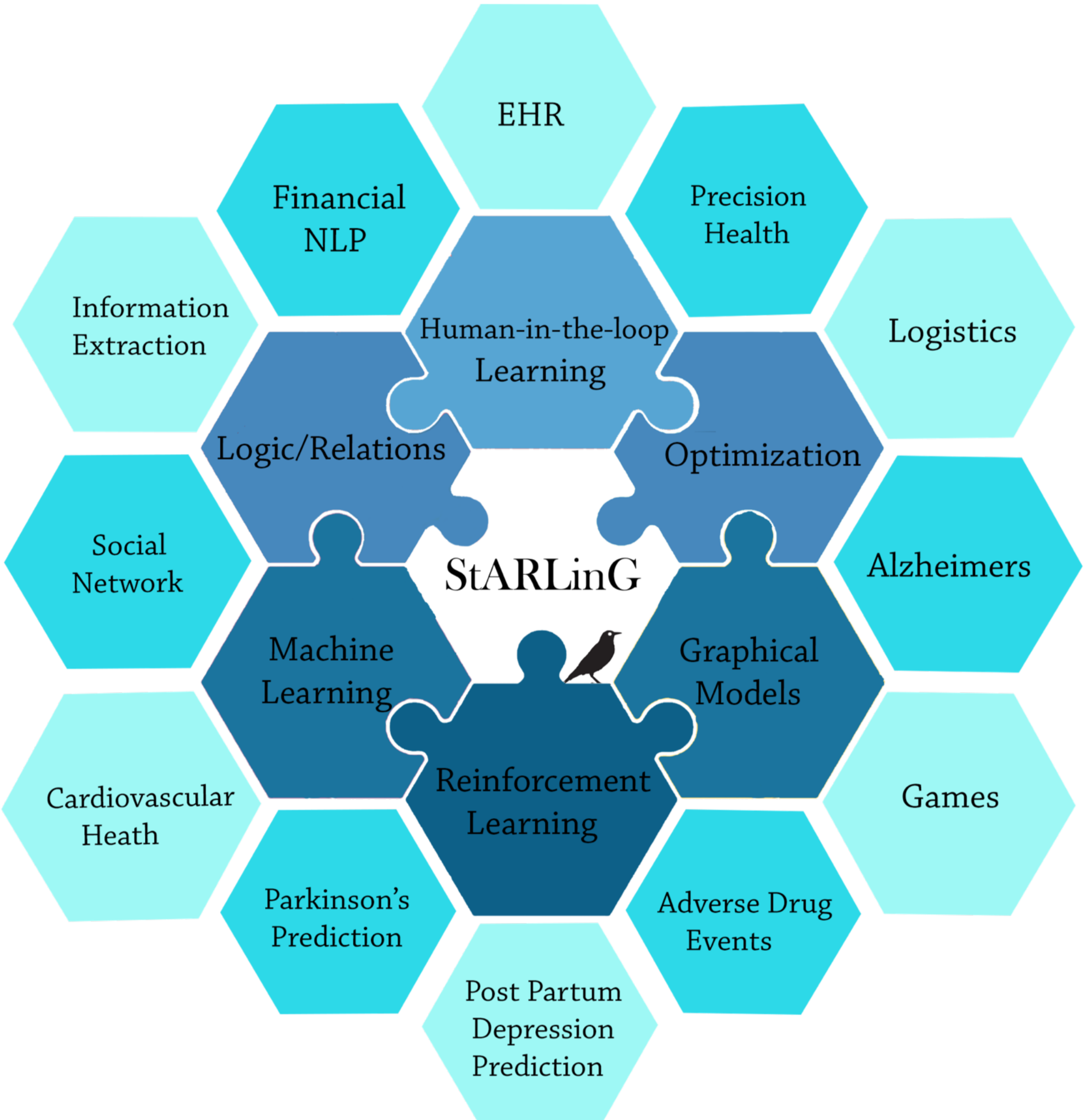
Phillip Odom , Shuo Yang, Tushar Khot

Key Collaborators

Kristian Kersting, Jude Shavlik, David Page, Dan Roth, Jana Doppa, Ron Parr, William Cohen, Kay Connelly, Clinical collaborators ….

Funding agencies

DARPA (CwC, DEFT & Machine Reading), NSF (SCH), AFRL, ARO (YIP, STIR), AFOSR (SBIR), NIH (R01), Indiana (Precision Medicine), XEROX PARC, Amazon, Intel and TURVO Inc.


EHR
Financial NLP
Precision Health
Information Extraction
Human-in-the-loop Learning
Logistics
Logic/Relations
Optimization
Social Network
StARLinG
Alzheimers
Machine Learning
Graphical Models
Cardiovascular Heath
Reinforcement Learning
Games
Parkinson's Prediction
Adverse Drug Events
Post Partum Depression Prediction

What we do!

Human Allied AI

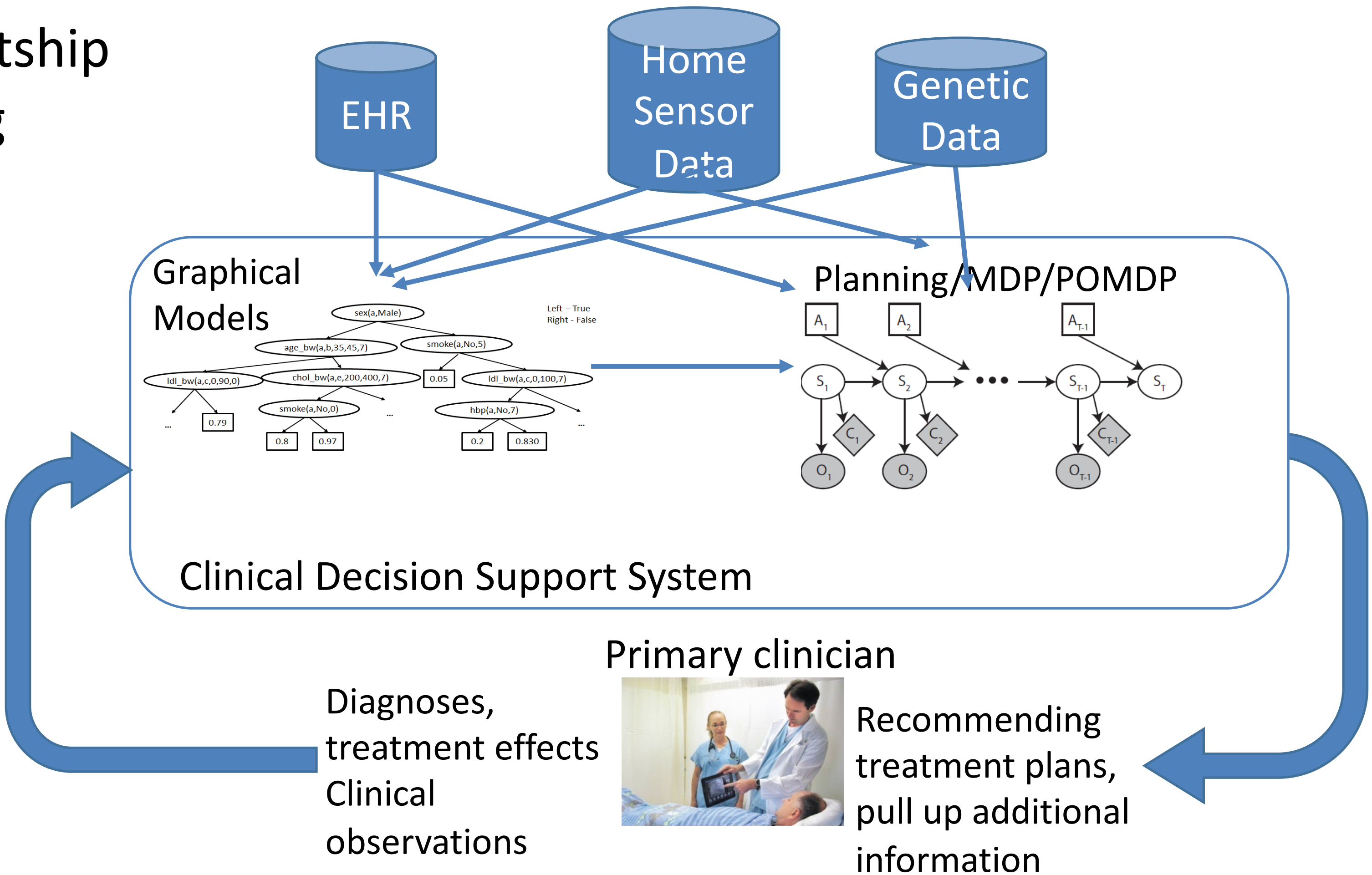
Can we build systems that can seamlessly interact with, learn from and collaborate with human expert?

NSF SCH 13
PHI16

Human Allied AI

Fern et al IJCAI 07
Natarajan et al ILP08, ILP09
Natarajan et al KAIS 11
Fern et al JAIR 14

Assistantship Learning



ARO YIP 13
DARPA DEFT
AFOSR STTR 14
NIH R01 15
PARC Faculty Award
ARO STIR 17

Human Allied AI

Kunapuli et al ICDM 13
Odom et al AAAI 15, AAMAS 16, ECML 16, ILP 16, AIME 15
Yang ICDM 14, ECML 13
Macleod CHASE 16
…

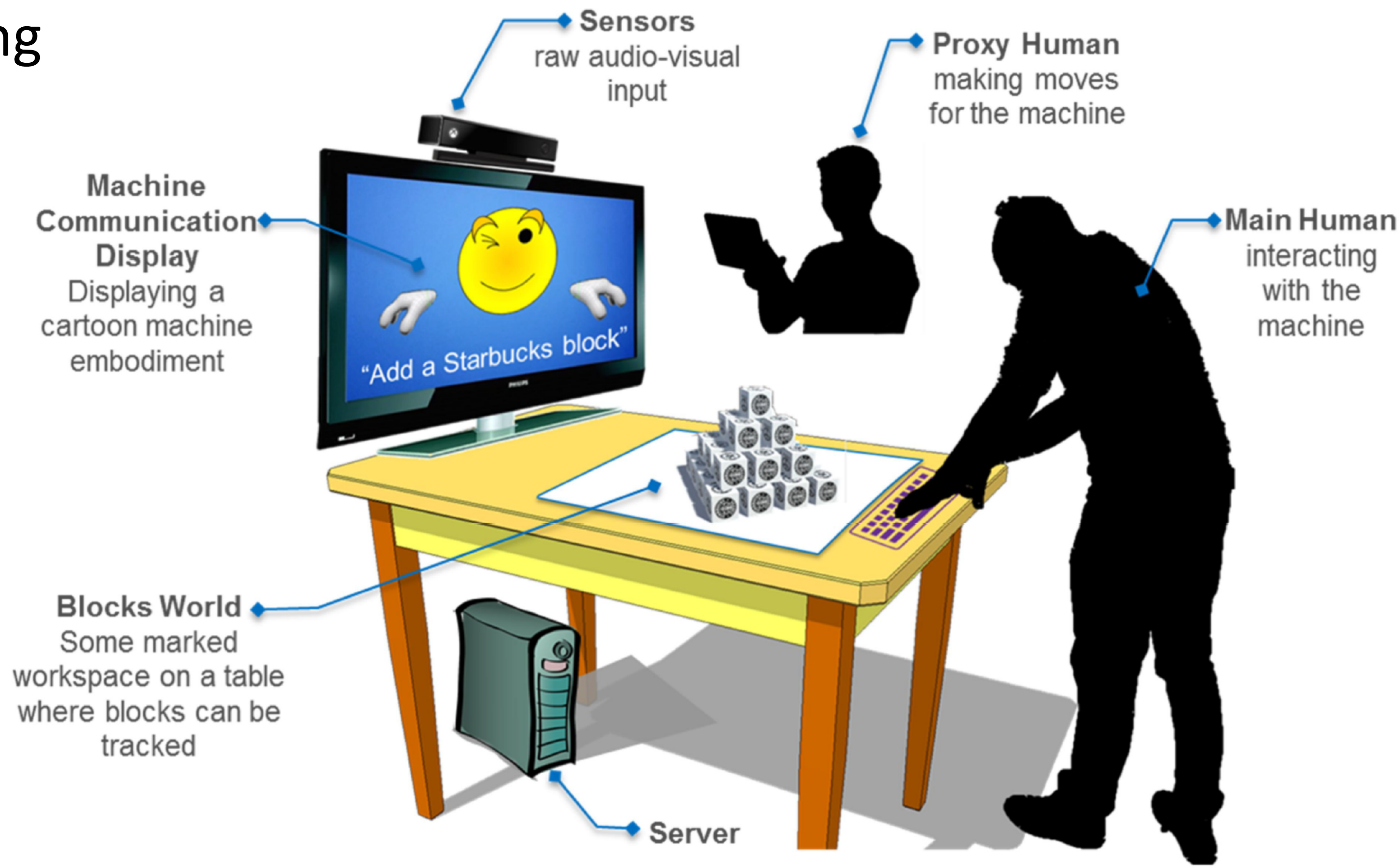
Apprenticeship Learning

DARPA CwC

Human Allied AI

Das SDM 16, HMCL WS 17, AAMAS 18
KBS (under review)

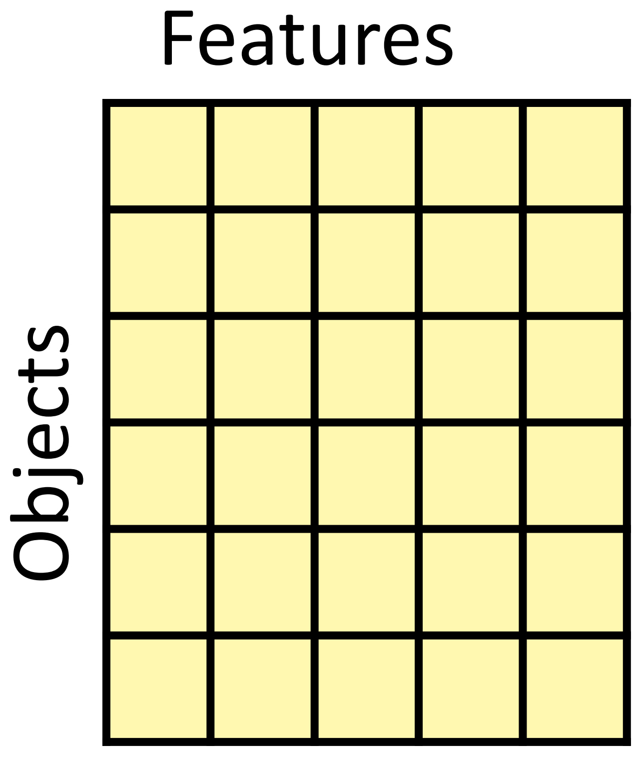
Collaborative Problem solving



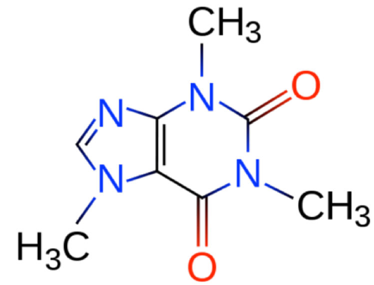
Our Prior Work towards HAAI

1. Tractable modeling of Multi-modal data
2. Efficient Reasoning in large domains
3. Faithful Temporal Modeling
4. Effective Sequential decision-making
5. Scalable learning with complex relational data
6. Explicit modeling of noise and uncertainty

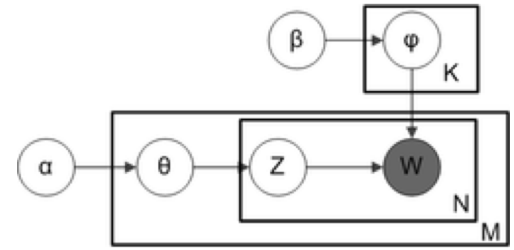
Bottom line:
Take your data spreadsheet …



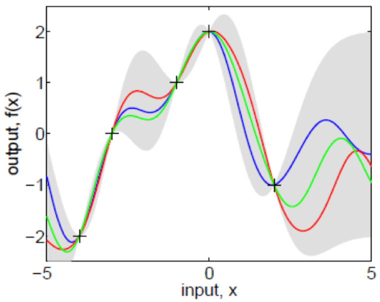
… and apply data mining



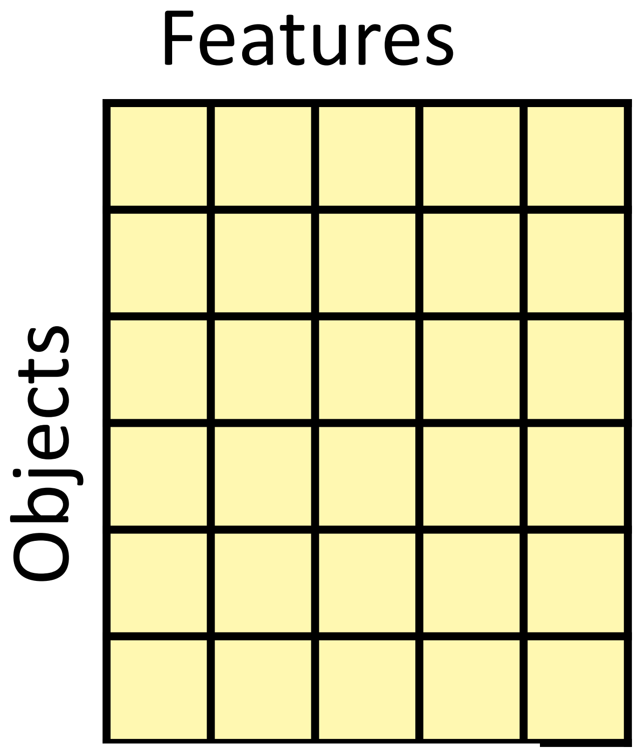
Graph Mining



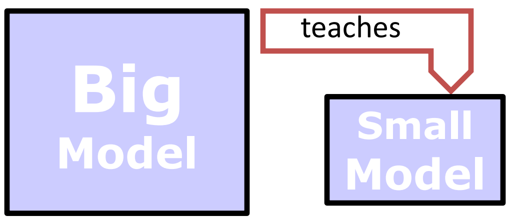
Latent Dirichlet Allocation



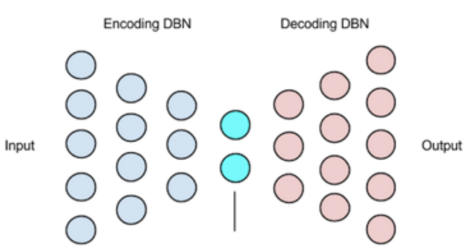
Gaussian Processes



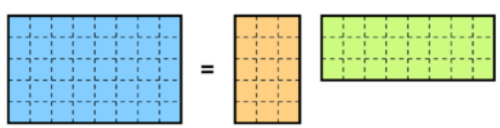
Boosting



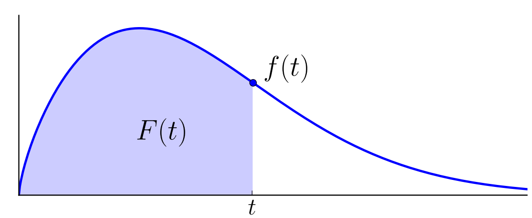
Distillation/LUPI



Autoencoder,
Deep Learning



Big Data Matrix Factorization

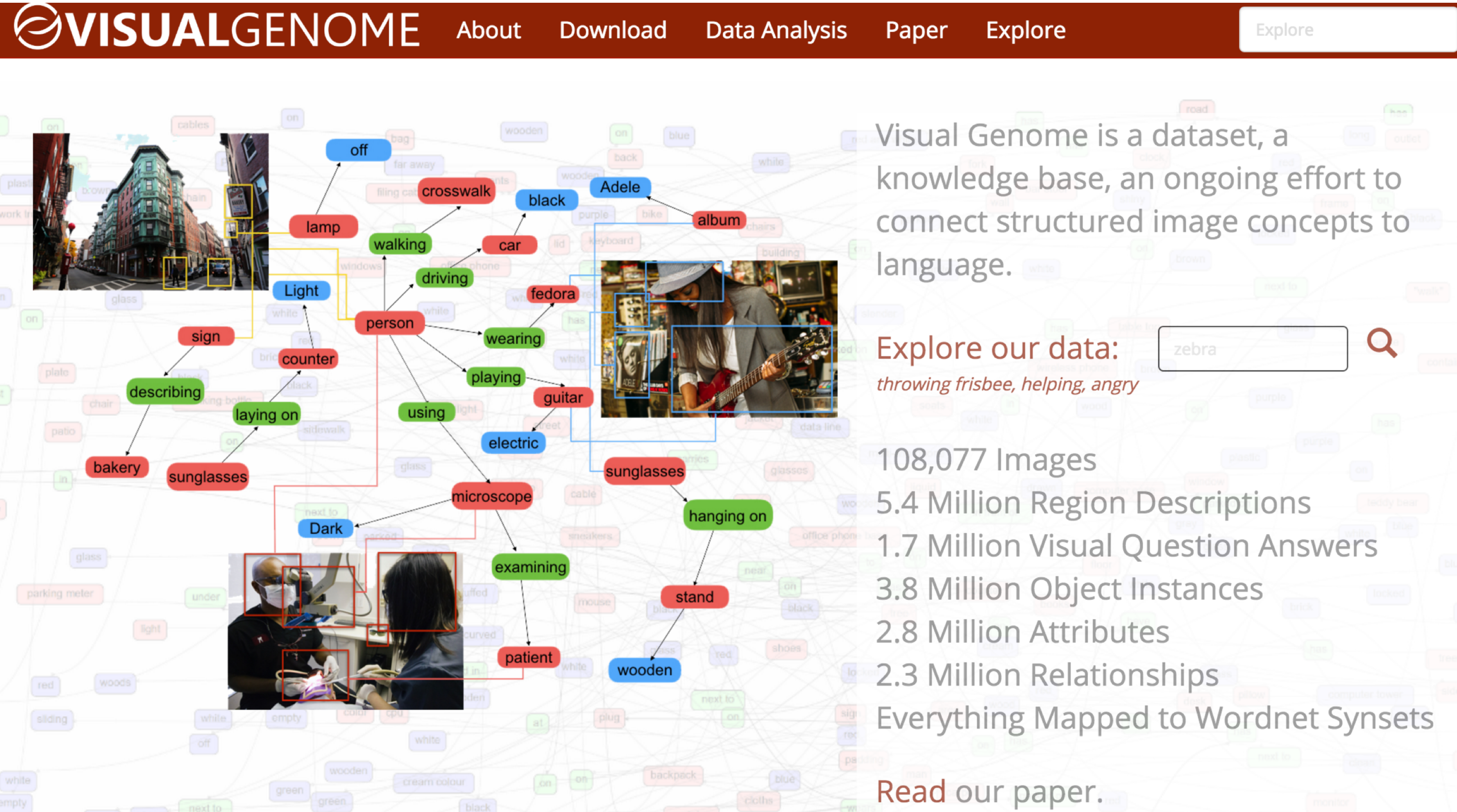


Diffusion Models

and many more …

Heterogenous data networks abound

[Lu, Krishna, Bernstein, Fei-Fei „Visual Relationship Detection“ CVPR 2016]



Most data in the world is stored in relational databases

Mining Electronic Health Records is an opportunity to save our lifes and a lot of money.

Unfortunately, they are noisy and interconnected

Patient Table

PatientID	Gender	Birthdate
P1	M	3/22/63

Visit Table

PatientID	Date	Physician	Symptoms	Diagnosis
P1 P1	1/1/01 2/1/03	Smith Jones	palpitations fever, aches	hypoglycemic influenza

Lab Tests

PatientID	Date	Lab Test	Result
P1	1/1/01	blood glucose	42
P1	1/9/01	blood glucose	??

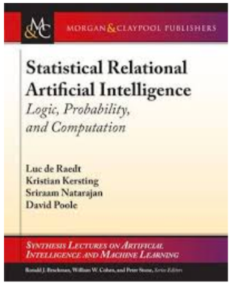
SNP Table

PatientID	SNP1	SNP2	…	SNP500K
P1 P2	AA AB	AB BB		BB AA

Prescriptions

PatientID	Date Prescribed	Date Filled	Physician	Medication	Dose	Duration
P1	5/17/98	5/18/98	Jones	prilosec	10mg	3 months

Actually, most data in the world stored in relational databases

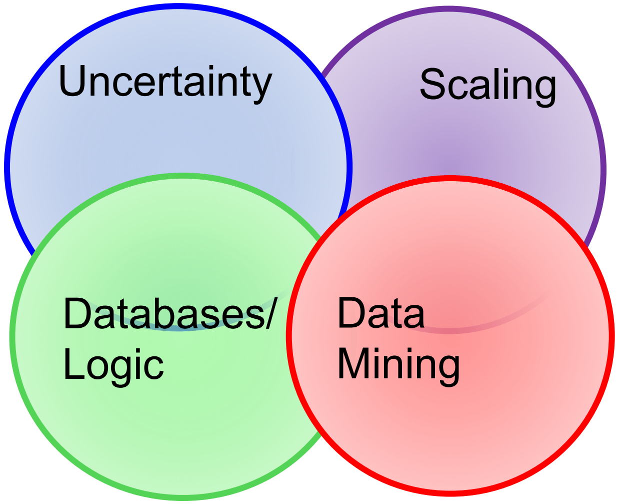


Key Observation

Two trends that drive modern data science

1. Race to deeply understand data
2. Data in a large number of formats

We need a crossover of Machine Learning and Probabilistic/ Statistical Databases



[Getoor, Taskar MIT Press ’07; De Raedt, Frasconi, Kersting, Muggleton, LNCS’08; Domingos, Lowd Morgan&Claypool ’09; Natarajan, Kersting, Khot, Shavlik Springer Brief’15; Russell CACM 58(7): 88-97 ’15]

AI and ML: State-of-the-Art

- **Learning**

  Decision trees, Optimization, SVMs, …
- **Logic**

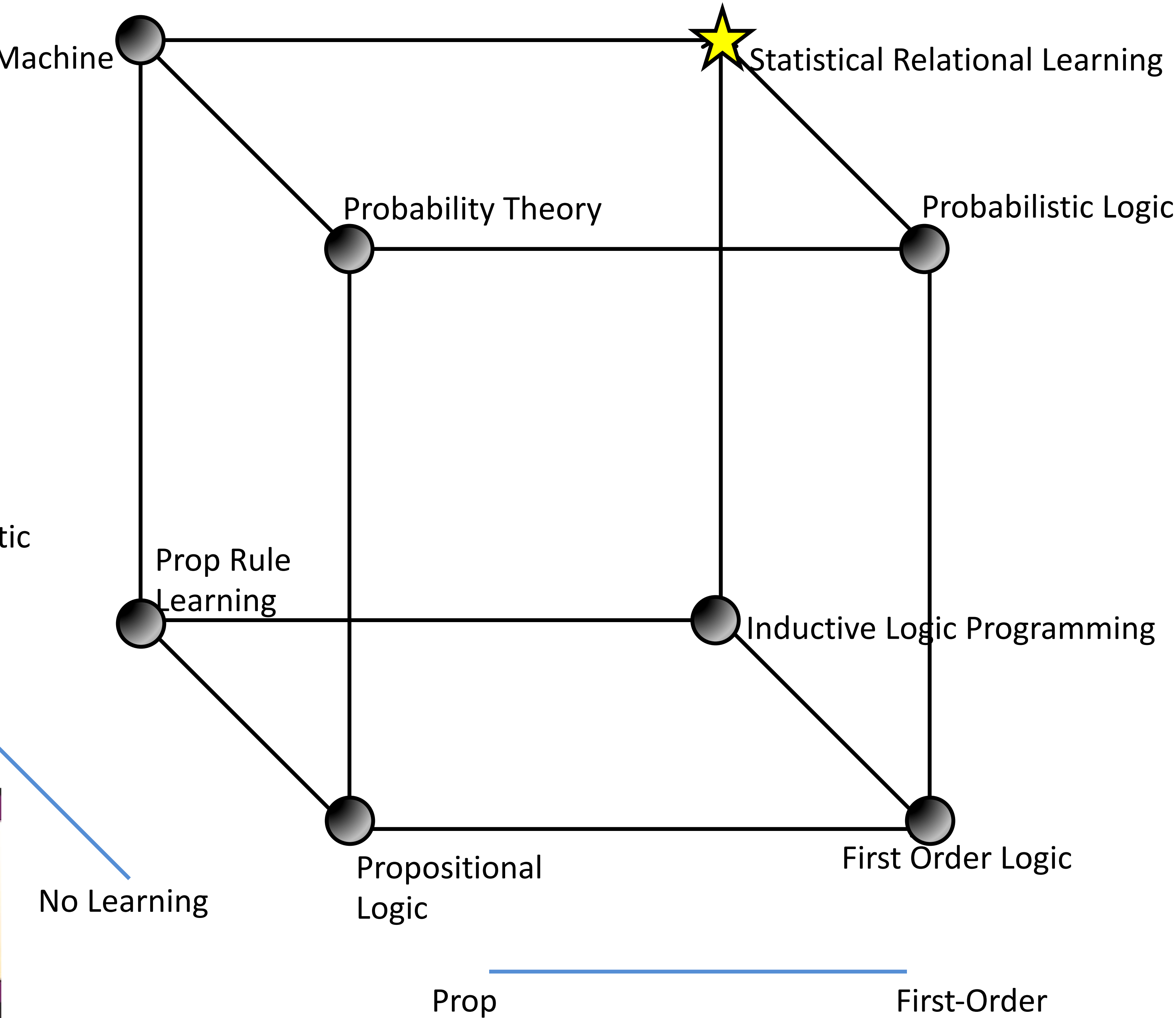
  Resolution, WalkSat, Prolog, description logics, …
- **Probability**

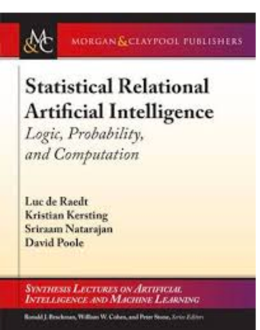
  Bayesian networks, Markov networks, Gaussian Processes…
- **Logic + Learning**

  Inductive Logic Programming (ILP)
- **Learning + Probability**

  EM, Dynamic Programming, Active Learning, …
- **Logic + Probability**

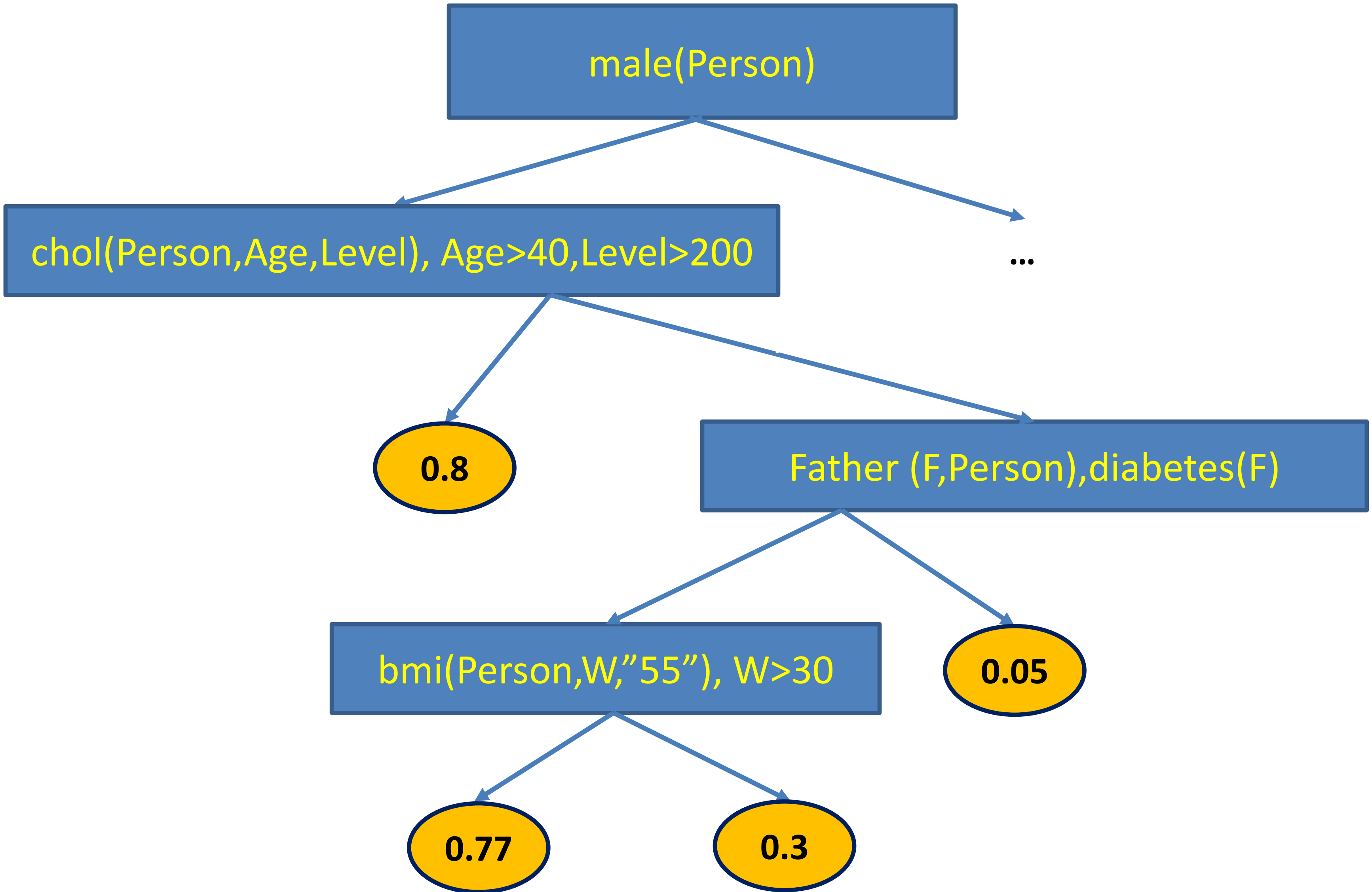
  Nillson, Halpern, Bacchus, KBMC, ICL, …


Classical Machine Learning
Statistical Relational Learning
Stochastic
Probability Theory
Probabilistic Logic
Deterministic
Prop Rule Learning
Inductive Logic Programming
Learning
Propositional Logic
First Order Logic
No Learning
Prop
First-Order


MORGAN & CLAYPOOL PUBLISHERS
Statistical Relational Artificial Intelligence
Logic, Probability, and Computation
Luc de Raedt
Kristian Kersting
Sriraam Natarajan
David Poole
Synthesis Lectures on Artificial Intelligence and Machine Learning

Example - Relational Probability Trees

To predict *heartAttack(Person)*



Outline

- State-of-the-art Structure Learning
- Effective Learning
- Advice Taking
- Actively Seeking Advice

Learning

- Parameter Learning – Where do the numbers come from
- Structure Learning – neither logic program nor models are fixed
- Evidence
  - Parital assignments of values to variables {*burglary = false, earthquake = true, alarm = ?, johncalls = ?, marycalls = true*}

Up next inside Learning

- Parameter learning – graphical models
- Parameter learning – StaR AI models
- Structure learning – logical models & graphical models
- Structure learning – StaRAI models

Maximum Likelihood Estimation

- MLE is the fraction of positive counts over total counts
- Example: Bayes net parameters – estimate each CPD using MLE
- For each P(X=x|Pa(X) = y), count the number of examples of x and y

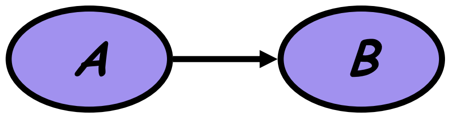
$$P(X = x \mid Pa(X) = y) = \frac{N(X = x, Pa(X) = y)}{N(Pa(X) = y)}$$

EM Idea

- If **data is complete**, ML parameter estimation is easy: **simple counting** (1 iteration)
- But what if there are missing values, i.e., we are facing **incomplete data**?

1. **Complete data** (Imputation)
   - most probable?, average?, ... value
2. **Count**
3. **Iterate**

EM Idea: complete the data



$$\theta_{A=\mathtt{true}} = \frac{1}{2}$$
$$\theta_{B=\mathtt{true}|A=\mathtt{true}} = \frac{1}{2}$$
$$\theta_{B=\mathtt{true}|A=\mathtt{false}} = \frac{1}{2}$$

incomplete data

A	B
true	true
true	**?**
false	true
true	false
false	**?**

complete

$$P(B = \mathtt{true}|A = \mathtt{true}) = 0.5$$
$$P(B = \mathtt{true}|A = \mathtt{false}) = 0.5$$

complete data

expected counts

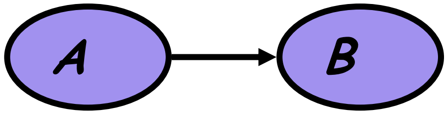
A	B	N
true	true	1.5
true	false	1.5
false	true	1.5
false	false	0.5

maximize

$$\theta_{A=\mathtt{true}} = \frac{1.5+1.5}{1.5+1.5+1.5+0.5} = 0.6$$
$$\theta_{B=\mathtt{true}|A=\mathtt{true}} = \frac{1.5}{1.5+1.5} = 0.5$$
$$\theta_{B=\mathtt{true}|A=\mathtt{false}} = \frac{1.5}{1.5+0.5} = 0.75$$

iterate

EM Idea: complete the data



$$\theta_{A=\texttt{true}} = 0.6$$
$$\theta_{B=\texttt{true}|A=\texttt{true}} = 0.5$$
$$\theta_{B=\texttt{true}|A=\texttt{false}} = 0.875$$

incomplete data

A	B
true	true
true	?
false	true
true	false
false	?

complete

$$P(B=\texttt{true}|A=\texttt{true}) = 0.5$$
$$P(B=\texttt{true}|A=\texttt{false}) = 0.875$$

complete data

expected counts

A	B	N
true	true	1.5
true	false	1.5
false	true	1.875
false	false	0.125

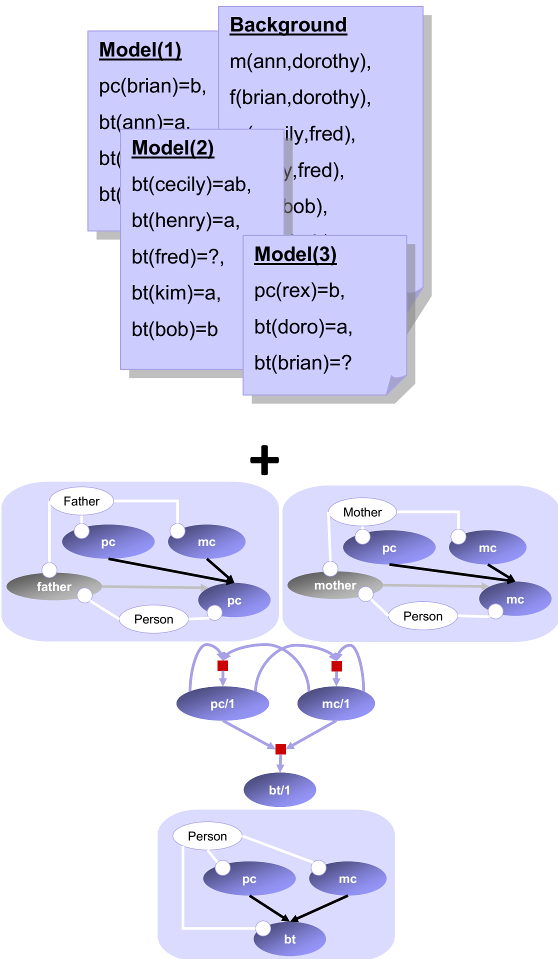
maximize

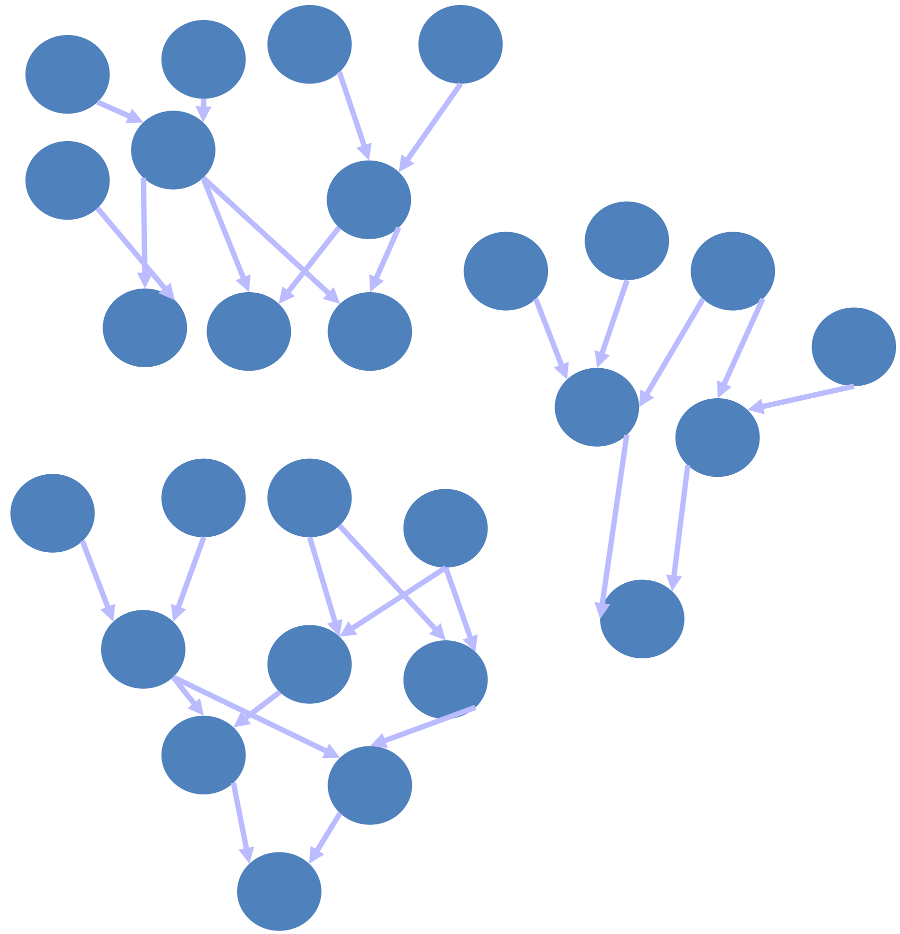
$$\theta_{A=\texttt{true}} = \frac{1.5+1.5}{1.5+1.5+1.875+0.125} = 0.6$$
$$\theta_{B=\texttt{true}|A=\texttt{true}} = \frac{1.5}{1.5+1.5} = 0.5$$
$$\theta_{B=\texttt{true}|A=\texttt{false}} = \frac{1.875}{1.875+0.125} = 0.9375$$

iterate

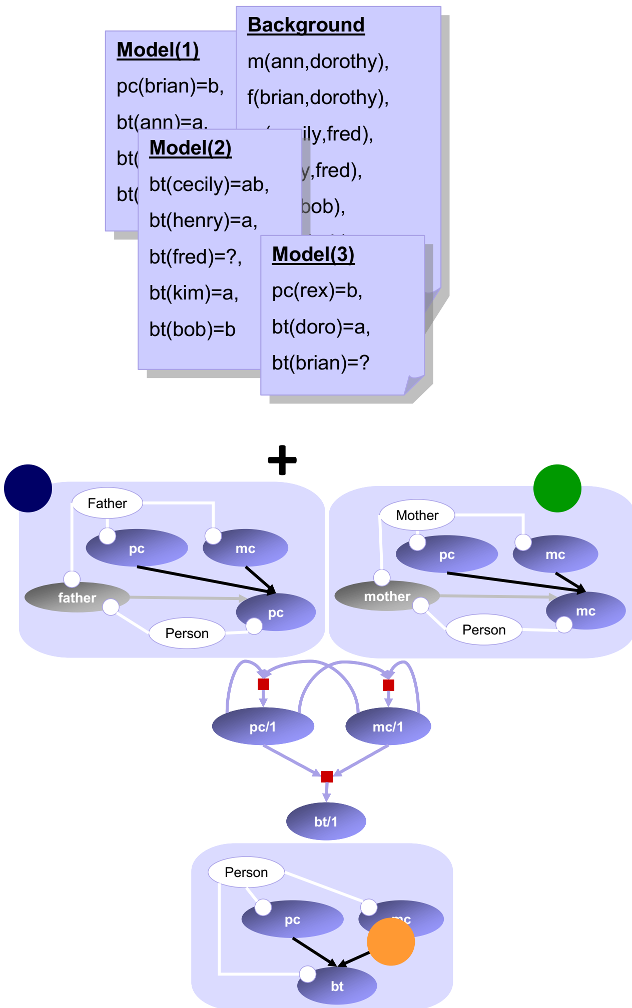
- Parameter learning – graphical models
- Parameter learning – StaR AI models
- Structure learning – logical models & graphical models
- Structure learning – StaRAI models

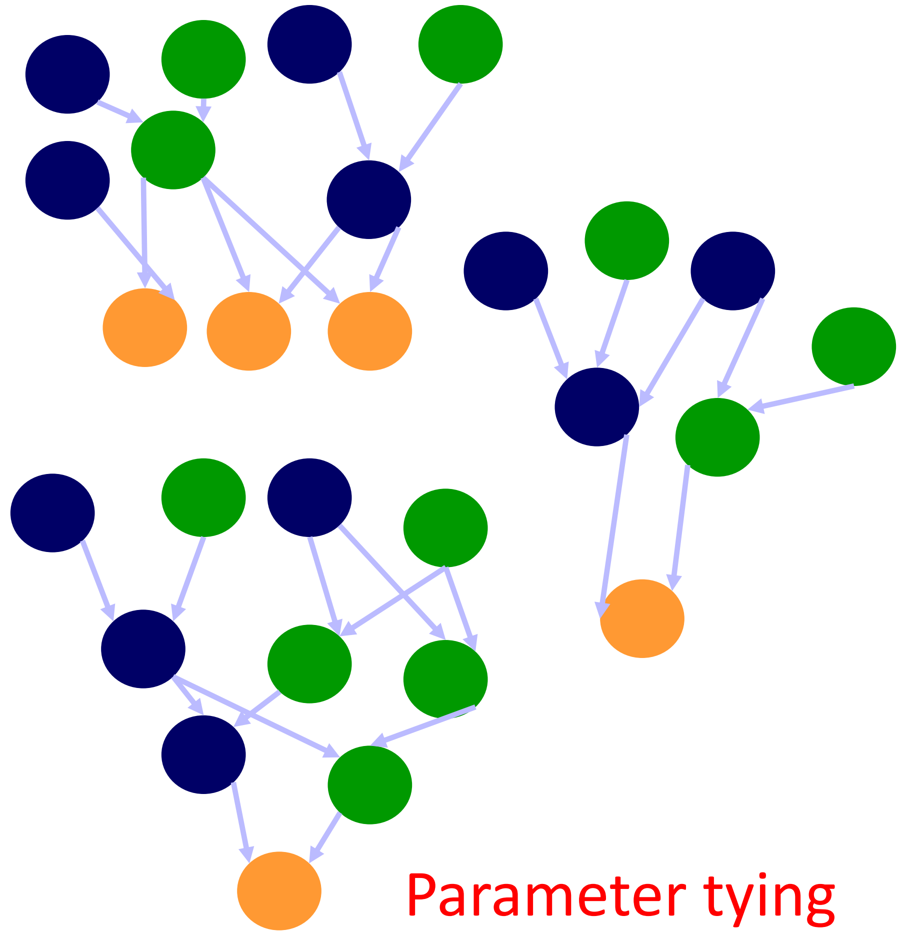
Relational Parameter Estimation





Relational Parameter Estimation





Parameter tying

So, apply „standard“ EM

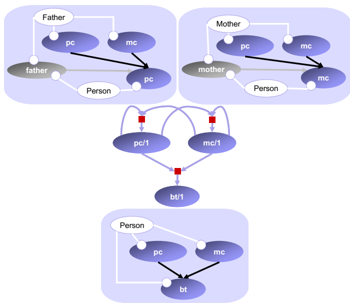
Model(1)
pc(brian)=b,
bt(ann)=a,
bt(
bt(

Background
m(ann,dorothy),
f(brian,dorothy),
ily,fred),
y,fred),
bob),

Model(2)
bt(cecily)=ab,
bt(henry)=a,
bt(fred)=?,
bt(kim)=a,
bt(bob)=b

Model(3)
pc(rex)=b,
bt(doro)=a,
bt(brian)=?

Logic Program L



iterate until convergence

Expectation

Initial Parameters q0

Current Model (M,qk)

Inference

Expected counts of a clause

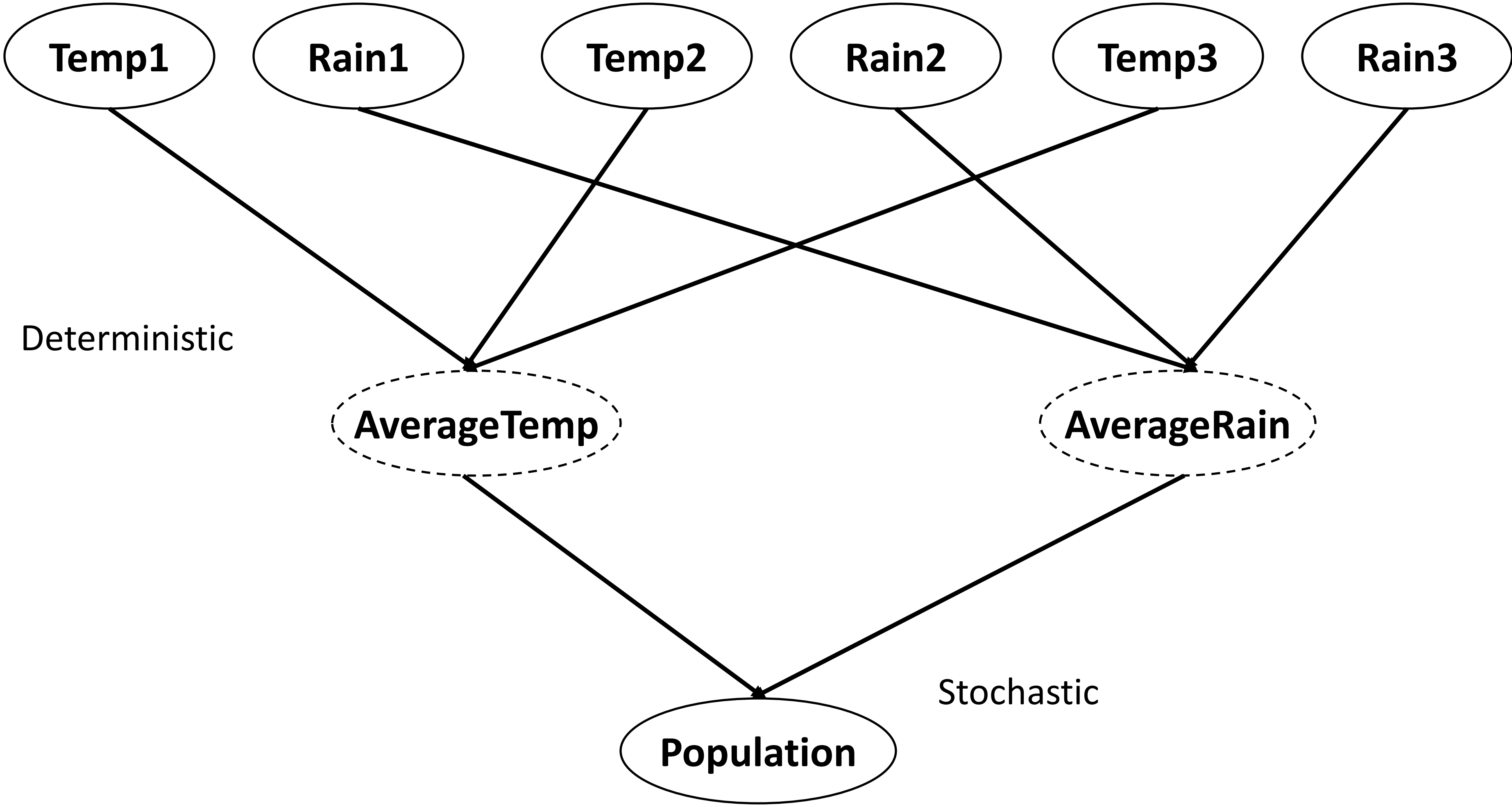
$$\sum_{\text{Ground Instance GI}} \sum_{\text{DataCase DC}} \textbf{P(head(GI), body(GI) | DC)}$$

$$\frac{\sum_{\text{Ground Instance GI}} \sum_{\text{DataCase DC}} \textbf{P(head(GI), body(GI) | DC)}}{\sum_{\text{Ground Instance GI}} \sum_{\text{DataCase DC}} \textbf{P(body(GI) | DC)}}$$

Maximization

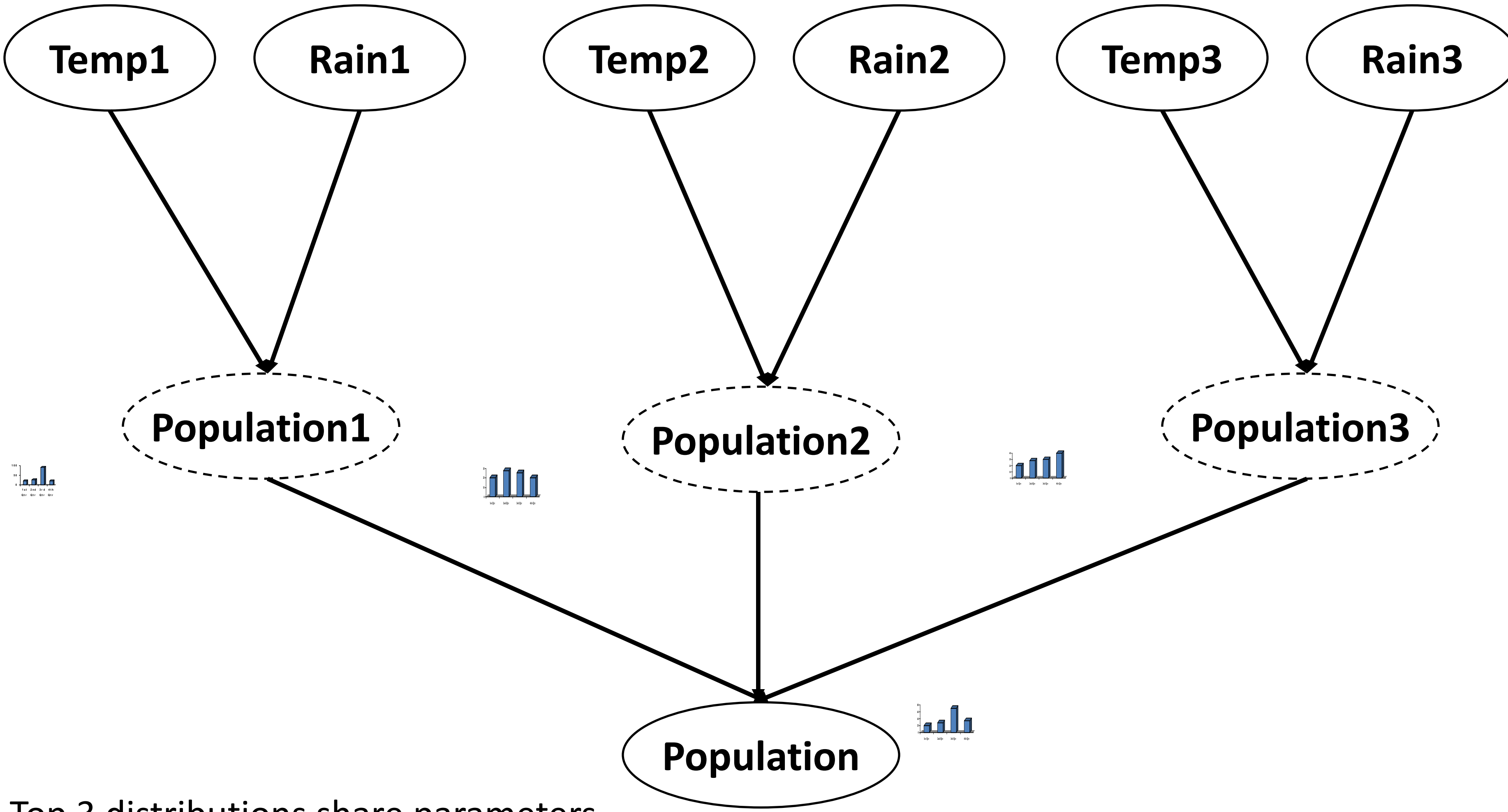
Update parameters (ML, MAP)

Solution 1: Aggregators



Problem: Does not take into account the interaction between Rain and Temp

Solution 2: Combining Rules



- Top 3 distributions share parameters
- The 3 distributions are combined into one final distribution
- Gradient-descent and EM Methods exist

MLN Weight Learning

- Parameter tying: Groundings of same clause

$$\frac{\partial}{\partial w_i} \log P_w(x) = \boxed{n_i(x)} - \boxed{E_w\left[n_i(x)\right]}$$

No. of times clause *i* is true in data

Expected no. times clause *i* is true according to MLN

- It is #P-complete to count the number of true groundings
- Generative learning: Pseudo-likelihood
- Discriminative learning: Cond. likelihood

Slide based on Domingos' talk

Generative Learning

- Function to optimize: $$f(w) = \sum_i w_i n_i(x) - \log Z$$

$$Z = \sum_x \exp\left(\sum_i w_i n_i(x')\right)$$

- Gradient:

$$\frac{\partial}{\partial w_j} f(w) = n_j(x) - \frac{1}{Z} \sum_{x'} \exp\left(\sum_i w_i n_i(x')\right) n_j(x')$$

$$= n_j(x) - \sum_{x'} P(x') \, n_j(x')$$

$$= \underbrace{n_j(x)}_{\text{Counts in training data}} - \underbrace{E[n_j(x)]}_{\text{Weighted sum over all possible worlds}}$$

Counts in training data

Weighted sum over all possible worlds
No evidence, just sets of constants
Very hard to approximate

Slide based on Domingos' talk

Pseudo-likelihood

$$PL(x) = \prod_l P(X_l = x_l \mid MB(x_l))$$

$$\log PL(x) = \sum_l \log P(X_l = x_l \mid MB(x_l))$$

$$P(X_l = x_l \mid MB(x_l)) = \frac{P(x)}{P(x_{[X_l=0]}) + P(x_{[X_l=1]})}$$

$$= \frac{1/Z \exp(\Sigma w_i n_i(x))}{1/Z \exp(\Sigma w_i n_i(x_{[X_l=0]})) + 1/Z \exp(\Sigma w_i n_i(x_{[X_l=1]}))}$$

$$\frac{\partial}{\partial w_j} \log PL(x) = \sum_l n_j(x) - P(X_l = 0 \mid MB(X_l))\, n_j(x_{[X_l=0]}) - P(X_l = 1 \mid MB(X_l))\, n_j(x_{[X_l=1]})$$

$$= \sum_l n_j(x) - E_{x_l'}[n_j(x_{[X_l=x_l']})]$$

Slide based on Domingos' talk

Pseudo-likelihood

$$PL(x) = \prod_l P(X_l = x_l \mid MB(x_l))$$

While effective, still hard to count in many data sets

- Approximate counting techniques exist

(Sarkhel et al. AAAI 2016, Das et al. SDM 2016)

$$\frac{\partial}{\partial w_j} \log PL(x) = \sum_l n_j(x) - P(X_l = 0 \mid MB(X_l))\, n_j(x_{[X_l=0]}) - P(X_l = 1 \mid MB(X_l))\, n_j(x_{[X_l=1]})$$

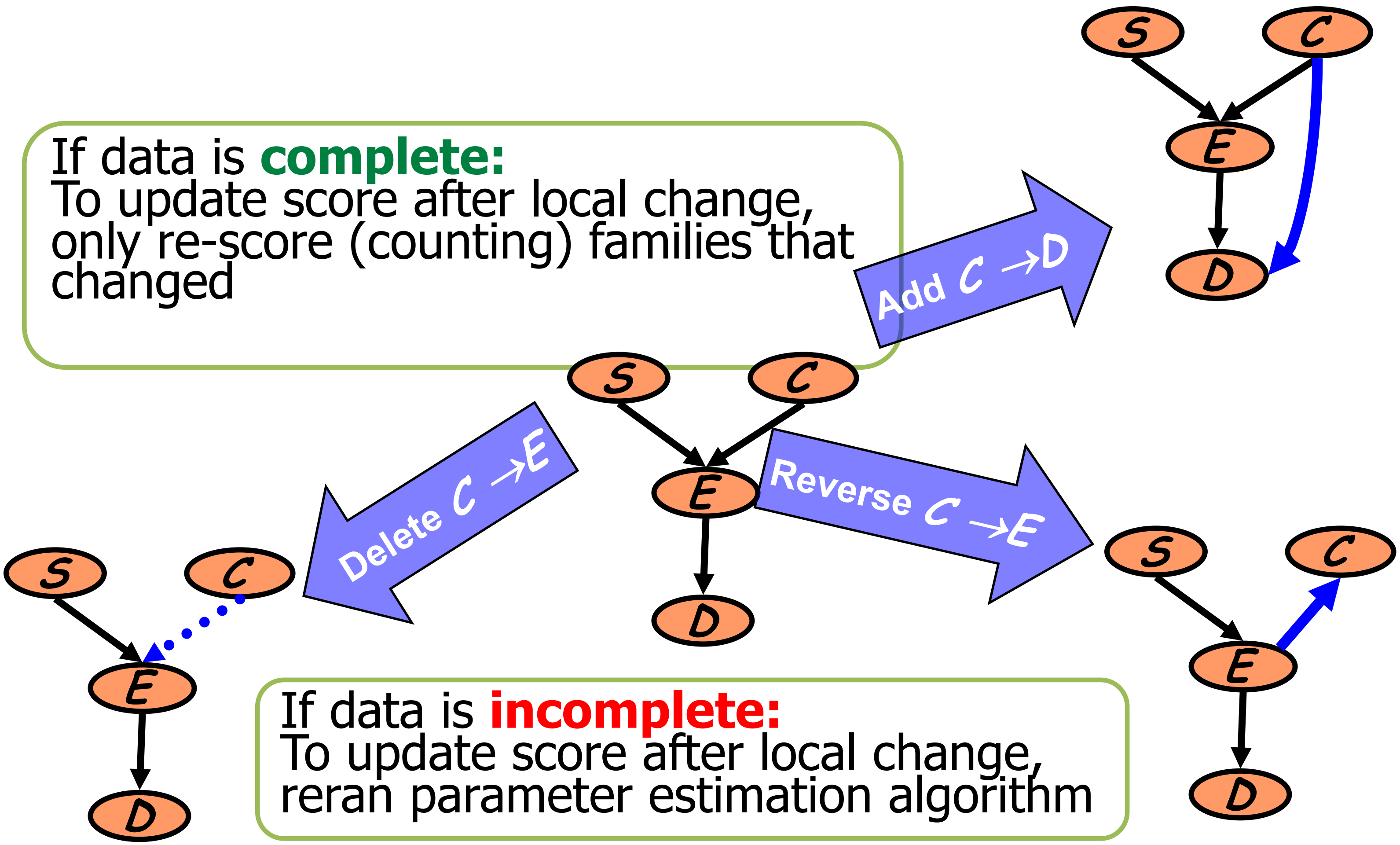
$$= \sum_l n_j(x) - E_{x'_l}[n_j(x_{[X_l=x'_l]})]$$

Slide based on Domingos' talk

- Parameter learning – graphical models
- Parameter learning – StaR AI models
- Structure learning – logical models & graphical models
- Structure learning – StaRAI models

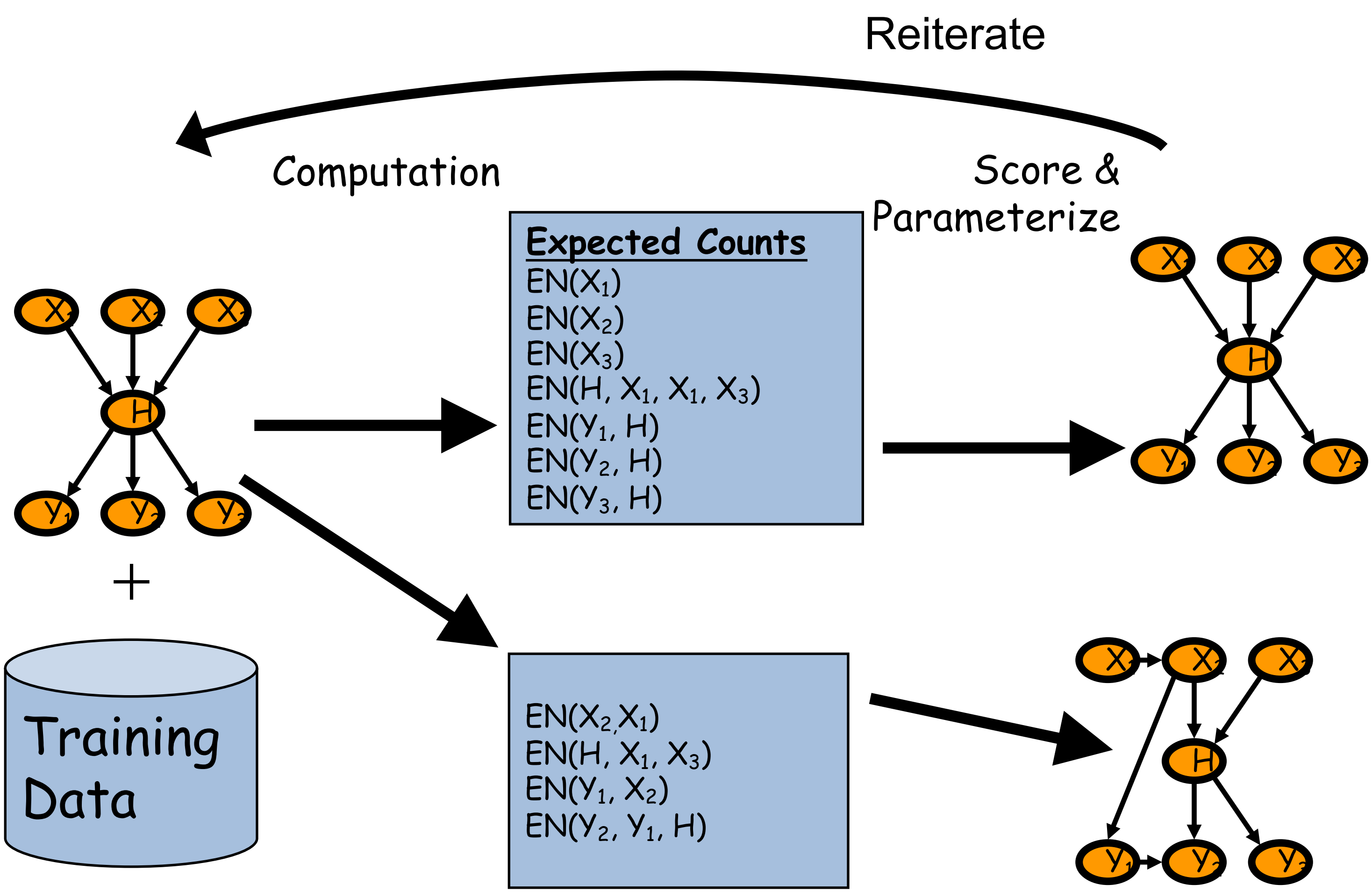
Probabilistic Graphical Models

If data is **complete:**
To update score after local change, only re-score (counting) families that changed



If data is **incomplete:**
To update score after local change, reran parameter estimation algorithm

Structural EM [Friedman et al. 98]



Inductive Logic Programming = Machine Learning + Logic Programming The Problem Specification

[Muggleton, De Raedt JLP96]

- *Given*:
  - *Examples:* first-order atomic formulas (atoms), each labeled positive or negative.
  - *Background knowledge:* definite clause (if-then rules) theory.
  - *Language bias:* constraints on the form of interesting new rules (clauses).

ILP Specification (Continued)

- *Find:*
  - A hypothesis *h* that meets the language constraints and that, when conjoined with *B*, implies (lets us prove) all of the positive examples but none of the negative examples.
- To handle real-world issues such as noise, we often relax the requirements, so that *h* need only entail significantly more positive examples than negative examples.

A Common Approach

- Use a greedy covering algorithm.
  - Repeat while some positive examples remain uncovered (not entailed):
    - Find a *good clause* (one that covers as many positive examples as possible but no/few negatives).
    - Add that clause to the current theory, and remove the positive examples that it covers.

- ILP algorithms use this approach but vary in their method for finding a *good clause*.

- Parameter learning – graphical models
- Parameter learning – StaR AI models
- Structure learning – logical models & graphical models
- Structure learning – StaRAI models

Vanilla SRL Approach [De Raedt, Kersting ALT04]

mutagenic(X) :- atom(X,A,n),charge(A,0.82)

mutagenic(X) :- atom(X,A,c),bond(A,B)

...

=0.882

- Traverses the hypotheses space a la ILP
- Replaces ILP's 0-1 covers relation by a "smooth", probabilistic one [0,1]

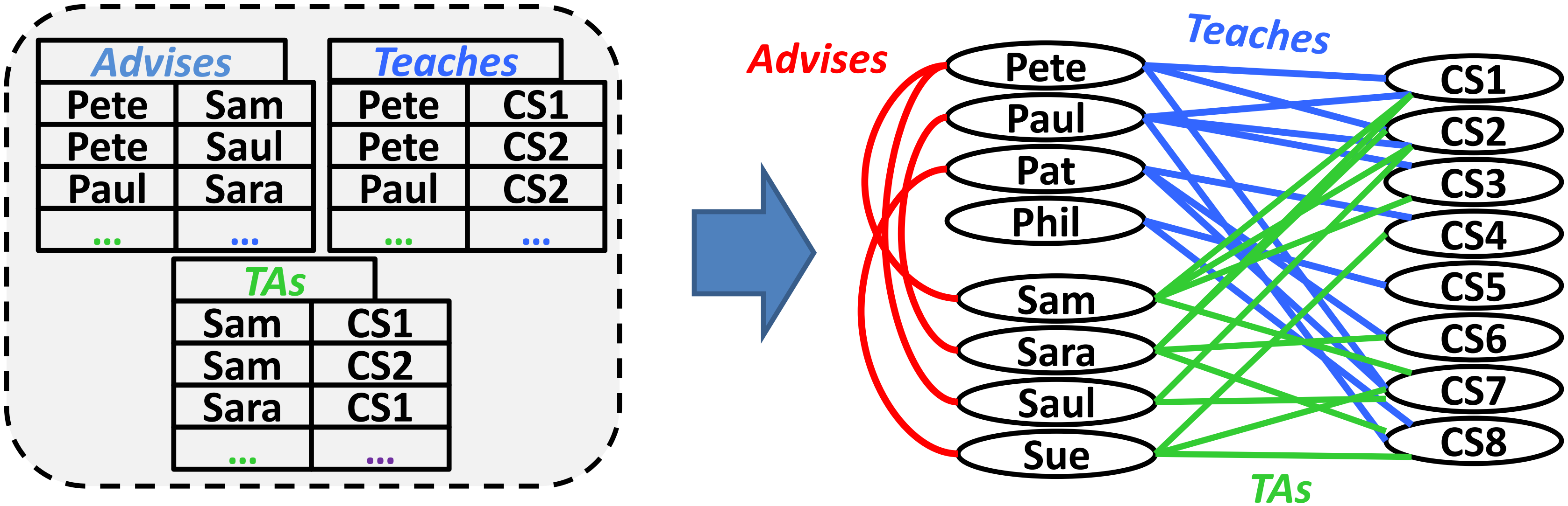
$$
\begin{aligned}
\text{cover}(e, H, B) &= P(e|H, B) \\
\text{cover}(E, H, B) &= \prod_{e \in E} \text{cover}(e, H, B)
\end{aligned}
$$

Structure learning methods for MLNs

- Top-down approach:
  - GSL[Kok & Domingos, 2005], DSL[Biba et al., 2008]
  - Start from unit clauses and search for new clauses
- Bottom-up approach:
  - BUSL [Mihalkova & Mooney, 2007], Hypergraph Lifting [Kok & Domingos, 2009], Structural Motifs [Kok & Domingos , 2010]
  - Use data to generate candidate clauses
- Max-Margin Approach:
  - Discriminative learning [Huynh & Mooney, 2008]
  - Effectively learns horn clauses
  - Uses regularization to force parameters to zero
  - Later extended to online setting

Learning via Hypergraph Lifting

[Kok & Domingos, ICML'09]



- Relational DB can be viewed as hypergraph
  - Nodes ´ Constants
  - Hyperedges ´ True ground atoms

Learning via Hypergraph Lifting

[Kok & Domingos, ICML'09]

Advises

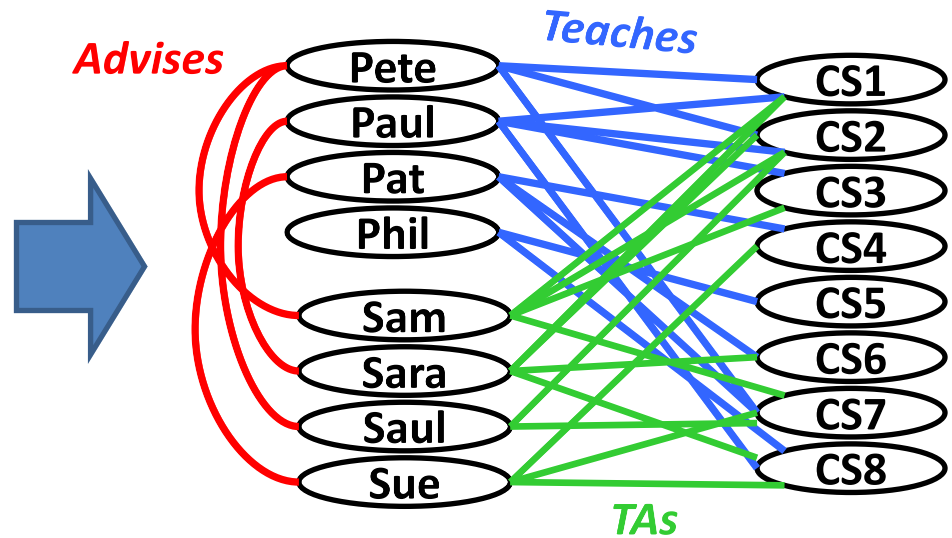
Pete	Sam
Pete	Saul
Paul	Sara
…	…

Teaches

Pete	CS1
Pete	CS2
Paul	CS2
…	…

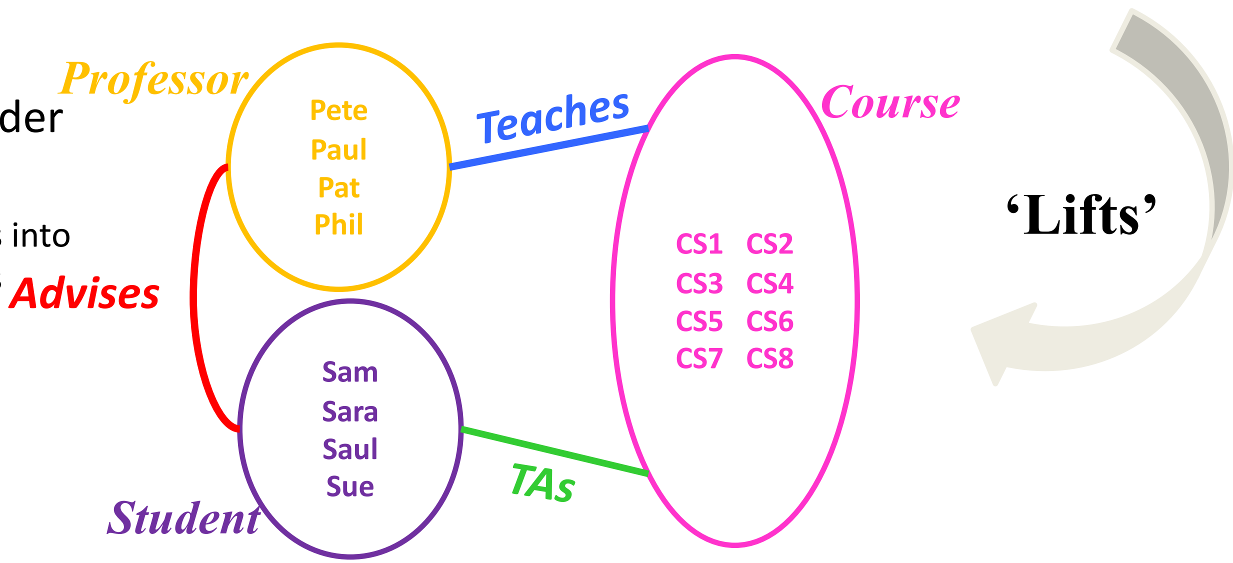
TAs

Sam	CS1
Sam	CS2
Sara	CS1
…	…



Clustering using "2nd"-order MLNs

- Jointly clusters nodes into higher-level concepts
- Clusters hyperedges



Learning via Hypergraph Lifting

[Kok & Domingos, ICML'09]

Advises

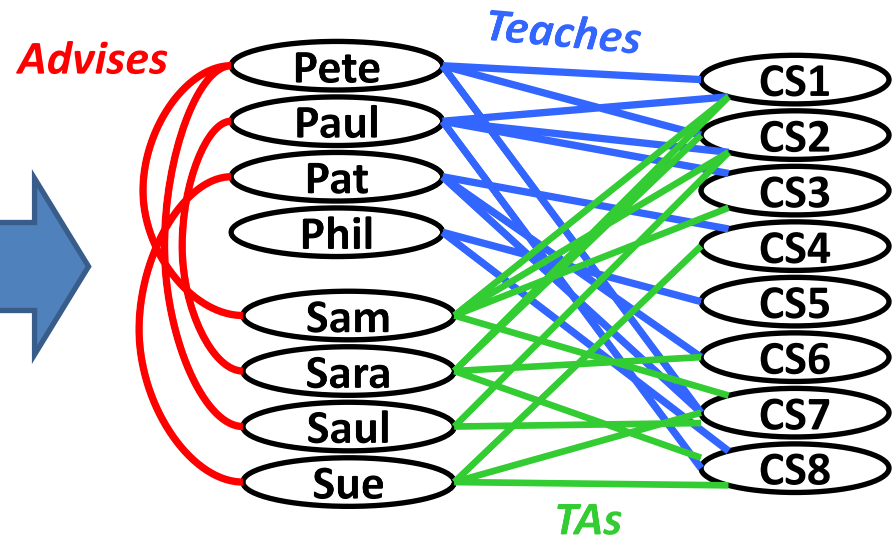
Pete	Sam
Pete	Saul
Paul	Sara
…	…

Teaches

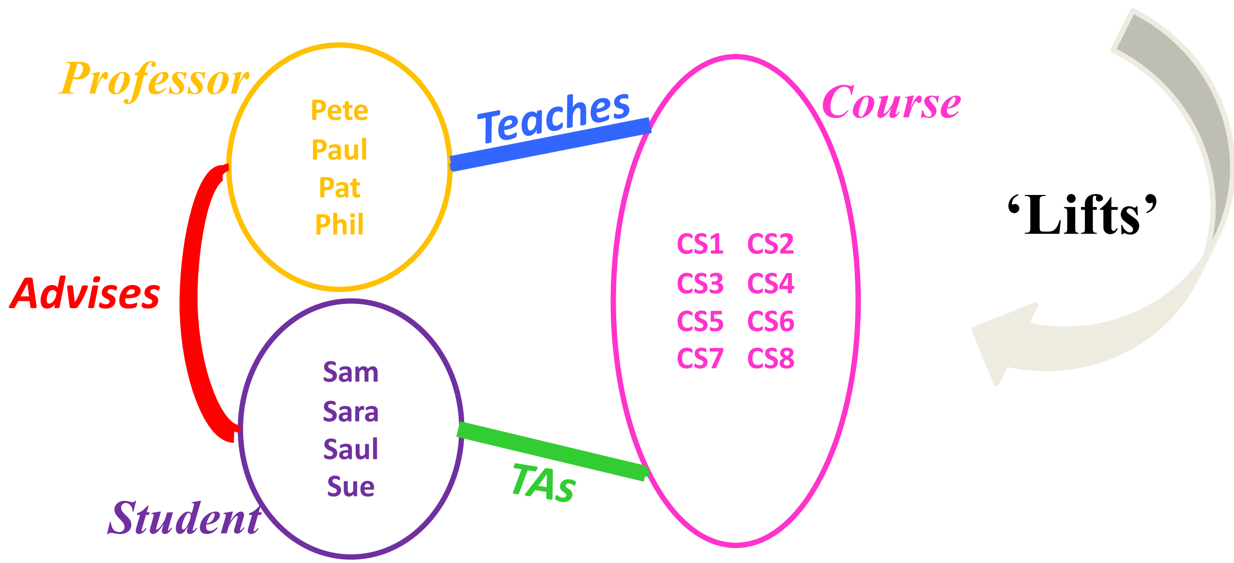
Pete	CS1
Pete	CS2
Paul	CS2
…	…

TAs

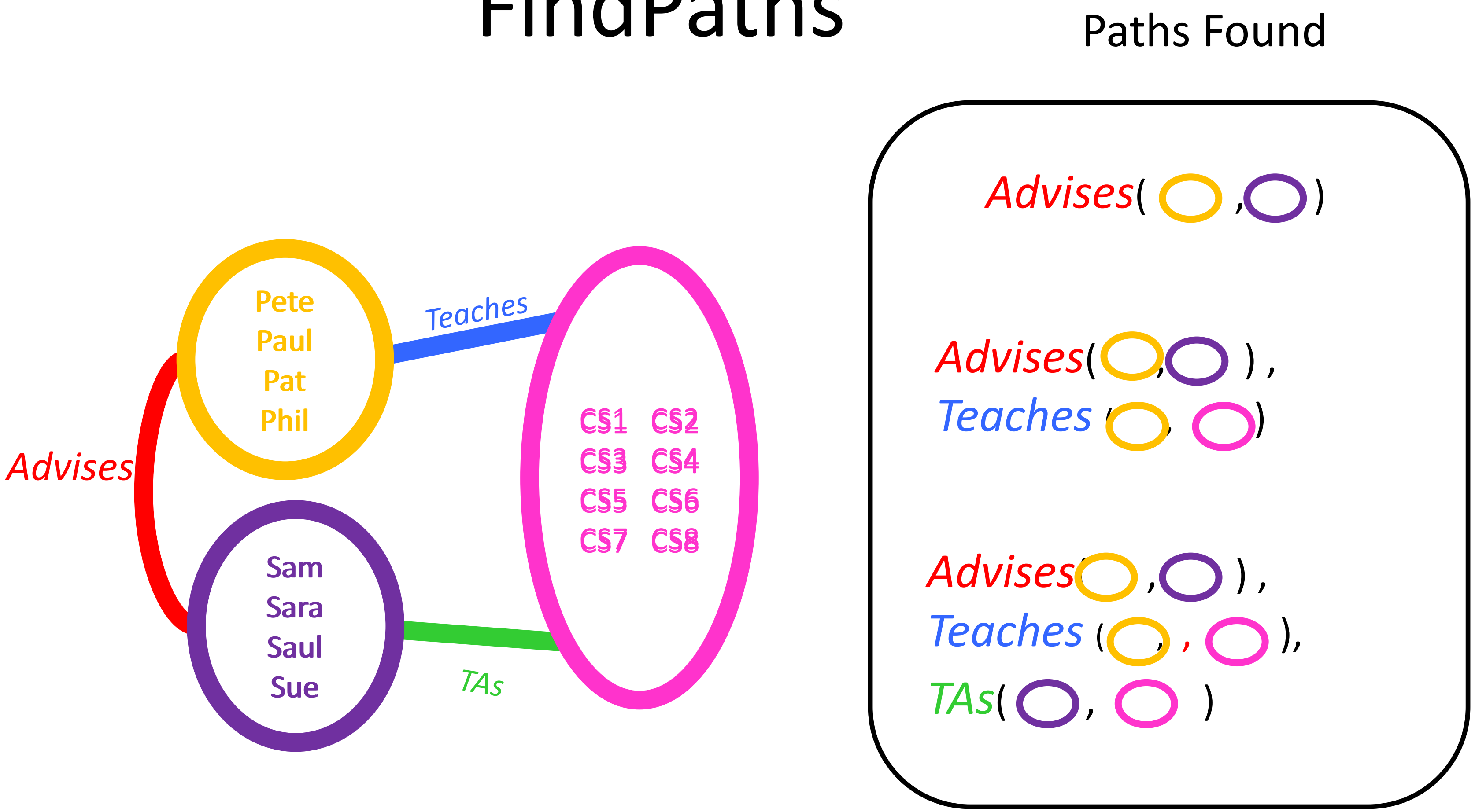
Sam	CS1
Sam	CS2
Sara	CS1
…	…



Trace paths & convert paths to first-order clauses



FindPaths


Paths Found
Pete
Paul
Pat
Phil
Teaches
Advises
Sam
Sara
Saul
Sue
TAs
CS1 CS2
CS3 CS4
CS5 CS6
CS7 CS8
Advises(,)
Advises(,) ,
Teaches (,)
Advises(,) ,
Teaches (, ,),
TAs(,)

Clause Creation

Advises(Pete Paul Pat Phil , Sam Sara Saul Sue) ,

and *Teaches*(Pete Paul Pat Phil , CS1 CS2 CS3 CS4 CS5 CS6 CS7 CS8) ,

and *TAs*(Sam Sara Saul Sue , CS1 CS2 CS3 CS4 CS5 CS6 CS7 CS8)

Clause Creation

Advises(p , s)

and *Teaches*(p , c)

and *TAs*(s , c)

Clause Creation

Advises(p, s) and *Teaches*(p, c) and *TAs*(s, c)

Advises(p, s) V not *Teaches*(p, c) V not*TAs*(s, c)

Advises(p, s) V *Teaches*(p, c) V not*TAs*(s, c)

…

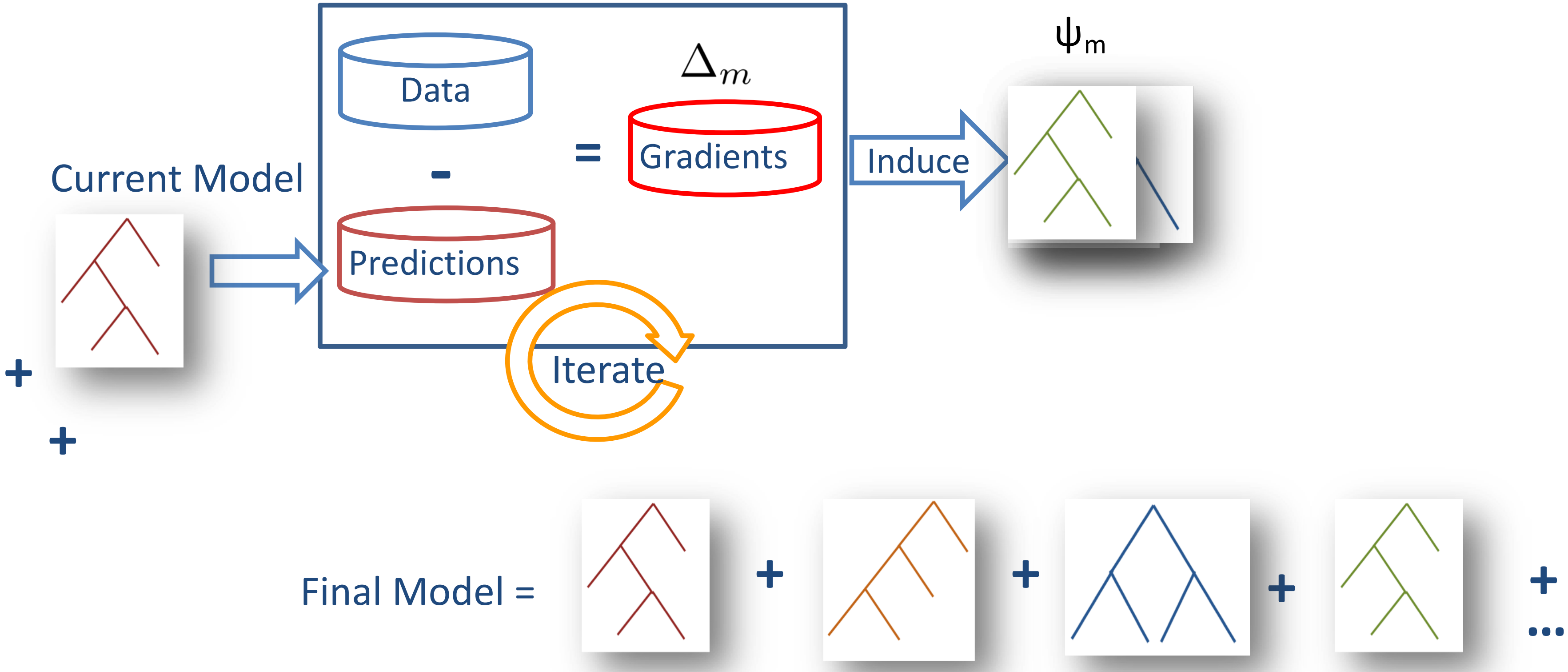
Outline (Higher Level)

- State-of-the-art Structure Learning
- Effective Learning
- Advice Taking
- Actively Seeking Advice

Functional Gradient Boosting

Key Insight: Learn multiple weak models rather than a single complex model

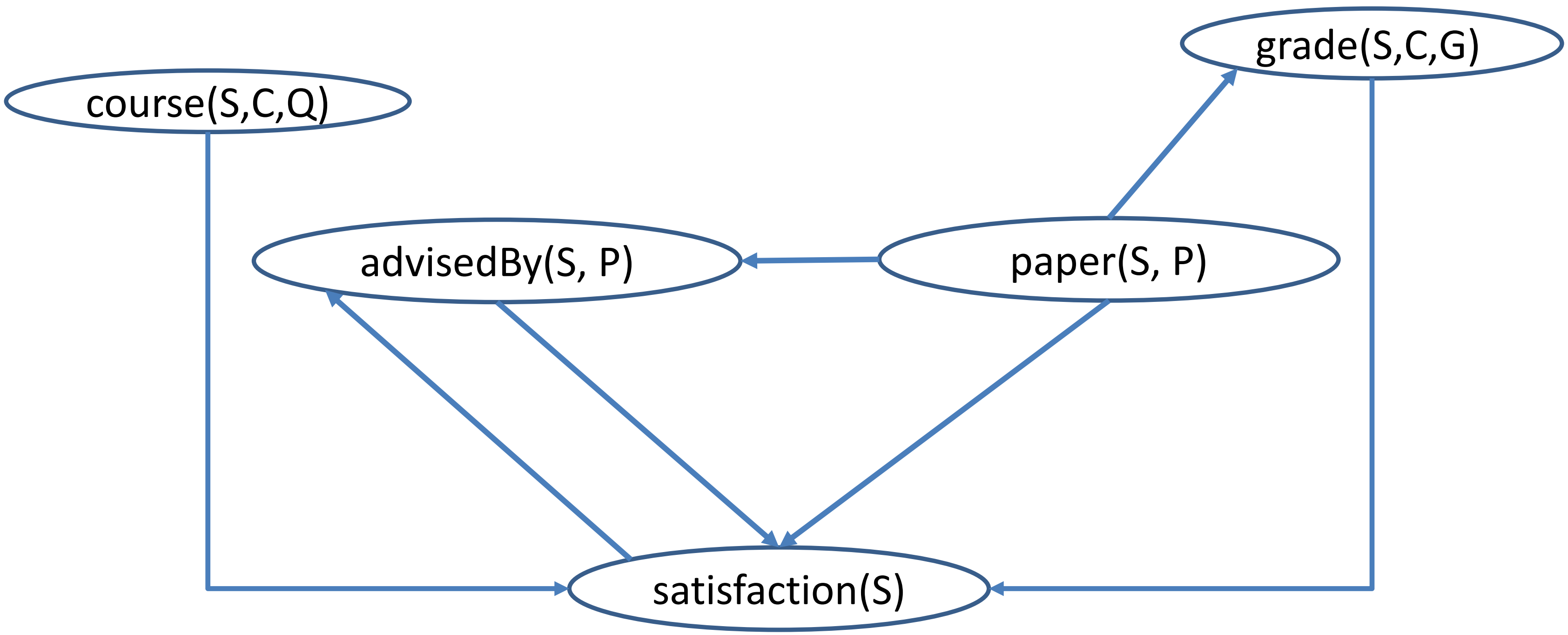
– First explored in the context of StaRAI by Kersting & Driessens



Friedman et al 2001, Dietterich et al. 2004, Kersting & Driessens 08, Natarajan et al. MLJ 2012, Springer Brief '15

Relational Dependency Network

- Cyclic directed graphs
- Approximated as product of conditional distributions



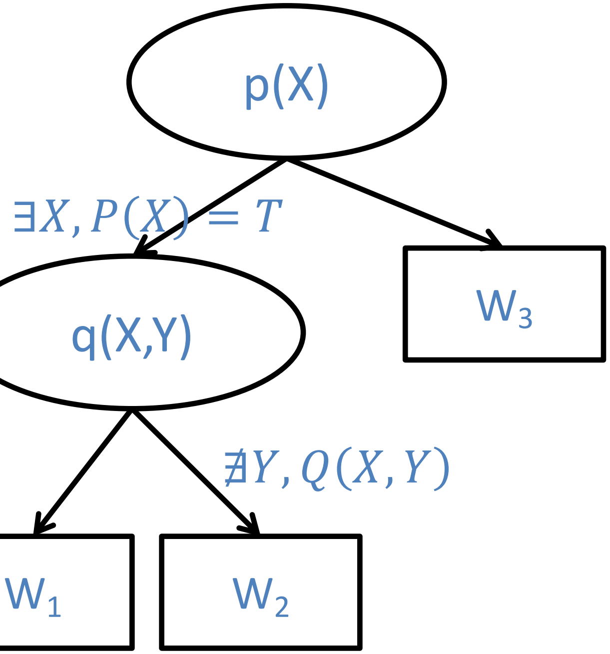
J. Neville and D. Jensen ’07,
D. Heckerman et al. ‘00

Functional Gradients for RDNs

x	Δ
target(x1)	0.7
target(x2)	-0.2
target(x3)	-0.9

- Probability of an example

$$P(x_i = true|\mathbf{Pa}(x_i)) = \frac{e^{\psi(x_i;\mathbf{Pa}(x_i))}}{e^{\psi(x_i;\mathbf{Pa}(x_i))} + 1}$$



- Functional gradient
  - Maximize

$$LL(\mathbf{X} = \mathbf{x}) = \sum_{x_i \in \mathbf{x}} \log P(x_i|\mathbf{Pa}(x_i))$$

  - Gradient of log-likelihood w.r.t ψ

$$\Delta(x_i) = \frac{\partial \log P(\mathbf{X} = \mathbf{x})}{\partial \psi(x_i; \mathbf{Pa}(x_i))} = I(x_i = true; \mathbf{Pa}(x_i)) - P(x_i = true; \mathbf{Pa}(x_i))$$

  - Sum all gradients to get final ψ

$$\psi_m = \psi_0 + \Delta_1 + ... + \Delta_m$$

Markov Logic Networks

Richardson & Domingos '05

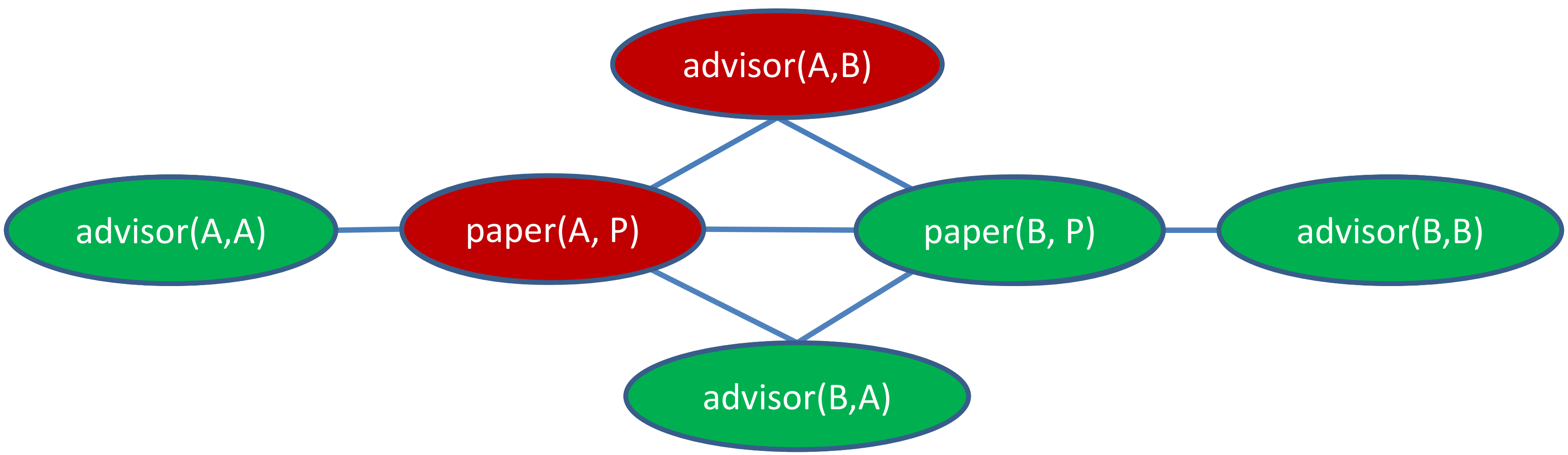
- Weighted logic

1.1	$\forall x, y, p \; advisor(x, y), paper(x, p) \Rightarrow paper(y, p)$

$$P(currInst) = \frac{1}{Z}\exp\left(\sum_i w_i \, n_i(currInst)\right)$$

Weight of formula *i*

Number of true groundings of formula *i* in *current instance*



Learning MLNs

$$P(currInst) = \frac{1}{\boxed{Z}} \exp\left(\sum_i \boxed{w_i} \, \boxed{n_i(currInst)} \right)$$

Weight of formula *i*

Number of true groundings of formula *i* in *current Instance*

- Normalization term sums over all world states

$$PLL(\mathbf{X} = \mathbf{x}) \quad = \quad \sum_{x_i \in \mathbf{x}} \log P(x_i | \mathbf{MB}(x_i))$$

- Learning approaches maximiz pseudo-
logli

Key Insight: View MLNs as sets of RDNs

Functional gradient for MLNs

RDN

Model defined as a product of conditional distributions

$x_i \in \mathbf{x}$

- Probability of x

Each conditional distribution can be learned independently

- $\Psi(\mathrm{x})$

Regression tree uses aggregators (e.g. Exists)

MLN

- Learning optimizes a product of conditional distributions

- Probability of x

Each conditional distribution not learned independently

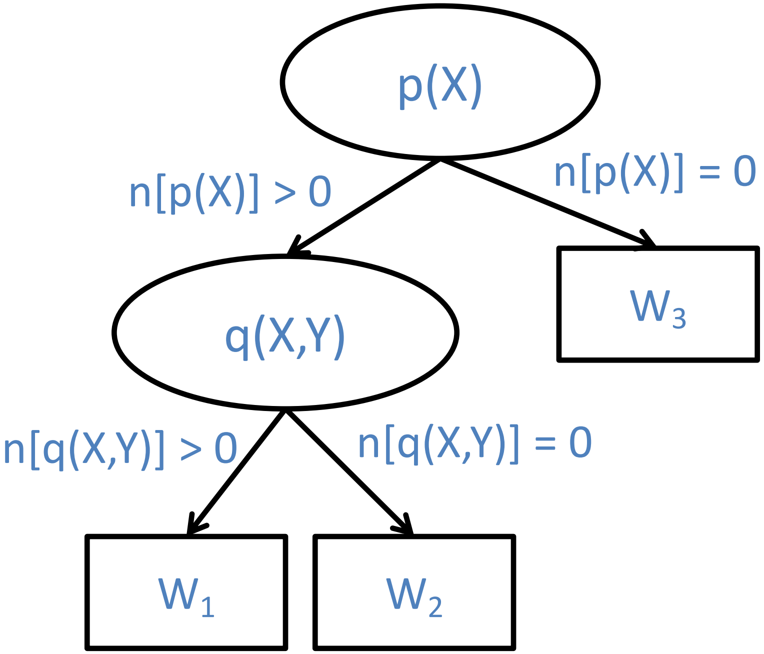
- $\Psi(\mathrm{x})$

Regression tree scales output by the number of groundings (Shavlik & Natarajan '09)

$- n_j(x_i = 0; \mathbf{MB}(x_i))$

Khot et al, '11

MLN from Trees



$$w_1 : p(X), q(X,Y) \to target(X)$$
$$w_2 : p(X), \neg(\exists Y, q(X,Y)) \to target(X)$$
$$w_3 : \neg p(X) \to target(X)$$

Learning Clauses

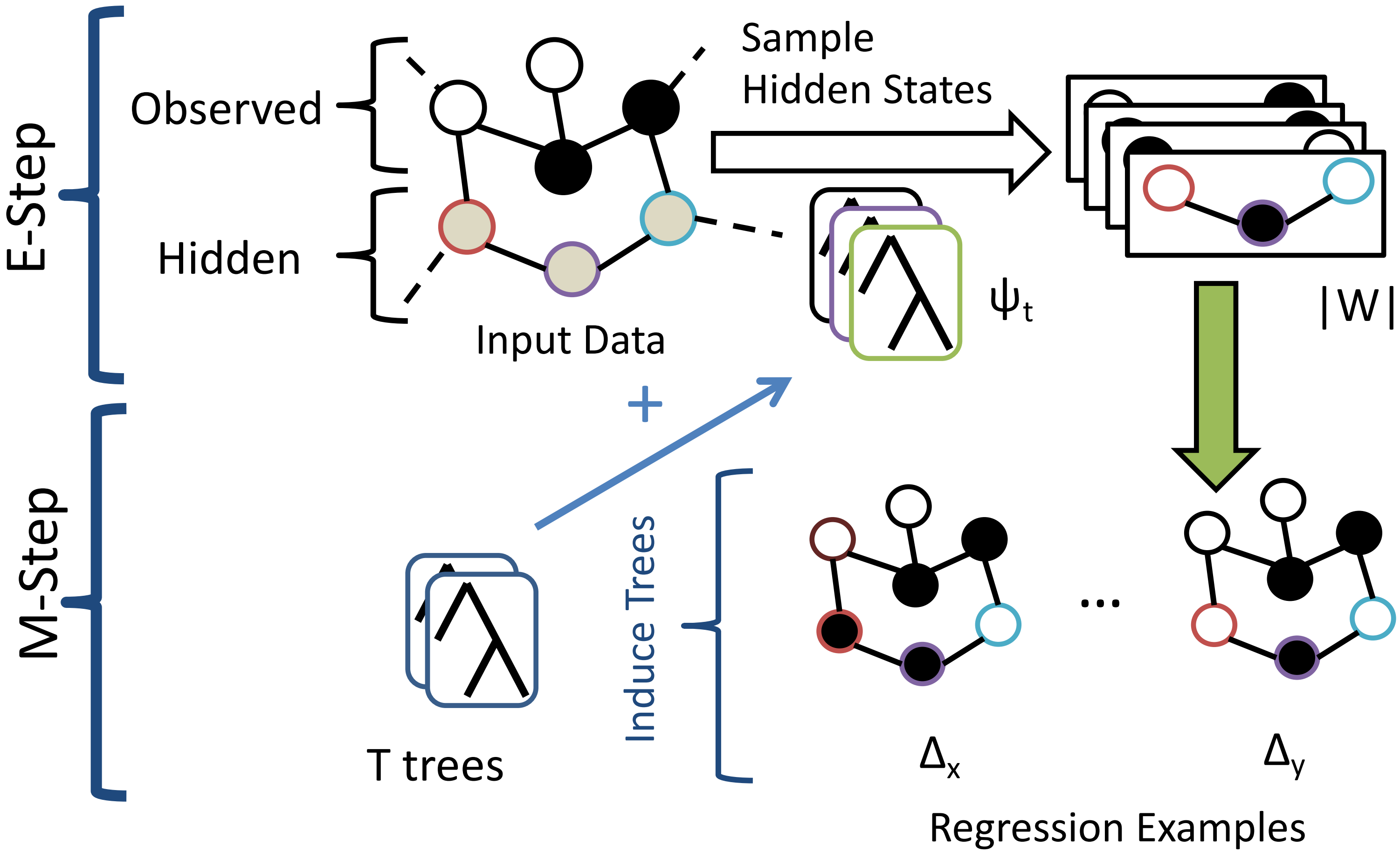
- Same as squared error for trees
- Force weight on false branches (w_3 ,w_2) to be 0
- Hence no existential vars needed

$$w_1 \quad : \quad p(X), q(X,Y) \to target(X)$$

Similar algorithm for learning **Relational Logistic Regression** (Ramanan et al., KR 18 - to appear)

Break!!!

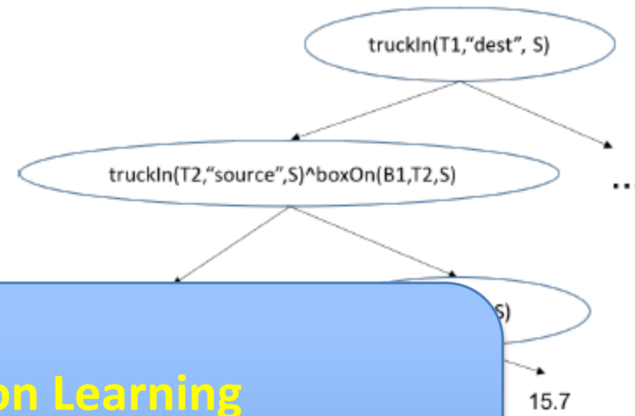
RFGB-EM



Khot et al MLJ ‘15

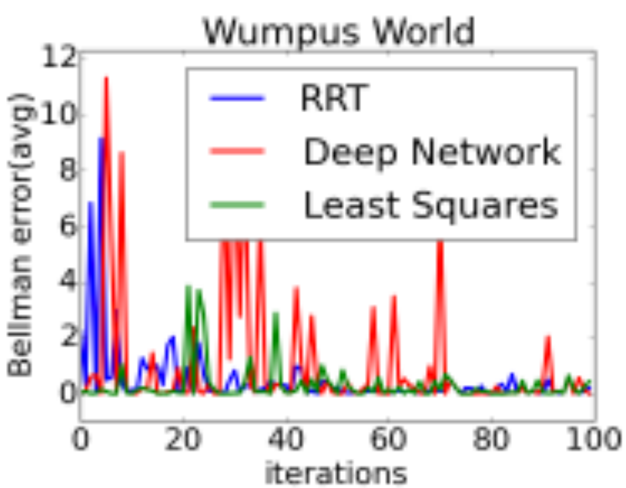
Relational RL (under review)

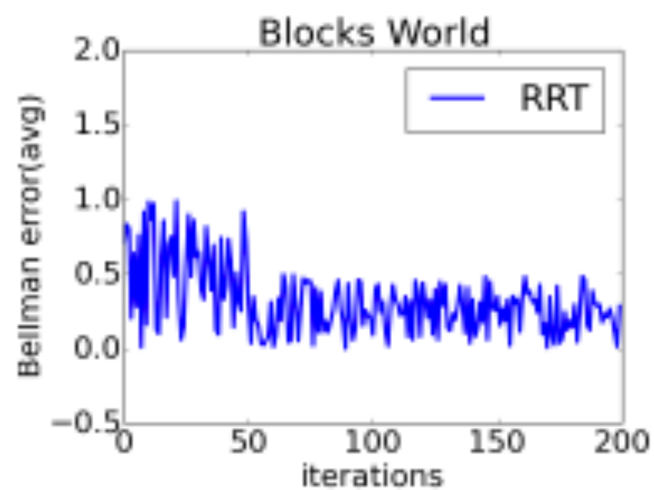
- Function Approximators in Relational Domains
- Fitted Q-iteration
- Huber



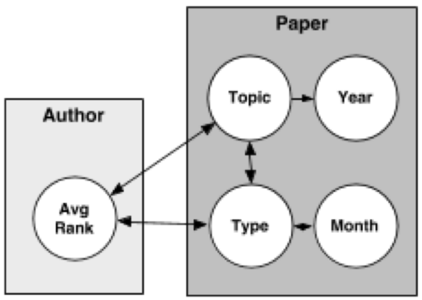
Relational Imitation Learning
(Natarajan et al. 2011)
Learning Relational Policies

nctions

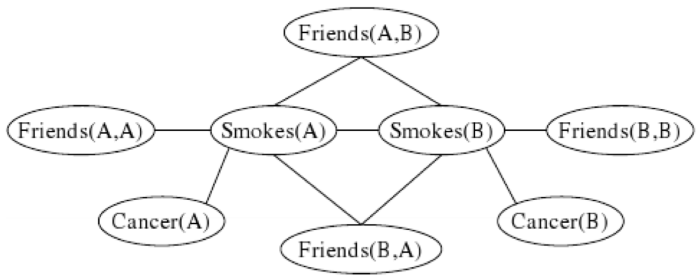




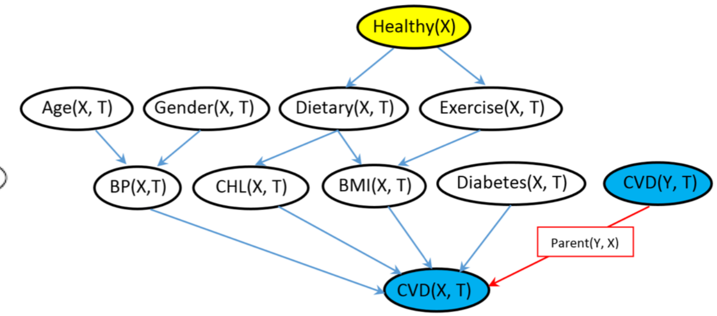
What can be learned?



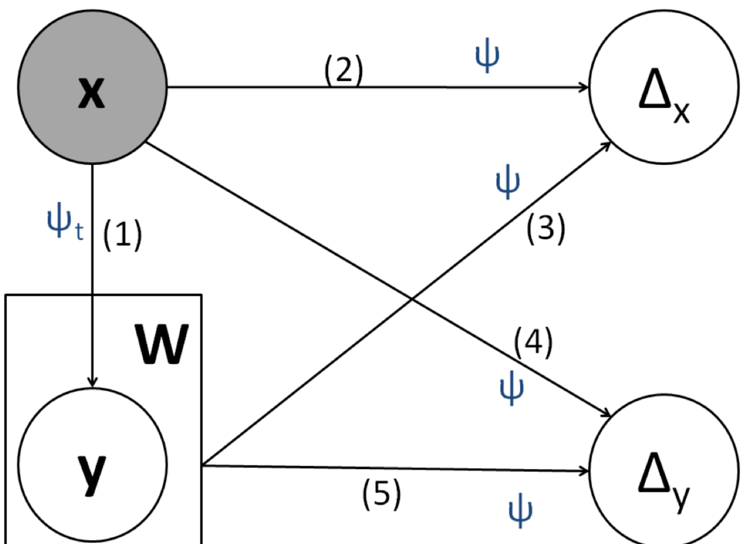
Relational Dependency network



Markov Logic network

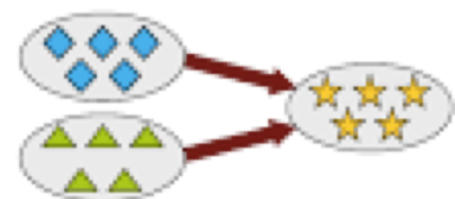


Relational CTBN



Learning with Hidden data

Imitation Learning

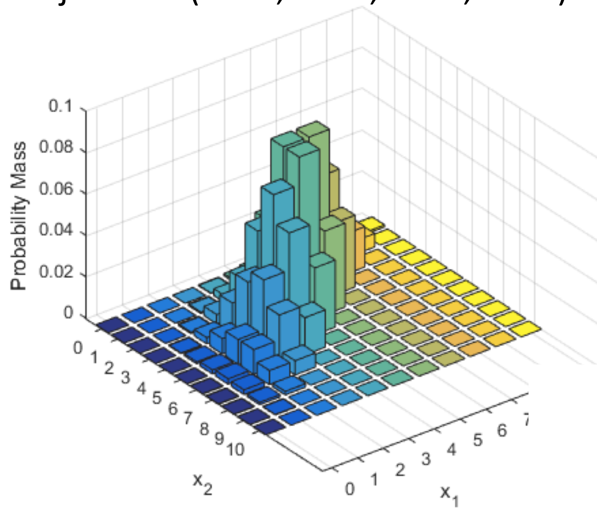


Transfer Learning

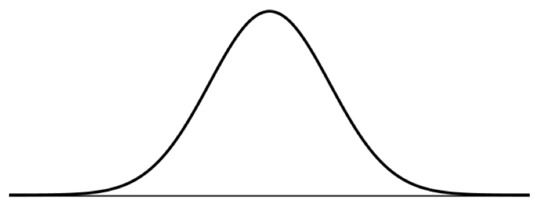
Natarajan et al (2010, 2011, 2012,2013) Khot et al (2011, 2013), Yang et al (2016)

What can be learned?

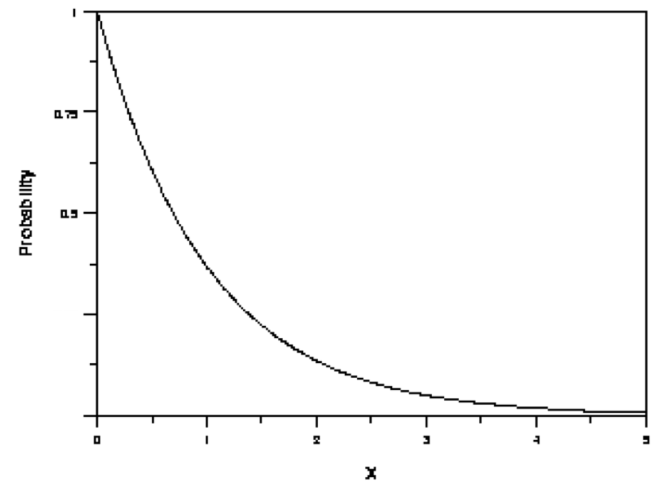
Natarajan et al (2010, 2011, 2012,2013) Khot et al (2011, 2013), Yang et al (2016), Hadiji et al (2015), Yang et al (2017)



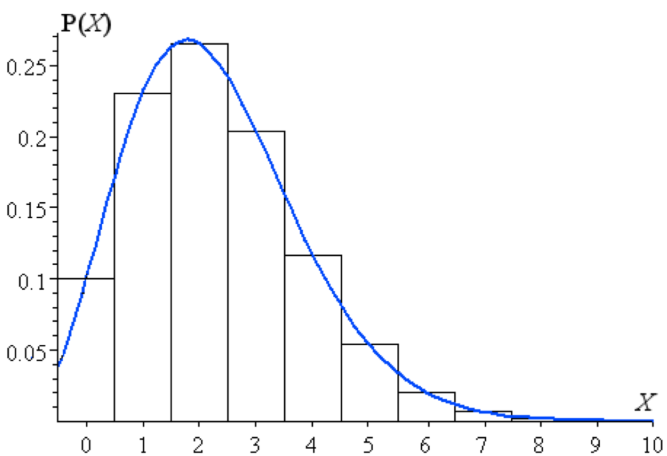
Multinomial



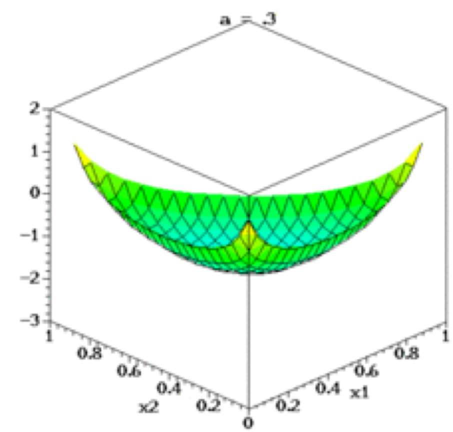
Gaussian



Exponential



Poisson



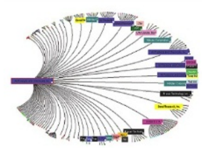
Dirichlet

Impressive Results on Standard Domains

Predicting the advisor for a student

Algo	Likelihood	AUC-ROC	AUC-PR	Time
Boosting	0.810	0.961	0.930	9s
RPT	0.805	0.894	0.863	1s
MLN	0.730	0.535	0.621	93 hrs

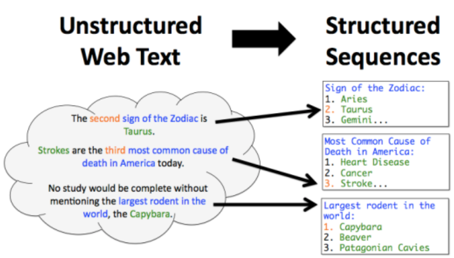
Movie Recommendation



Citation Analysis

Discovering Relations

Learning from Demonstrations



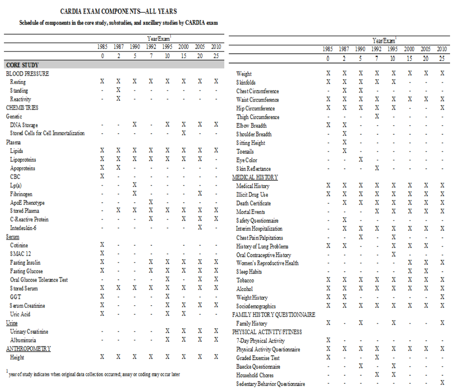
Information Extraction

Scale of Learning Structure

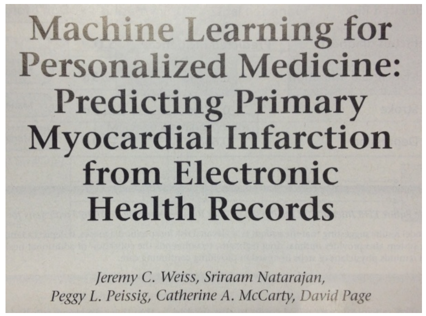
- 10 G facts describing the recommendations
- 115k drug-disease interactions
- 11 M facts on NLP tasks

Natarajan et al. MLJ'12, Khot et al. ICDM '11, Natarajan et al. IJCAI '11, Natarajan et al. IAAI '13
Weiss et al. IAAI '12 AI Magazine '12, Natarajan et al. IJMLC '13, Khot et al. MLJ' 14

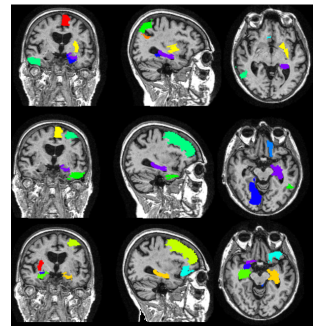
Several **Real** Applications



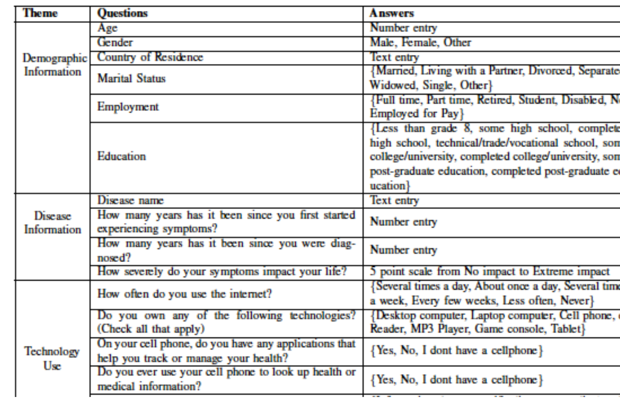
Cardiovascular Study



EHR

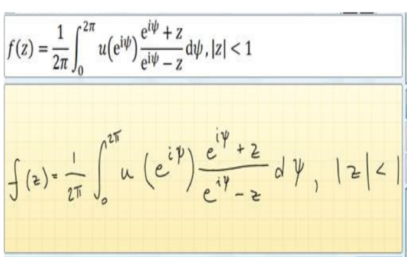


Alzheimer's



Theme	Questions	Answers
Demographic Information	Age	Number entry
	Gender	Male, Female, Other
	Country of Residence	Text entry
	Marital Status	{Married, Living with a Partner, Divorced, Separated, Widowed, Single, Other}
	Employment	{Full time, Part time, Retired, Student, Disabled, Not Employed for Pay}
	Education	{Less than grade 8, some high school, completed high school, technical/trade/vocational school, some college/university, completed college/university, some post-graduate education, completed post-graduate education}
Disease Information	Disease name	Text entry
	How many years has it been since you first started experiencing symptoms?	Number entry
	How many years has it been since you were diagnosed?	Number entry
	How severely do your symptoms impact your life?	5 point scale from No impact to Extreme impact
Technology Use	How often do you use the internet?	{Several times a day, About once a day, Several times a week, Every few weeks, Less often, Never}
	Do you own any of the following technologies? (Check all that apply)	{Desktop computer, Laptop computer, Cell phone, e-Reader, MP3 Player, Game console, Tablet}
	On your cell phone, do you have any applications that help you track or manage your health?	{Yes, No, I dont have a cellphone}
	Do you ever use your cell phone to look up health or medical information?	{Yes, No, I dont have a cellphone}

Predicting rare diseases/PPD from survey/sensor data



Handwriting Recognition

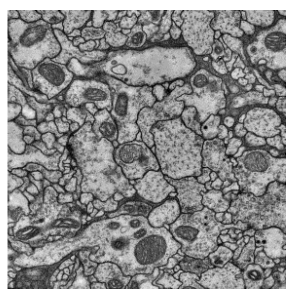
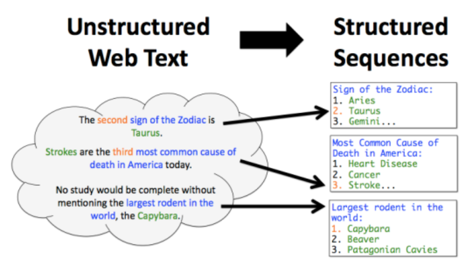
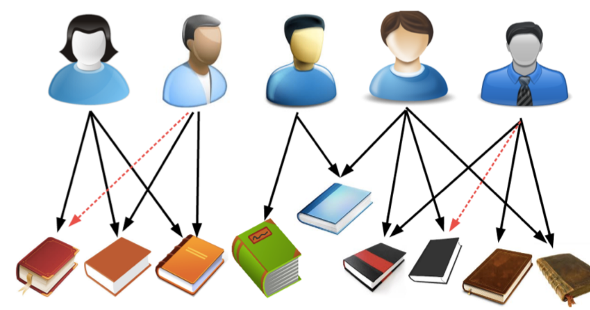


Image Segmentation/ Classification



Information Extraction



Recommendation System

Weiss et al (2012,2013). Natarajan et al (2013,2012, 2014, 2015), Shivram et al (2014), Picado et al (2014) Soni et al (2016), Viswanathan et al (2016), Odom et al (2014,2015a, 2015b), Yang et al (KBS 2017)

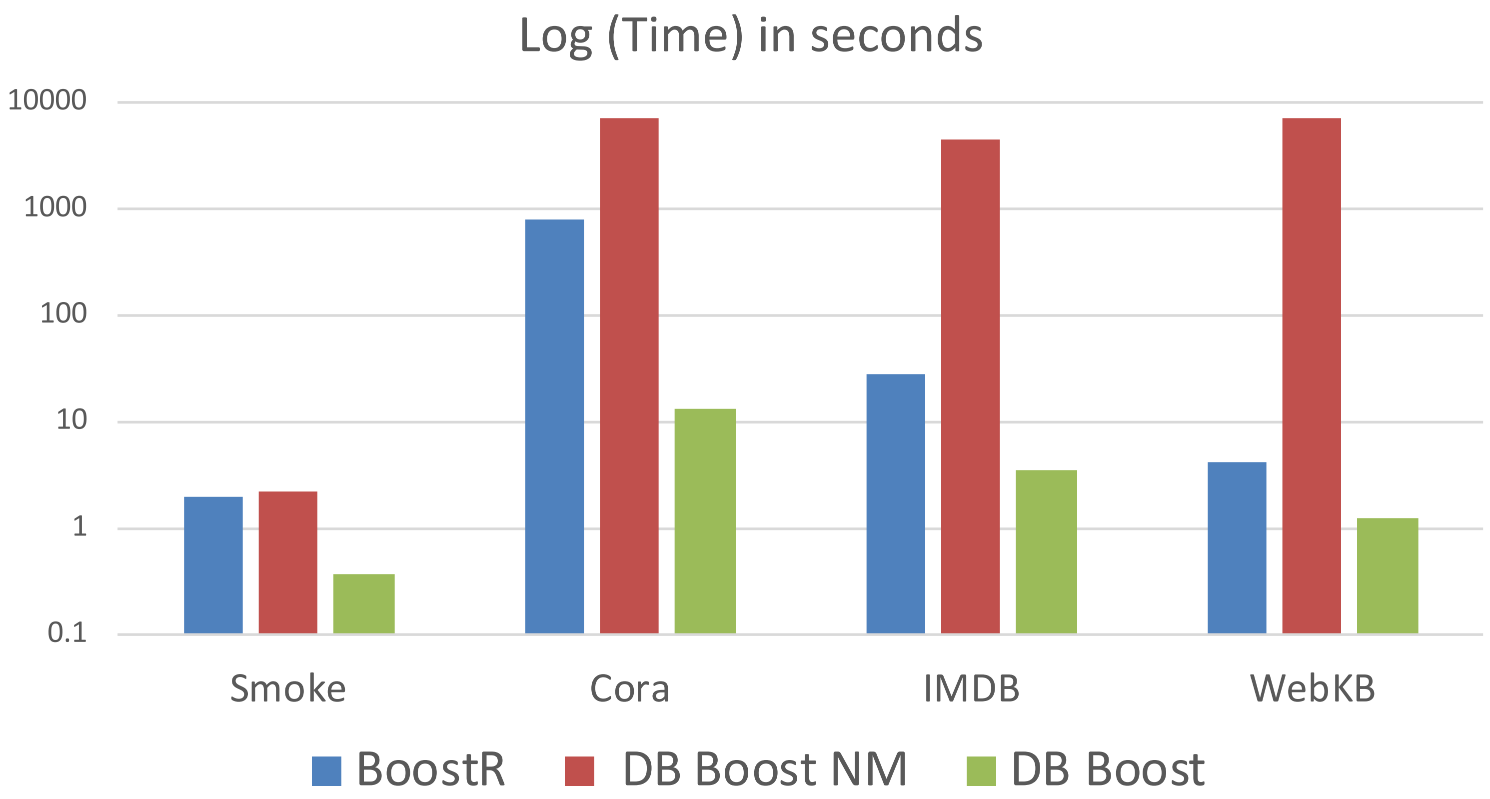
Scaling Boosting

Efficient counting using Databases

1. Encode the data into relational database
2. Reformulate counting as join queries
3. Reuse the join query results for different clauses with common predicates and in subsequent induction steps
4. Use modes to restrict the search process

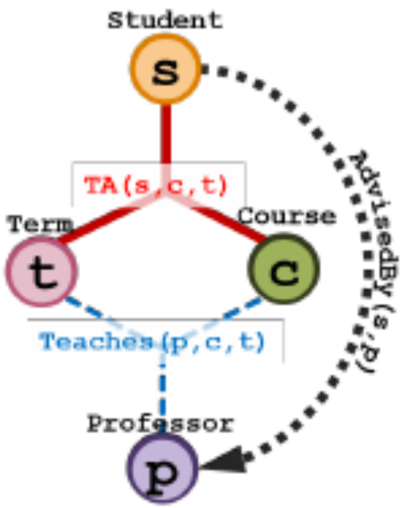
Malec et al. ’16 (Best Student Paper)

DB Boost Results



Approximate Counting via Graph DB

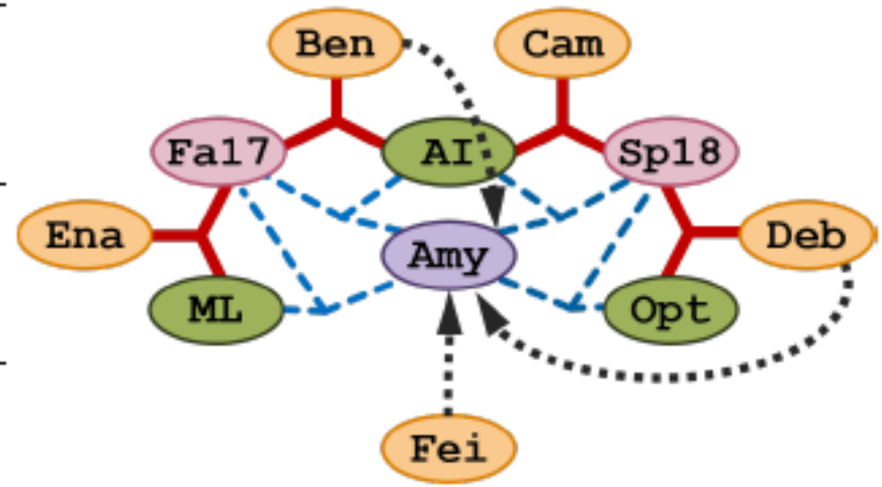
(Das et al., under review)



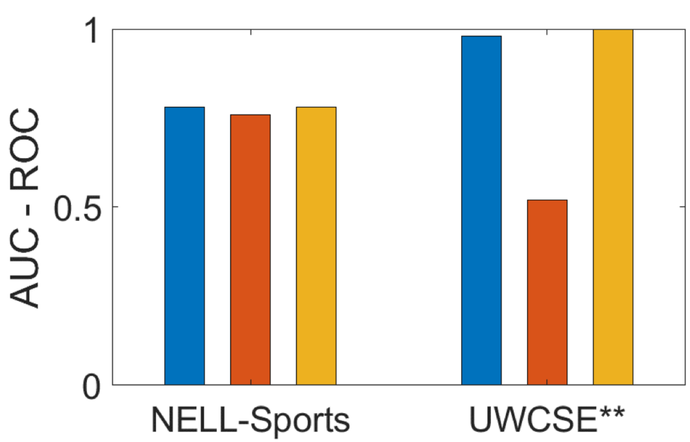
Teaches(Amy, AI, Fa17),
Teaches(Amy, ML, Fa17)
Teaches(Amy, AI, Sp18),
Teaches(Amy, Opt, Sp18)

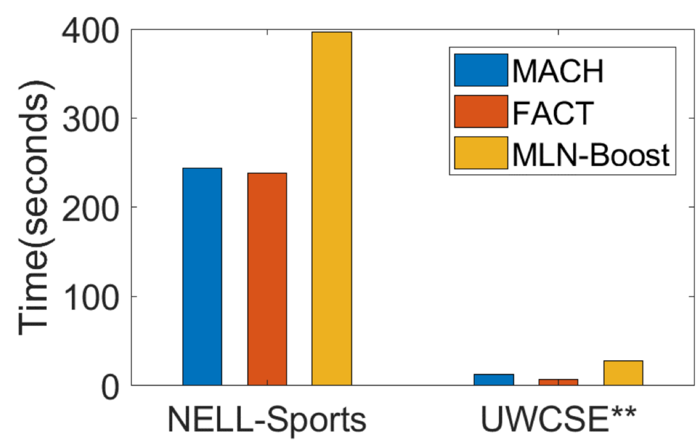
TA(Ben, AI, Fa17),
TA(Ena, ML, Fa17)
TA(Cam, AI, Sp18),
TA(Deb, Opt, Sp18)

AdvisedBy(Ben, Amy),
AdvisedBy(Deb, Amy),
AdvisedBy(Fei, Amy)



Compute summary statistics of the motifs on the resulting hypergraph
Approximate the exact counts with these statistics





Related work by Das '16, Venugopal et al, '15 (MLNs)

Try it yourself

- https://starling.utdallas.edu/software/boostsrl/

Tutorial

- https://starling.utdallas.edu/software/boostsrl/wiki/

As with the standard gradient-boosting approach, our approach turns the model-learning problem to learning a sequence of regression models. The key difference to the standard approaches is that we learn relational regression models (i.e. regression models that operate on relational data). We assume the data to be in predicate-logic format and the output are essentially first-order regression trees where the inner nodes contain conjunctions of logical predicates.

Latest Release	License	Wiki
tag v1.0.0	license GPL-3.0	Documentation

Getting Started

Prerequisites:

- Java (tested with *openjdk 1.8.0_144*)

Installation:

- Download stable jar file.
- Download stable source with git.
  `git clone -b master https://github.com/boost-starai/BoostSRL.git`
- Nightly builds with git.
  `git clone -b development https://github.com/boost-starai/BoostSRL.git`

Basic Usage

```
[hayesall@hawk Datasets]$ ls -R Cora/
Cora/:
cora_bk.txt  test  train

Cora/test:
test_bk.txt  test_facts.txt  test_neg.txt  test_pos.txt

Cora/train:
train_bk.txt  train_facts.txt  train_neg.txt  train_pos.txt
[hayesall@hawk Datasets]$
```

BoostSRL assumes that data are contained in files with data structured in predicate-logic format.

Positive Examples:

```
father(harrypotter,jamespotter).
father(ginnyweasley,arthurweasley).
father(ronweasley,arthurweasley).
...
```

Negative Examples:

Outline

- State-of-the-art Structure Learning
- Effective Learning
- Advice Taking in SRL
- Actively Seeking Advice

Humans can do more (or less)!

- SRL uses FOL as underlying representation
  - Used for designing classical AI systems
- SRL research is primarily data-driven or fully expert-driven

Human involvement in SRL

Examples

Full Model

"Mere labeler"

Active Learning/ Feature selection

Advice giving

Weight Learning

All knowing Oracle

RFGB

Structure Learning methods

EM

Pure reasoning

Advice for SRL

Goal: Expand Knowledge-Based Systems to SRL

Quantity of Expert Knowledge

Expert Systems

Leverage years of experience
Intuitive method to convey knowledge

Knowledge-Based Learning
KBSVMs
KBANNs
KBIRL

Combine knowledge and data
Correct systematic noise
Build robust models

Properly leverage high-quality data
Requires little human intervention

Standard Machine Learning
SVMs
ANNs
FGB

Quantity of Training Data

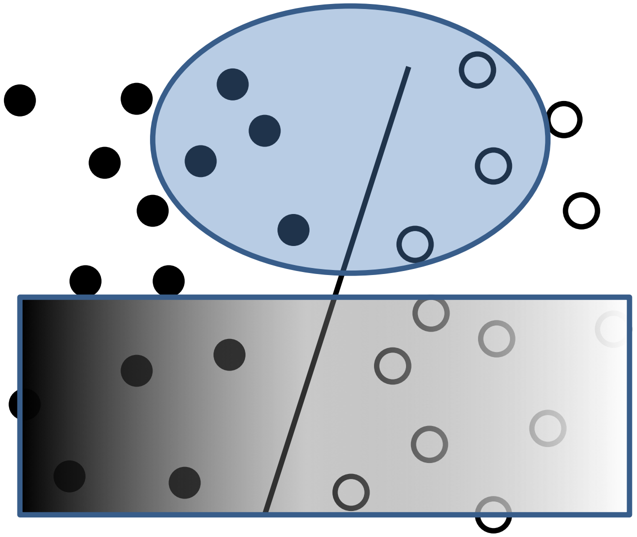
Fung et al. 2002, Towell and Shavlik 1994, Kunapuli et al. 2013

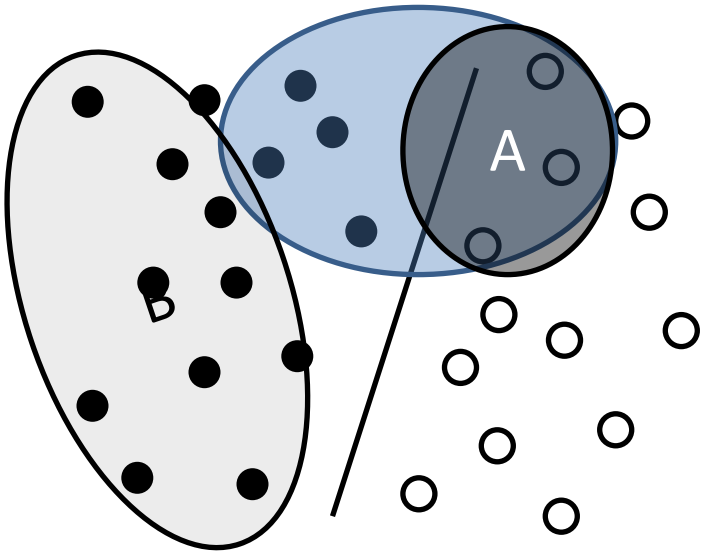
Fung et al. 2002, Towell and Shavlik 1994, Kunapuli et al. 2013

Types of Advice

Monotonicity

As feature x ↑,
P(positive) ↑



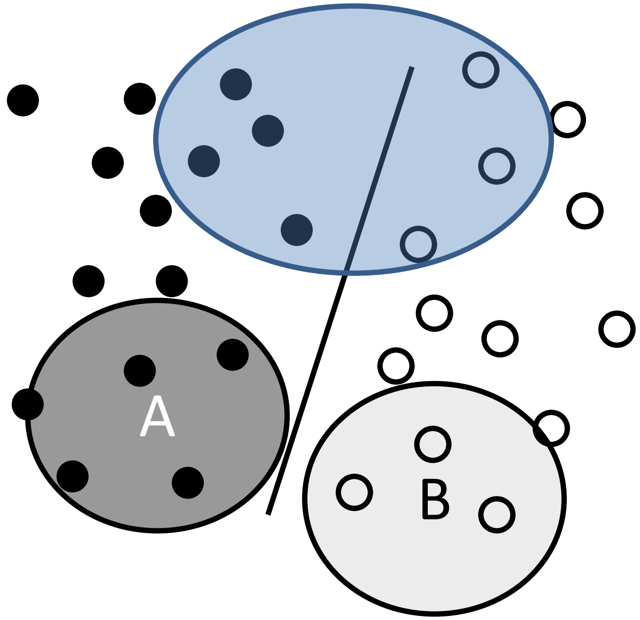


Precision/Recall Tradeoff

Yang & Natarajan ECML ‘13, Yang et al. ICDM ‘14

Types of Advice

Preference Knowledge



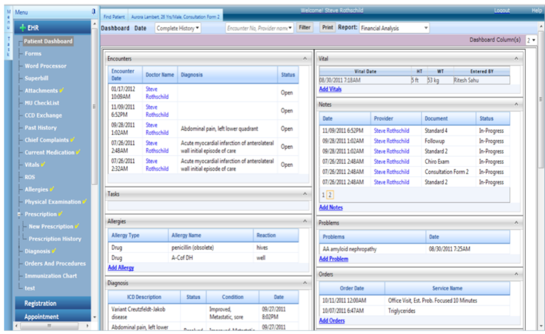
Powerful framework that can incorporate different kinds of advice

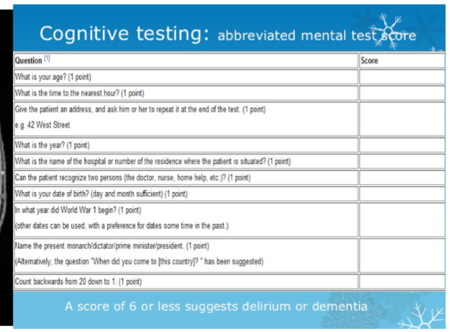
Odom et al. AAAI '15

Types of Advice

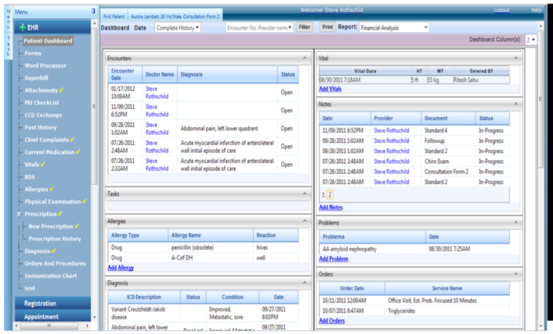
Privileged Information

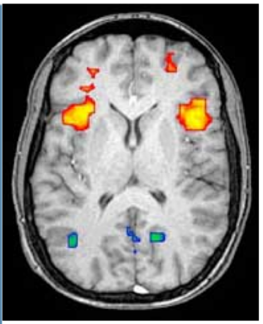
Training Phase



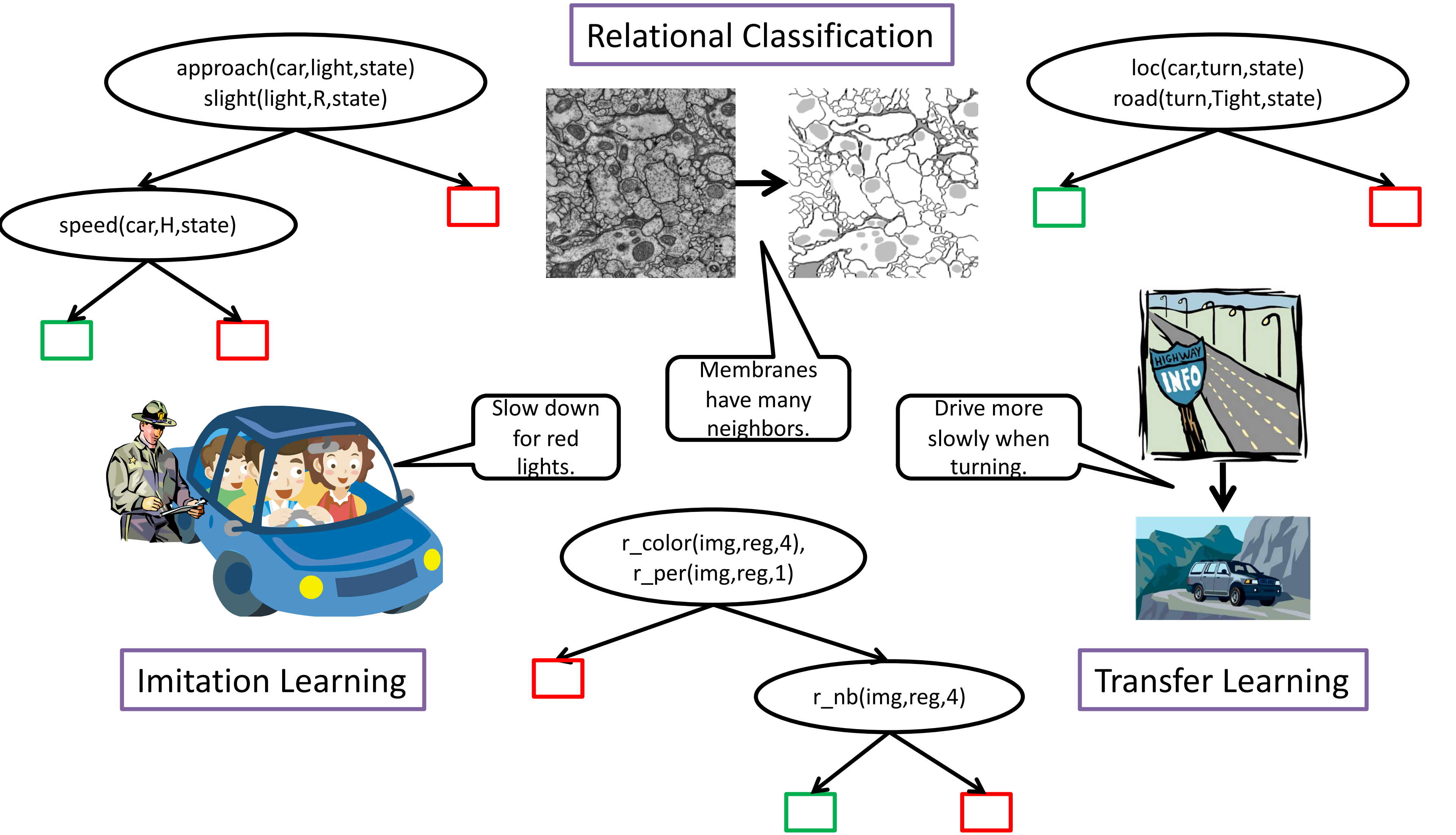


Deployment/ Test



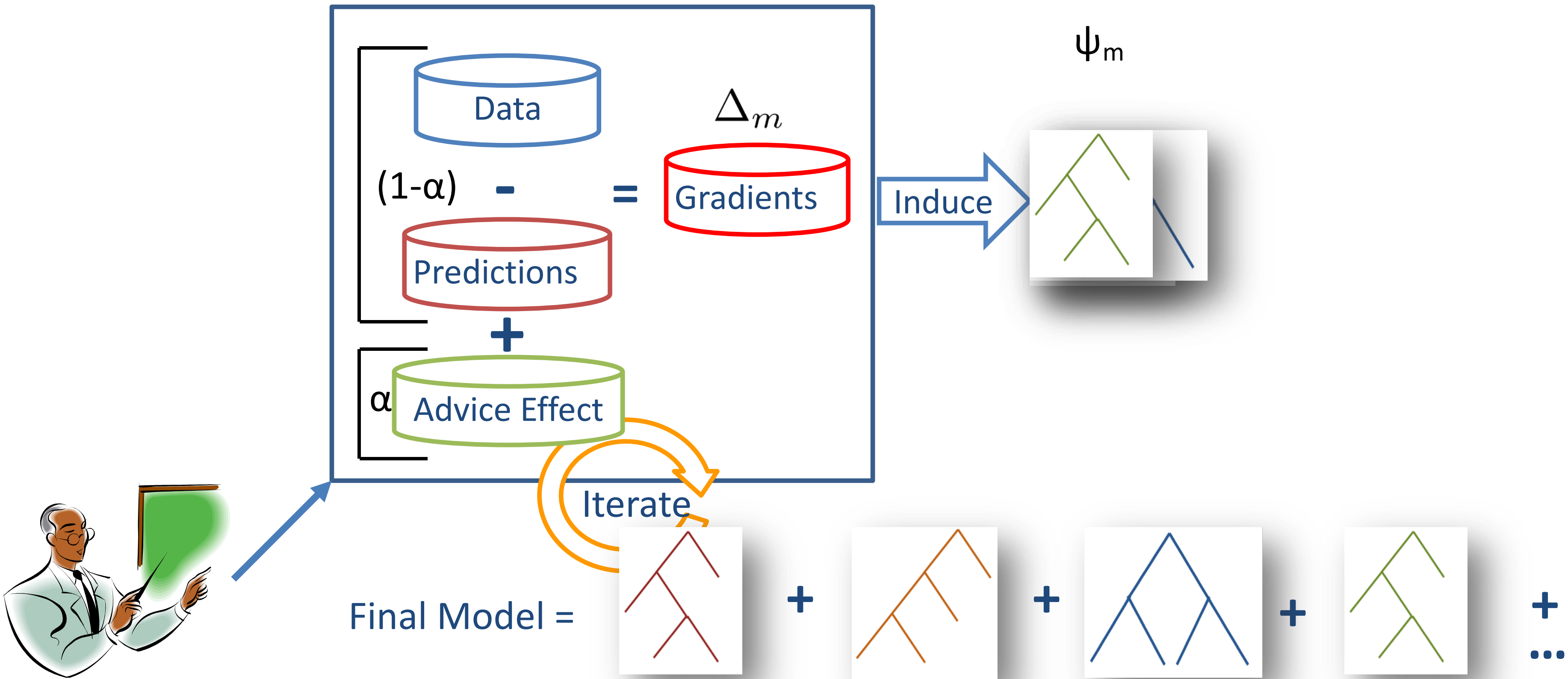


What is this Advice?



Advice for RFGB: Update Gradients

Idea: Incorporate advice into the gradients



Odom et al. AAAI '15, Yang et al. ICDM '14

Learning Framework

$$MLL(\boldsymbol{x}, \boldsymbol{y}) = \sum_{x_i \in \boldsymbol{x}} \log \frac{\exp(\varphi(x_i; y_i))}{\sum_{y'} \exp(\varphi(x_i; y'\ ) + cost(y_i, y', \varphi)}$$

Label Preferences:

$$cost(x_i, \varphi) = -\lambda \times \varphi(x_i) \times [n_t(x_i) - n_f(x_i)]$$

Advice $\uparrow P(label_i)$

Advice $\downarrow P(label_i)$

Gradients:

$$\Delta(x_i) = I(y_i = 1) - P(y_i = 1; \psi) + \lambda[n_t(x_i) - n_f(x_i)]$$

(Aside) Linear Program Formulation for Sequential Decision-Making

trajectory constraints

$$\max_{r,\xi_i,\zeta_j} -\|\boldsymbol{r}\|_1 + \boxed{\lambda_t} \sum_{i=1}^{n} \xi_i + \boxed{\lambda_a} \sum_{j \epsilon S_e} \zeta_j$$

weights on advice/trajectories

advice constraints

s.t.

$$\left(\boldsymbol{P}_{a*}^{i} - \boldsymbol{P}_{a}^{i}\right)(I - \lambda \boldsymbol{P}_{a*})^{-1}\boldsymbol{r} \geq \xi_i, \qquad \forall a \in A \backslash a_*, i = 1, \ldots, n$$

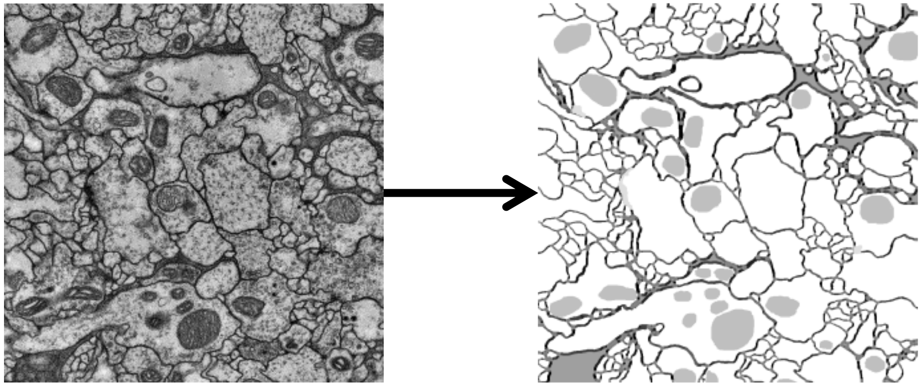
$$\left(\boldsymbol{P}_{a*}^{i} - \boldsymbol{P}_{a}^{i}\right)(I - \lambda \boldsymbol{P}_{a*})^{-1}\boldsymbol{r} \geq \zeta_j + \delta_j, \forall a \in Pref_j, a' \in Avoid_j, j \in S_e$$

$$\zeta_j \geq 0, \qquad \forall j \in S_e$$

$$|r_i| \leq r_{max}, \qquad \forall i = 1, \ldots, n$$

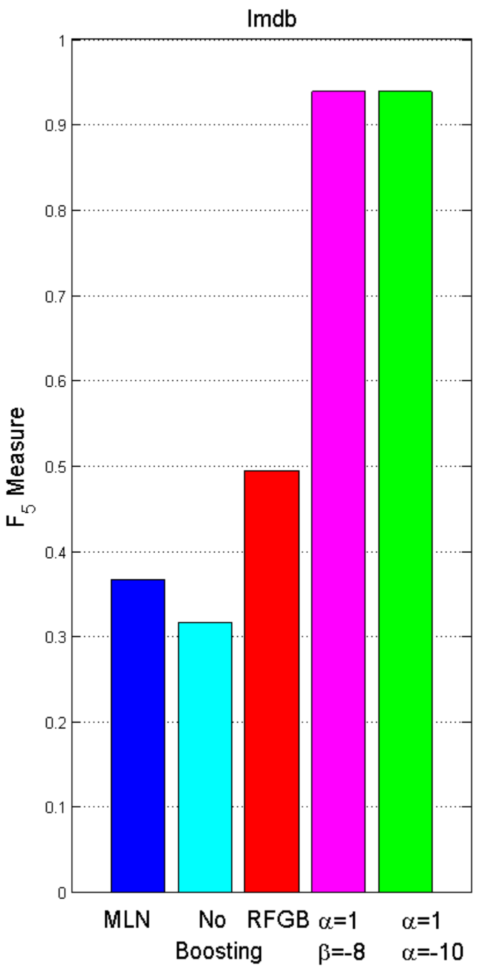
Kunapuli et al. (2013)

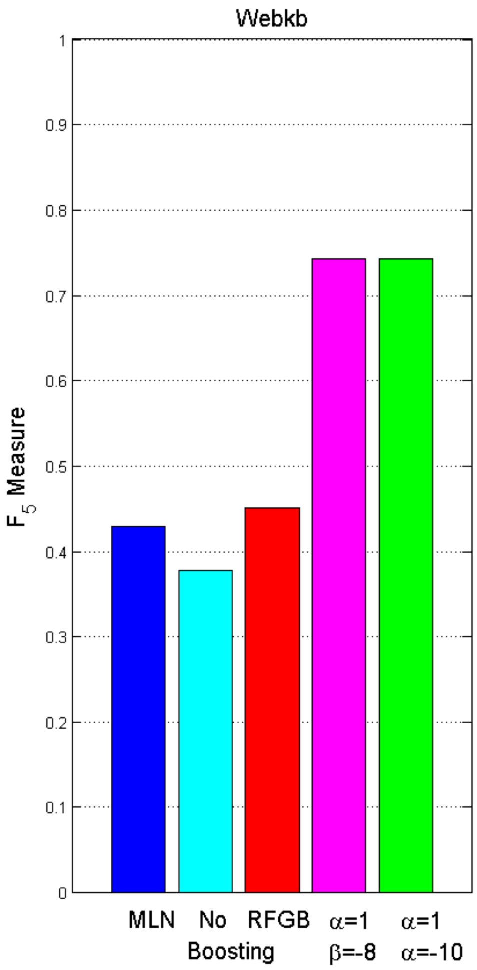
Sample Results
Relational Classification



Model	Accuracy
Propositional	68.6%
Relational	69.3%
Adv-Initial (Prop)	72.2%
Adv-Initial (Rel)	91.5%
Adv-Relational	**99.1%**

Sample Results
Cost-sensitive Learning

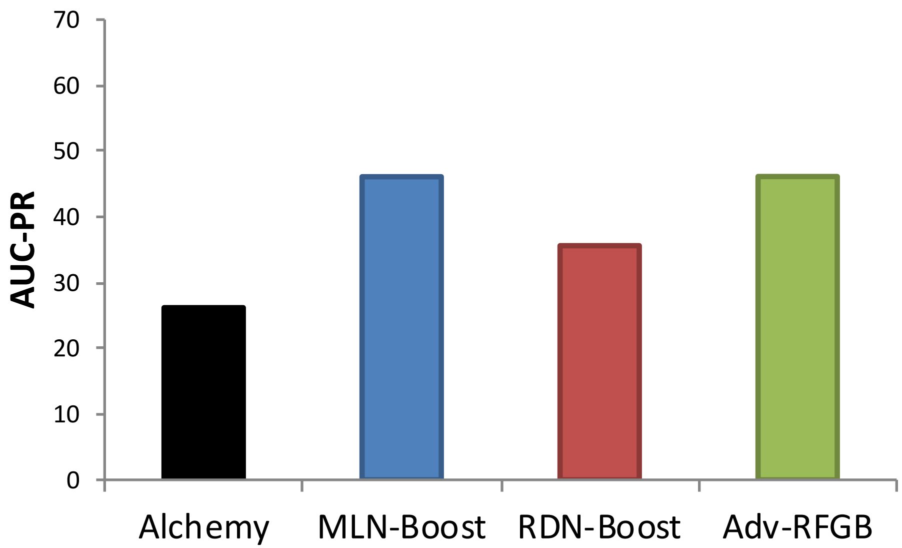


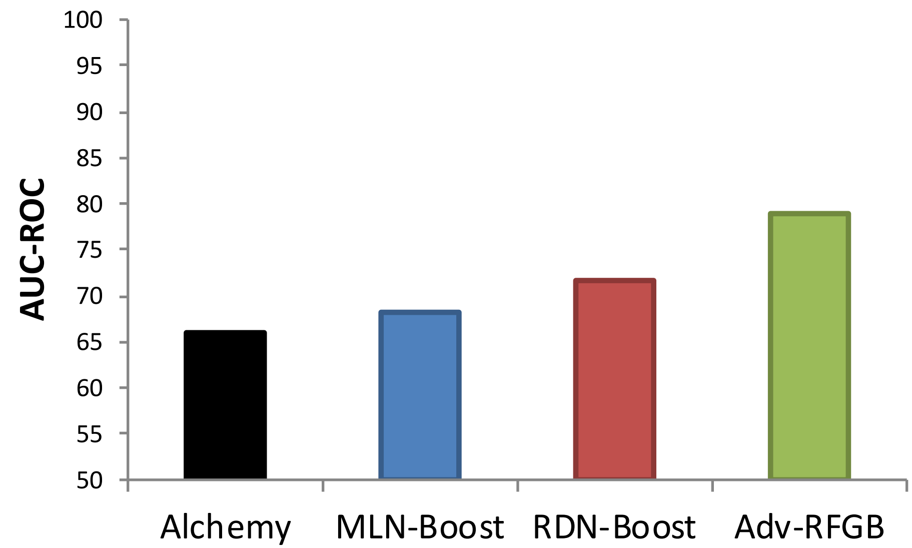


Advice for Adverse Drug Events NLP

- “If a drug and an effect are present in a proposed ADE and a sentence contains both the drug and effect, the ADE is true”
- “If a drug and an effect are present in a proposed ADE, and a sentence contains both the drug and effect, and the sentence contains the pattern ‘effect after drug’, the ADE is true”
- “If a drug and an effect are present in a proposed ADE, and a sentence contains both the drug and effect, and the sentence contains the pattern ‘drug-induced effect’, the ADE is true”

Experiments: Adverse Drug Events

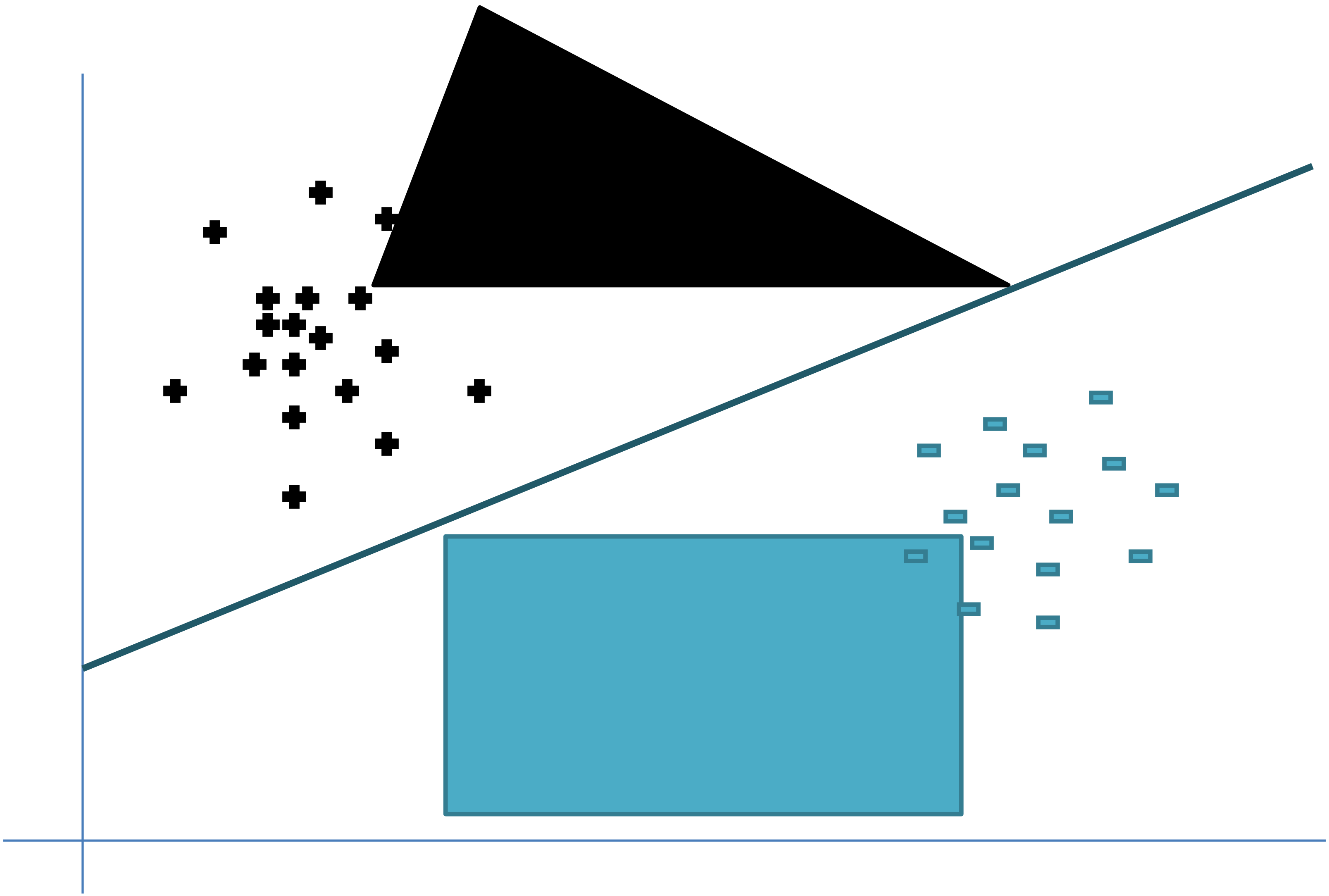




Outline

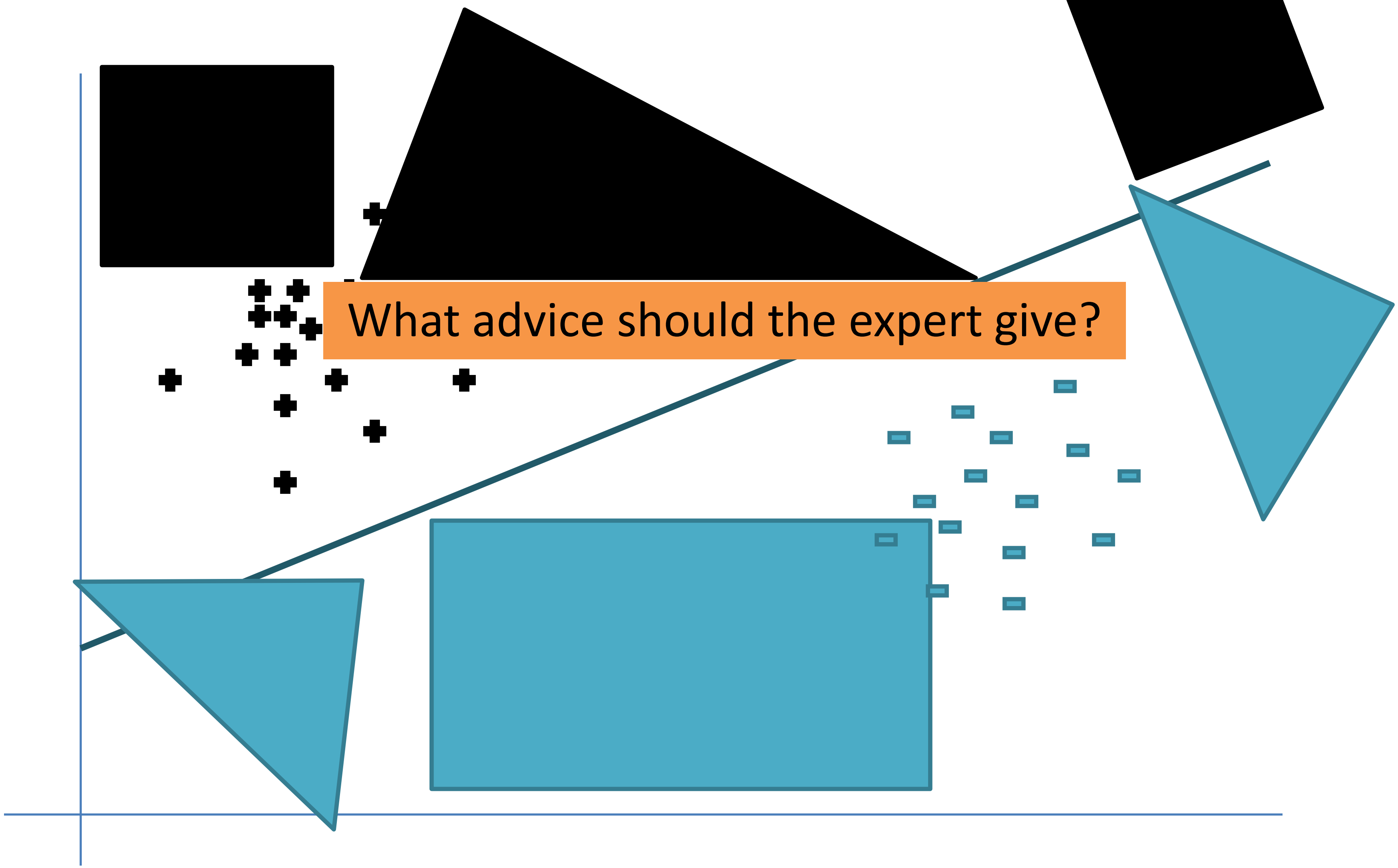
- State-of-the-art Structure Learning
- Effective Learning
- Advice Taking in SRL
- Actively Seeking Advice

Knowledge-Based Learning

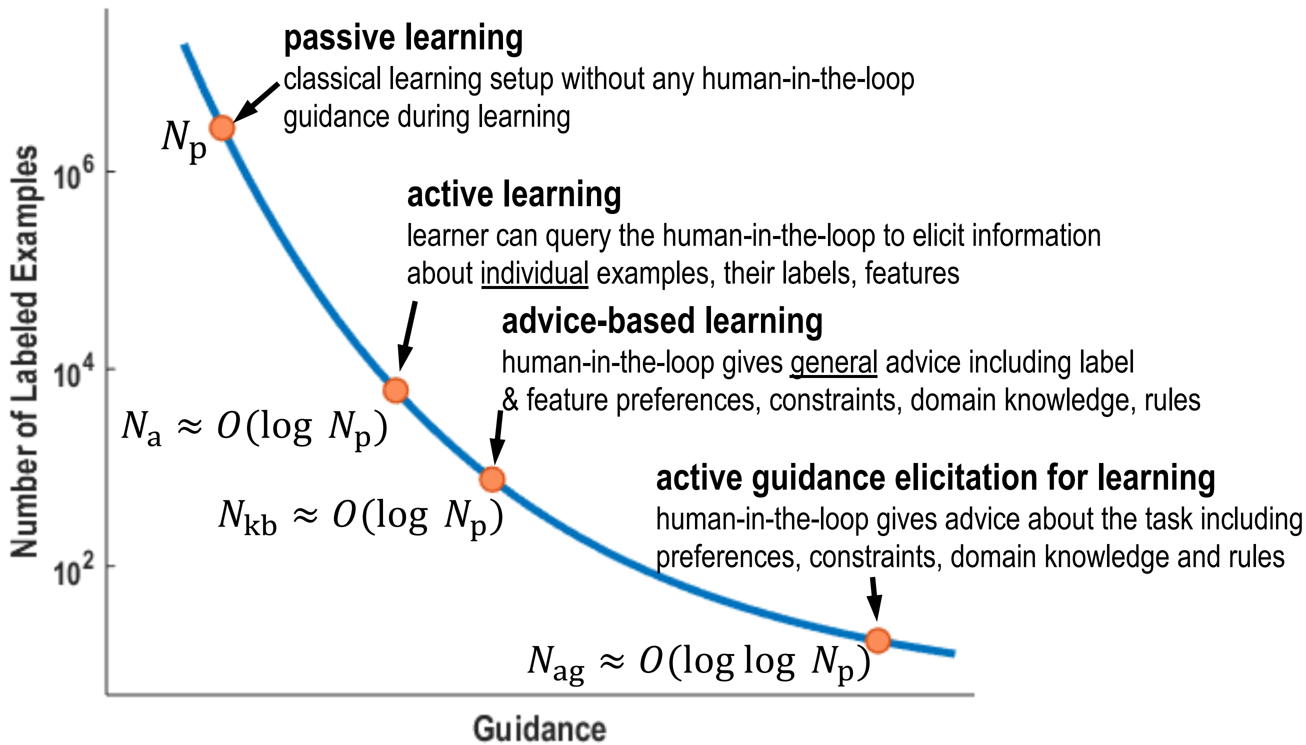


Fung et al. 2002

Knowledge-Based Learning



Active vs Passive Learning



Active Learning

- Learn initial model from training data - m_i
- Generate prediction over data - $P_{m_i}(y_i|x_i)$
- Calculate uncertainty – $H(P_{m_i}(y_i|x_i))$
- Select example(s) -$\underset{x_i}{\operatorname{argmax}} H(P_{m_i}(y_i|x_i))$

Relational Active Advice Seeking

Relational Model

- Learn initial model from training data -
- Generate prediction over data - $P_{m_i}(y_i|x_i)$
- Calculate uncertainty – $H(P_{m_i}(y_i|x_i))$
- Select ~~example~~(s) -argmax $H(P_{m_i}(y_i|x_i))$

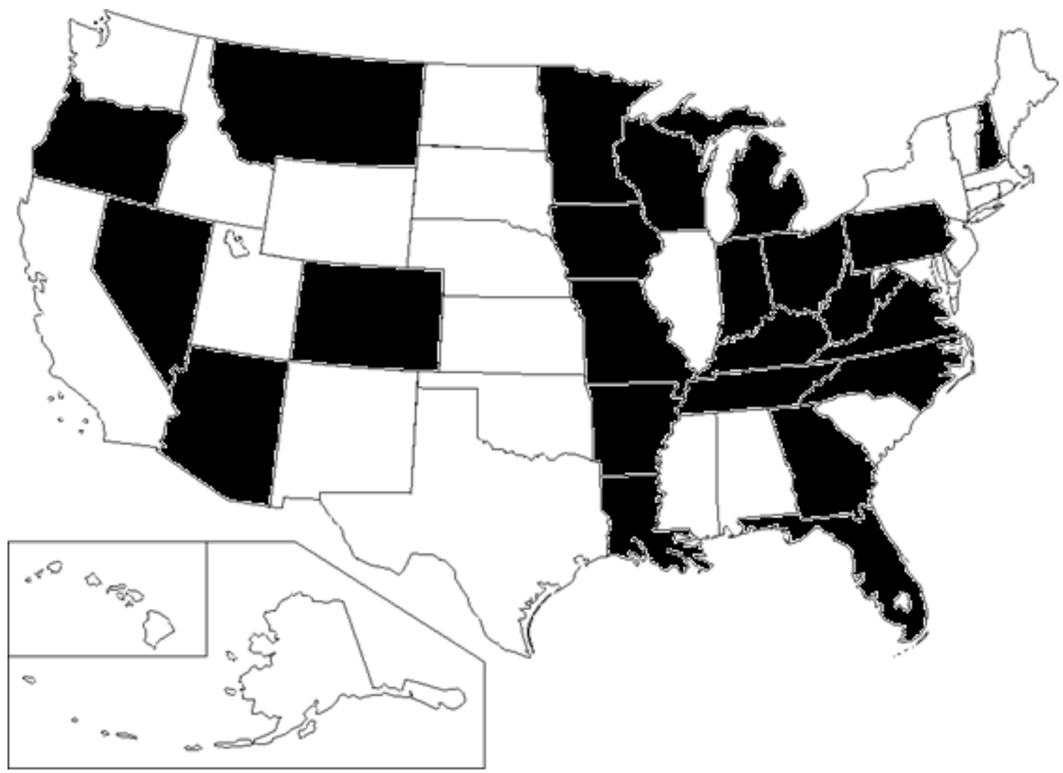
Select **advice clause** with the highest uncertainty

Odom & Natarajan (2016)

Relational Query Generation

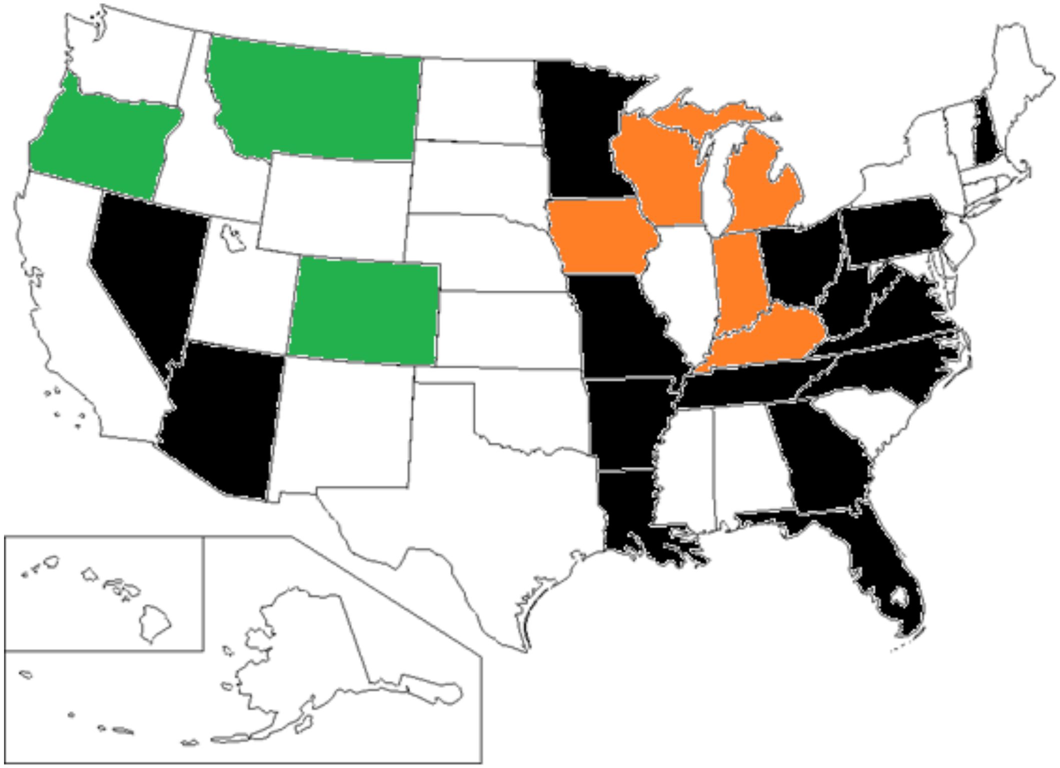
Predict which states vote democrat?

Relational Query Generation



Predict which states vote democrat?

Relational Query Generation



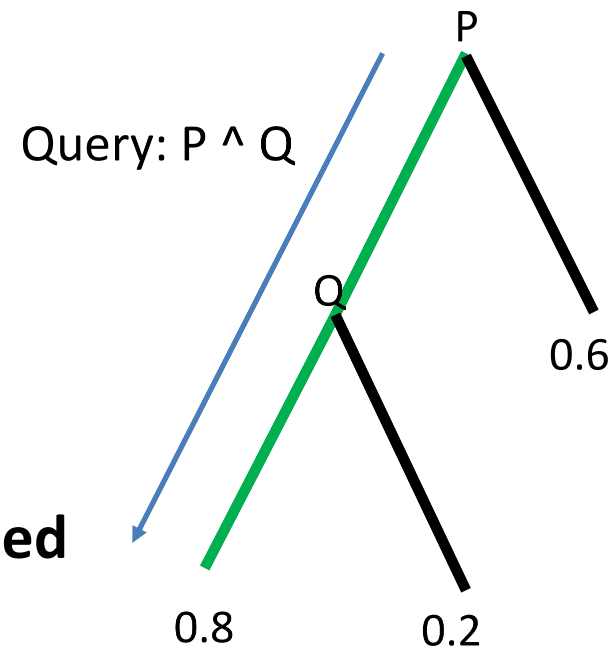
Queries:

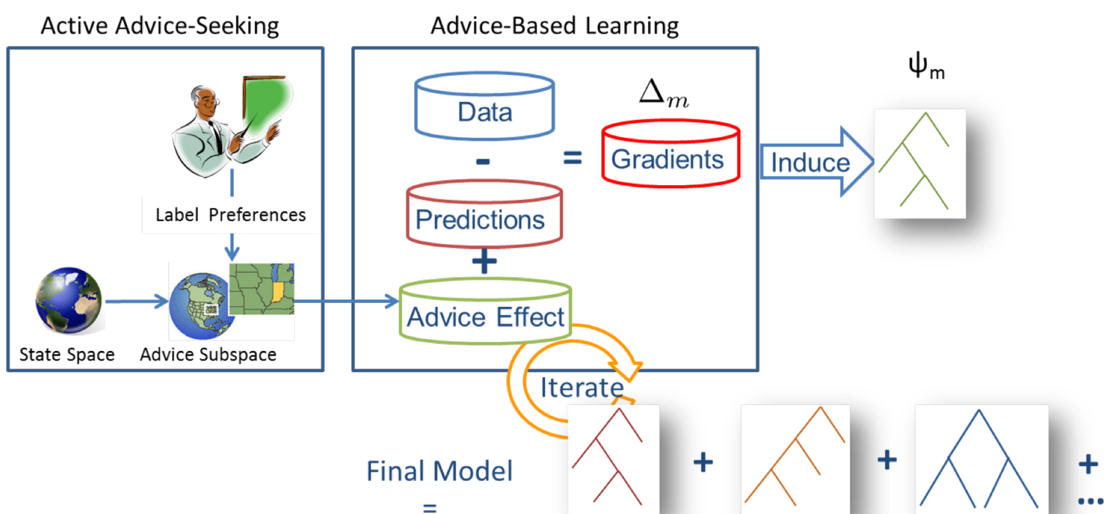
$Region(x, West), Gov(x, y), Party(y, Dem)$

PrimaryIndustry(x, Manufacturing)

Relational Active Advice Seeking

- Query – Conjunction of literals
  - Defines set of examples
  - Learned by fitting regression trees on current uncertainty
- Expert Response – Selects **Preferred/Avoided** labels





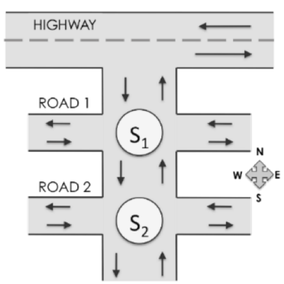
(Aside) Sequential Decision-Making

Goal: Select states ($\boldsymbol{c}_j$) according to their cluster quality (f) and the total uncertainty of all states ($S = \boldsymbol{c}_j \cup \boldsymbol{c}_k$)

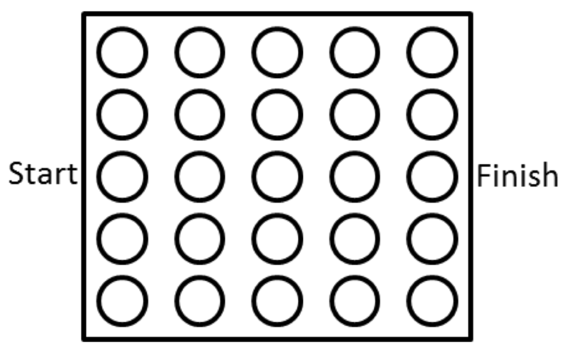
$$\arg\max_{\boldsymbol{c}_j} f\left(\left|\boldsymbol{c}_j\right|\right)\left[U_c\left(\boldsymbol{c}_j\right) + U_c(\boldsymbol{c}_k)\right]$$

We simplify this intractable global optimization problem by:

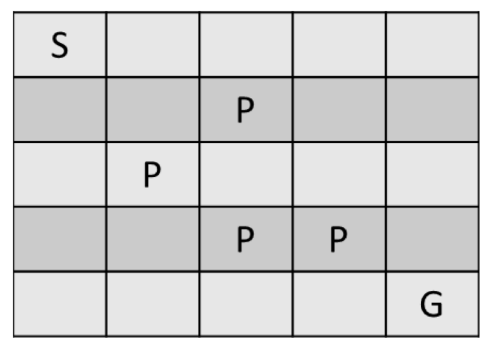
1. Clustering the states to define a finite set of queries
2. Considering an uniform cluster quality when selecting a query



Traffic Routing



Sailing

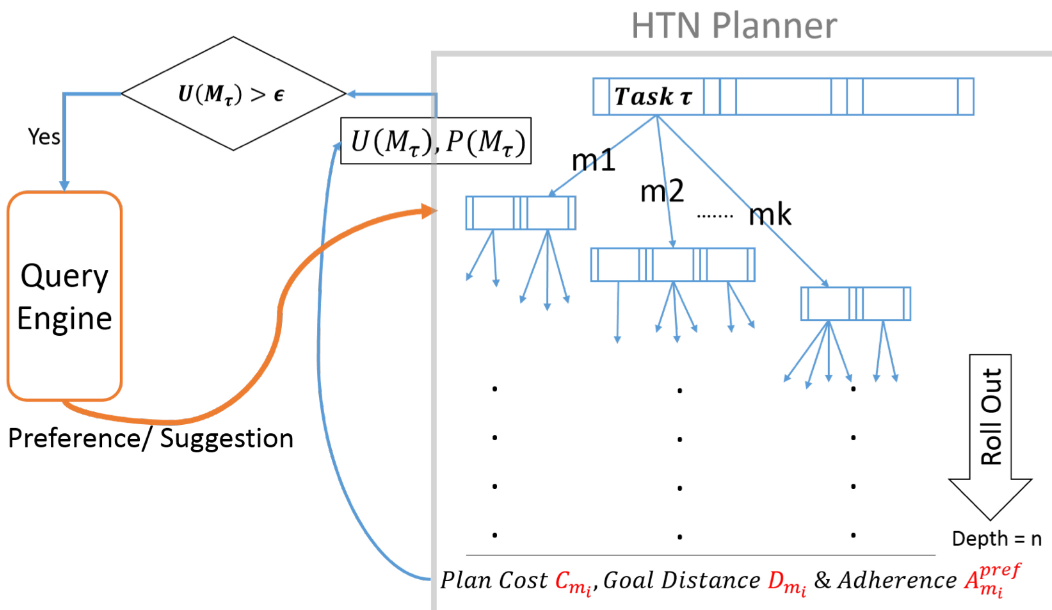


WumpusWorld

Drone Flying

Odom & Natarajan (2016)

(Aside) Preference Guided Planning



- Guides search progressively
- n-step Roll-out
- Evaluate each ***option*** $\mathrm{m_i}$: $S(m_i)$ *where* $m_i \in M_\tau$
- $S(m_i) = Func\left(C_{m_i}, D_{m_i}, A_{m_i}^{(pref)}\right)$
- $S(M_\tau) \rightarrow P(M_\tau)$ (induce distribution)
- Uncertainty $U(M_\tau) = -\sum_{m \in M_\tau} p(m) \log(\frac{1}{p(m)})$
- Query human based on uncertainty in decision

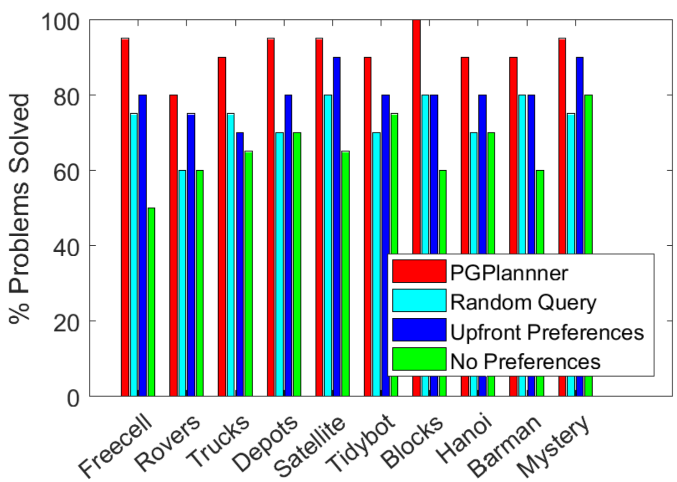
HMCL '17 Workshop at AAAI, AAMAS 18, KBS (under review)

Joint work with Jana Doppa, Rakibul Islam and Dan Roth

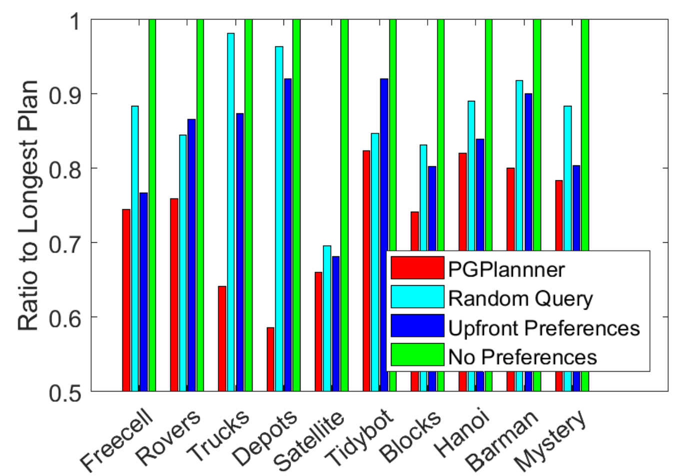
(Aside) HTN Planning Results

10 Domains ; Baselines : (1) Pre-Encoded/Upfront Preferences, (2) Random Queries & (3) No Preferences

AAMAS 2018



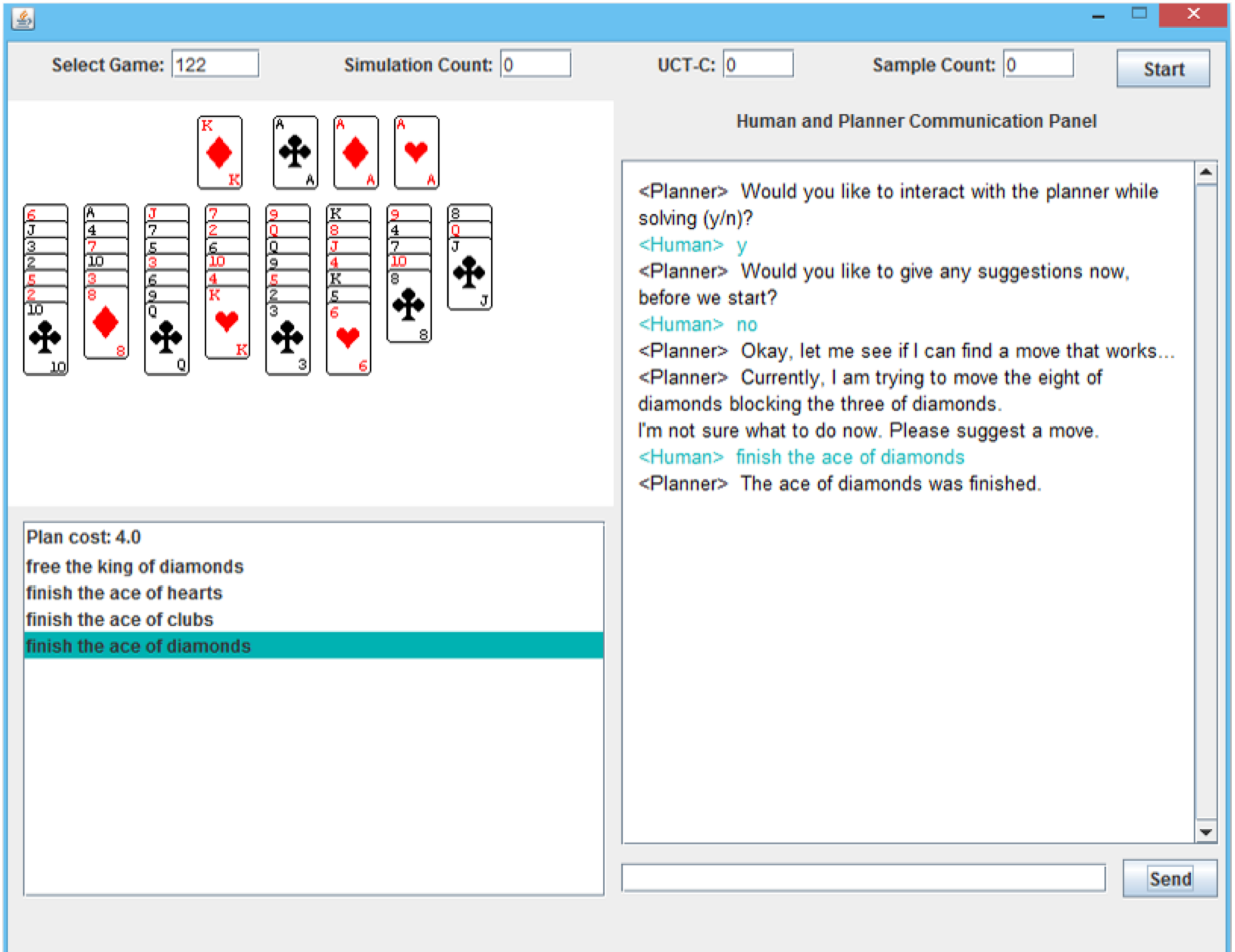
More Problems solved



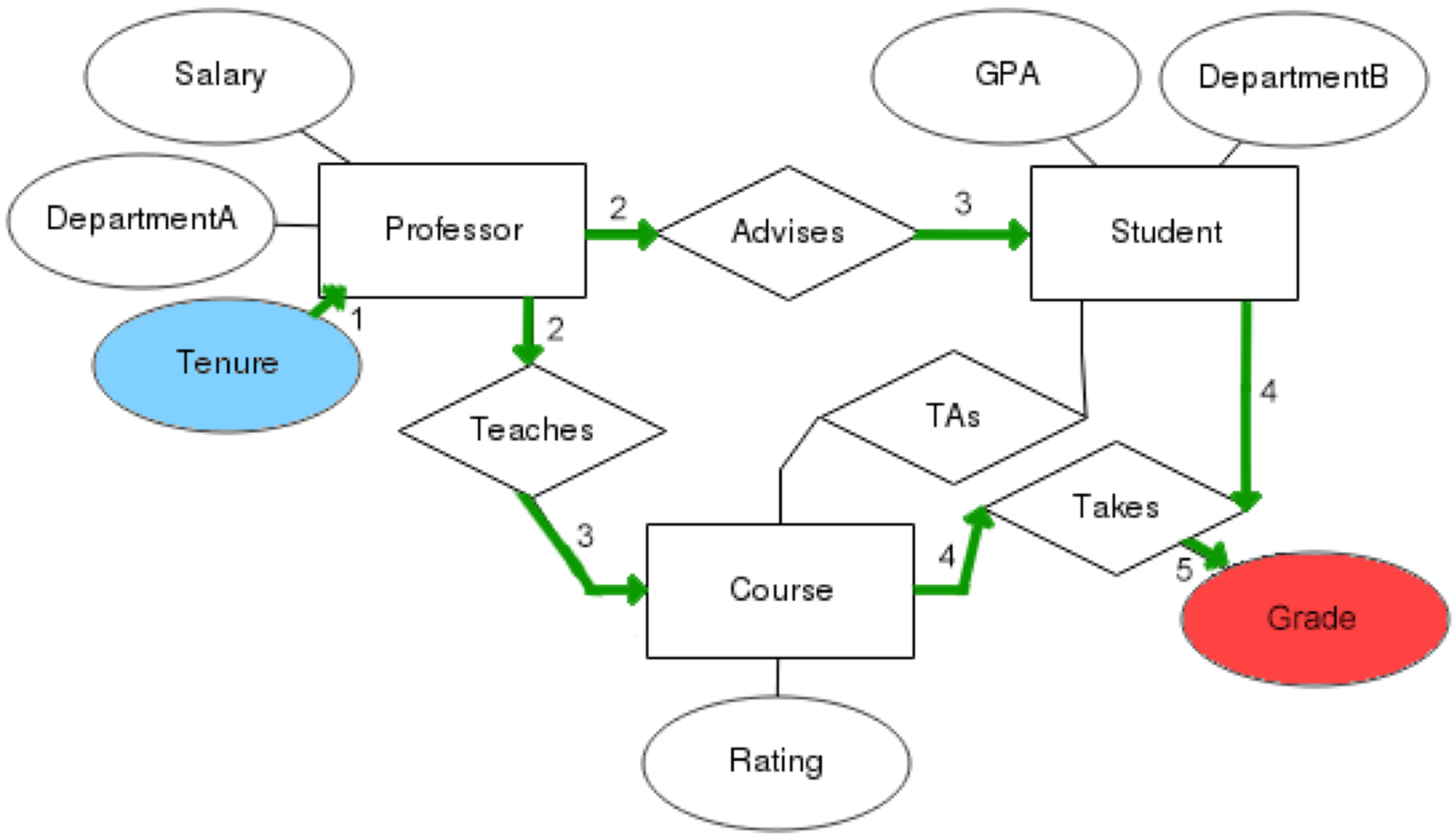
Better solutions obtained

(Aside) Interfaces for Communication

CwC – Joint work with Dan Roth, Jana Doppa, Julia Hockenmaier



ER Diagrams for Preference Specification?
(KCAP 2017)



Outline

- Effective Learning
- Advice Taking in SRL
- Actively Seeking Advice
- Wrap-Up

Precision Health Tasks

CARDIA EXAM COMPONENTS—ALL YEARS

Schedule of components in the core study, substudies, and ancillary studies by CARDIA exam

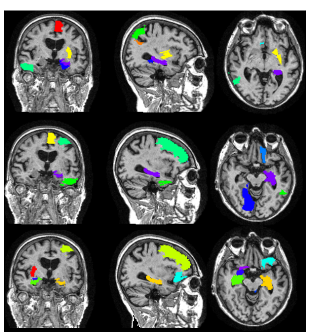
	Year/Exam[1]							
	1985	1987	1990	1992	1995	2000	2005	2010
	0	2	5	7	10	15	20	25
CORE STUDY								
BLOOD PRESSURE								
Resting	X	X	X	X	X	X	X	X
Standing	-	X	-	-	-	-	-	-
Reactivity	-	X	-	-	-	-	-	-
CHEMISTRIES								
Genetic								
DNA Storage	-	-	X	-	X	X	X	X
Stored Cells for Cell Immortalization	-	-	-	-	-	X	-	-
Plasma								
Lipids	X	X	X	X	X	X	X	X
Lipoproteins	X	X	X	X	X	X	X	-
Apoproteins	X	X	-	-	-	-	-	-
CBC	X	-	-	-	-	-	-	-
Lp(a)	-	-	X	-	-	-	-	-
Fibrinogen	-	-	X	-	-	-	X	-
ApoE Phenotype	-	-	-	X	-	-	-	-
Stored Plasma	-	X	X	X	X	X	X	X
C-Reactive Protein	-	-	-	X	-	X	X	X
Interleukin-6	-	-	-	-	-	-	X	-
Serum								
Cotinine	X	-	-	-		-	-	-
SMAC 12	X	-	-	-	-	-	-	-
Fasting Insulin	X	-	-	X	X	X	X	X
Fasting Glucose	X	-	-	X	X	X	X	X
Oral Glucose Tolerance Test	-	-	-	-	X	-	X	X
Stored Serum	X	X	X	X	X	X	X	X
GGT	X	-	-	-	X	-	-	-
Serum Creatinine	X	-	-	-	X	X	X	X
Uric Acid	X	-	-	-	X	X	-	-
Urine								
Urinary Creatinine	-	-	-	-	X	X	X	X
Albuminuria	-	-	-	-	X	X	X	X
ANTHROPOMETRY								
Height	X	X	X	X	X	X	X	X
Weight	X	X	X	X	X	X	X	X
Skinfolds	X	X	X	X	X	-	-	-
Chest Circumference	-	X	X	-	-	-	-	-
Waist Circumference	X	X	X	X	X	X	X	X
Hip Circumference	X	X	X	X	X	-	-	X
Thigh Circumference	-	-	-	X	-	-	-	-
Elbow Breadth	X	X	-	-	-	-	-	-
Shoulder Breadth	-	X	-	-	-	-	-	-
Sitting Height	-	X	-	-	-	-	-	-
Toenails	-	X	-	-	-	-	-	-
Eye Color	-	-	X	-	-	-	-	-
Skin Reflectance	-	-	-	X	-	-	-	-
MEDICAL HISTORY								
Medical History	X	X	X	X	X	X	X	X
Illicit Drug Use	X	X	X	X	X	X	X	X
Death Certificate	-	X	X	X	X	X	X	X
Mortal Events	-	-	-	X	X	X	X	X
Safety Questionnaire	-	X	-	-	-	-	-	-
Interim Hospitalization	-	X	X	X	X	X	X	X
Chest Pain/Palpitations	-	-	X	-	X	-	-	-
History of Lung Problems	X	X	-	-	X	X	X	-
Oral Contraceptive History	-	-	-	-	X	-	-	-
Women's Reproductive Health	-	-	-	-	-	X	X	X
Sleep Habits	-	-	-	-	-	X	X	-
Tobacco	X	X	X	X	X	X	X	X
Alcohol	X	X	X	X	X	X	X	X
Weight History	X	X	-	-	-	-	-	X
Sociodemographics	X	X	X	X	X	X	X	X
FAMILY HISTORY QUESTIONNAIRE								
Family History	X	-	X	-	X	-	-	X
PHYSICAL ACTIVITY/FITNESS								
7-Day Physical Activity	X	-	-	-	-	-	-	-
Physical Activity Questionnaire	X	X	X	X	X	X	X	X
Graded Exercise Test	X	-	-	X	-	-	-	-
Baecke Questionnaire	-	-	X	-	X	-	-	-
Household Chores	-	-	-	X	X	-	-	-
Sedentary Behavior Questionnaire	-	-	-	-	-	-	-	X

[1] year of study indicates when original data collection occurred; assay or coding may occur later

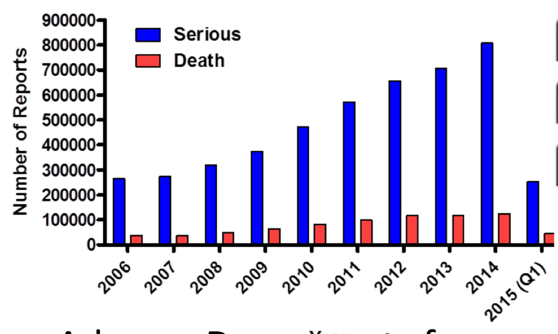
Cardiovascular events/procedures from EHR/Clinical Study (IAAI 12, IAAI13, AI Magazine 12, AIME 15a, AAAI 15, AAAI 16, AIME 15b, AIME 17a, AIME 17b)

Theme	Questions	Answers
Demographic Information	Age	Number entry
	Gender	Male, Female, Other
	Country of Residence	Text entry
	Marital Status	{Married, Living with a Partner, Divorced, Separated, Widowed, Single, Other}
	Employment	{Full time, Part time, Retired, Student, Disabled, Not Employed for Pay}
	Education	{Less than grade 8, some high school, completed high school, technical/trade/vocational school, some college/university, completed college/university, some post-graduate education, completed post-graduate education}
Disease Information	Disease name	Text entry
	How many years has it been since you first started experiencing symptoms?	Number entry
	How many years has it been since you were diagnosed?	Number entry
	How severely do your symptoms impact your life?	5 point scale from No impact to Extreme impact
Technology Use	How often do you use the internet?	{Several times a day, About once a day, Several times a week, Every few weeks, Less often, Never}
	Do you own any of the following technologies? (Check all that apply)	{Desktop computer, Laptop computer, Cell phone, e-Reader, MP3 Player, Game console, Tablet}
	On your cell phone, do you have any applications that help you track or manage your health?	{Yes, No, I dont have a cellphone}
	Do you ever use your cell phone to look up health or medical information?	{Yes, No, I dont have a cellphone}

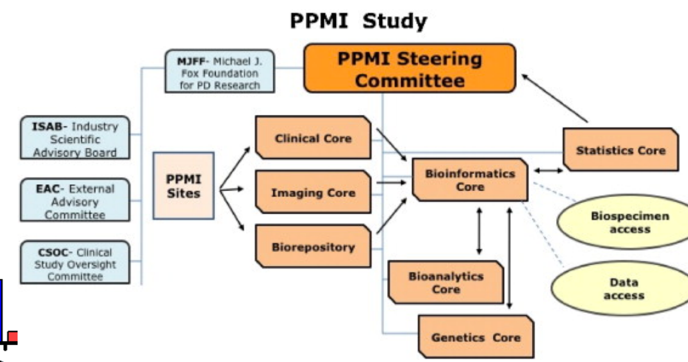
Predicting rare diseases/PPD from survey data (CHASE 16, CHASE 17)



Alzheimer's /diabetes prediction from FMRI (IJMLC 13, Neuroradiology, ICMLA 12Jour of Neurotrauma 15)



Adverse Drug events from EHR/Med abstracts AAAI 12, AIME 15, EMNLP (under preparation)



Parkinsons' prediction from Study data (AIME 17)

Conclusions

- AI in the wild is more than a single table - Graphs, different data types, relational DBs, … are central to HAAI
- Symmetry/Relation aware AI is essential
- Sequential models and dynamic models that do not make simplistic assumptions are necessary
- Human is an ally in learning and AI system needs to efficiently use human knowledge and input
- Deployment: medicine, social science, traffic, journalism, … and the Data Science Genome: Machines read and understand data science publications and help the user with their problem at hand

Next Steps

- Web-scale StaRAI
- More applications
  - NELL, relation extraction, more EHRs, imaging data
- Learning from Multiple Experts
- Efficient Inference for CTBNs
- “Truly” Hybrid Models
- Combine with efficient inference techniques
  - Approximate counting & Approximate inference
- Solve a *dream* problem
  - Relational POMDPs