Entity-Oriented Data Science

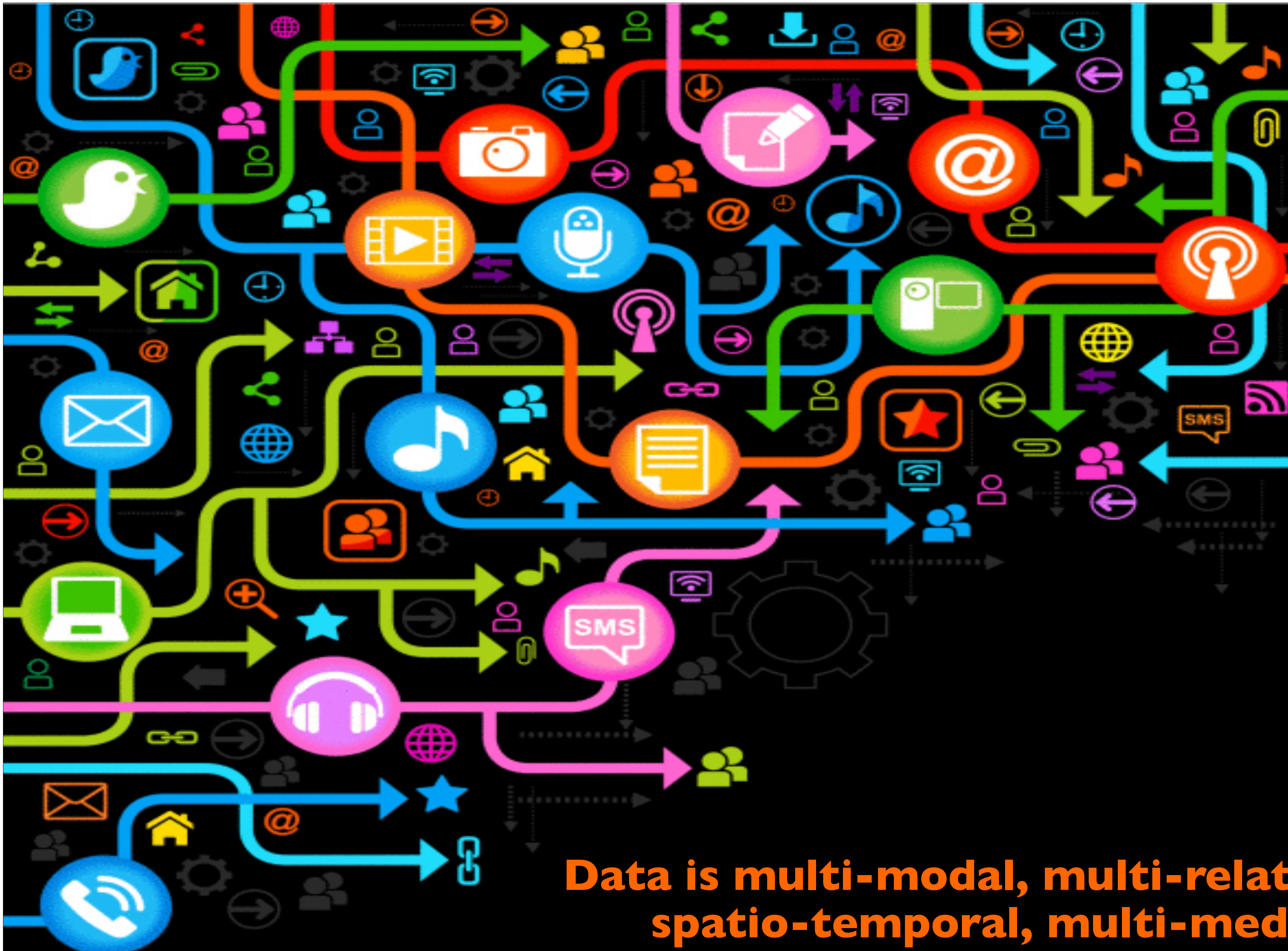
Prof. Lise Getoor

University of Maryland, College Park

http://www.cs.umd.edu/~getoor

September 5, 2013


BIG **Data is not flat**
©2004-2013 lonnitaylor



Data is multi-modal, multi-relational, spatio-temporal, multi-media

Entities and relationships are important!

NEED: Data Science for Graphs

Statistical Relational Learning (SRL)

- AI/DB representations + statistics for multi-relational data
  - Entities can be of different types
  - Entities can participate in a variety of relationships
  - examples: Markov logic networks, relational dependency networks, Bayesian logic programs, probabilistic relational models, many others…..

- Key ideas
  - Relational feature construction
  - Collective reasoning
  - ‘Lifted’ representation, inference and learning

- Related areas
  - structured prediction, hierarchical models, latent-variable relational models, multi-relational tensors, representation learning, …

For more details, see NIPS 2012 Tutorial, http://linqs.cs.umd.edu/projects//Tutorials/nips2012.pdf

Common Graph Data Analysis Patterns

- Joint inference over large networks for:
  - **Collective Classification**
  - **Link Prediction**
  - **Entity Resolution**

Common Graph Data Analysis Patterns

- Joint inference over large networks for:
  - **Collective Classification** – inferring labels of nodes in graph
  - **Link Prediction**
  - **Entity Resolution**

Common Graph Data Analysis Patterns

- Joint inference over large networks for:
  - **Collective Classification** – inferring labels of nodes in graph
  - **Link Prediction** – inferring the existence of edges in graph
  - **Entity Resolution**

Common Graph Data Analysis Patterns

- Joint inference over large networks for:
  - **Collective Classification** – inferring labels of nodes in graph
  - **Link Prediction** – inferring the existence of edges in graph
  - **Entity Resolution** – clustering nodes that refer to the same underlying entity

What's Needed Next?

- Methods which can perform and interleave these tasks
- Methods which support:
  - **Graph identification** – inferring a graph from noisy observations
  - **Graph alignment** - mapping components in one graph to another
  - **Graph summarization** - clustering the nodes and edges in a graph
- Desiderata: Flexible, scalable, declarative support for collective classification, link prediction, entity resolution and other information alignment and information fusion problems….

Probabilistic Soft Logic (PSL)


Stephen Bach


Matthias Broecheler


Alex Memory


Lily Mihalkova


Stanley Kok


Bert Huang


Angelika Kimmig

Probabilistic Soft Logic (PSL)

Declarative language based on logics to express collective probabilistic inference problems

- Predicate = relationship or property
- Atom = (continuous) random variable
- Rule = capture dependency or constraint
- Set = define aggregates

PSL Program = Rules + Input DB

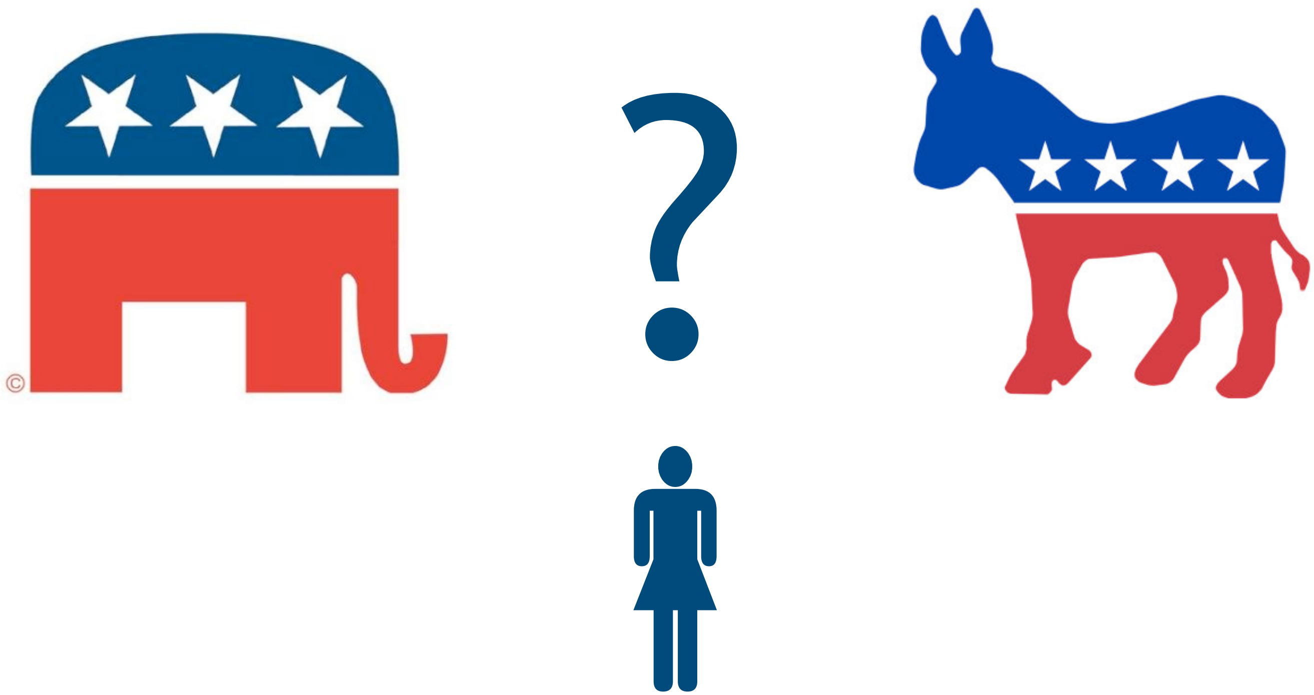
Probabilistic Soft Logic (PSL)

Declarative language based on logics to express collective probabilistic inference problems

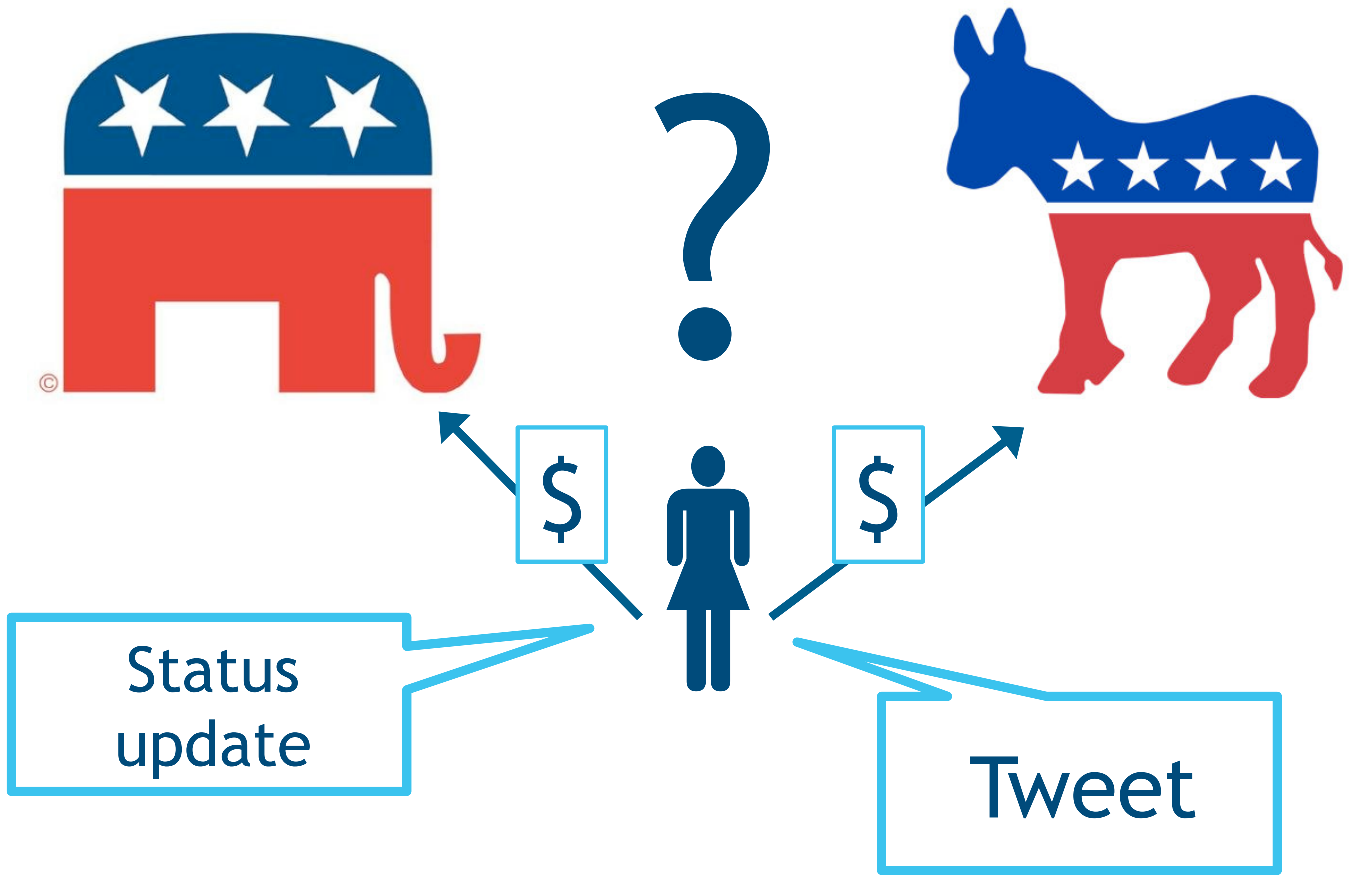
- Predicate = relationship or property
- Atom = (continuous) random variable
- Rule = capture dependency or constraint
- Set = define aggregates

PSL Program = Rules + Input DB

Node Labeling

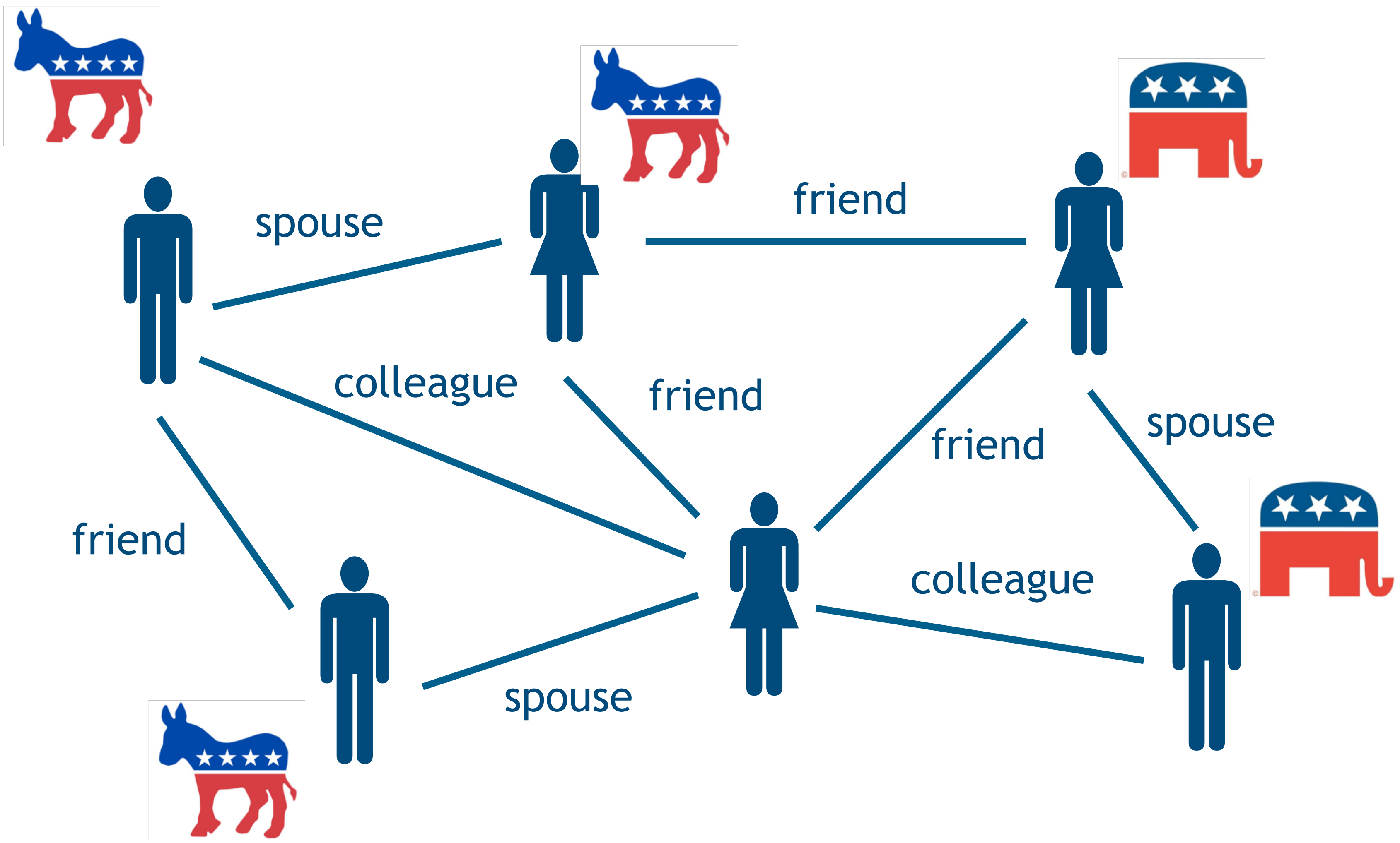


Voter Opinion Modeling



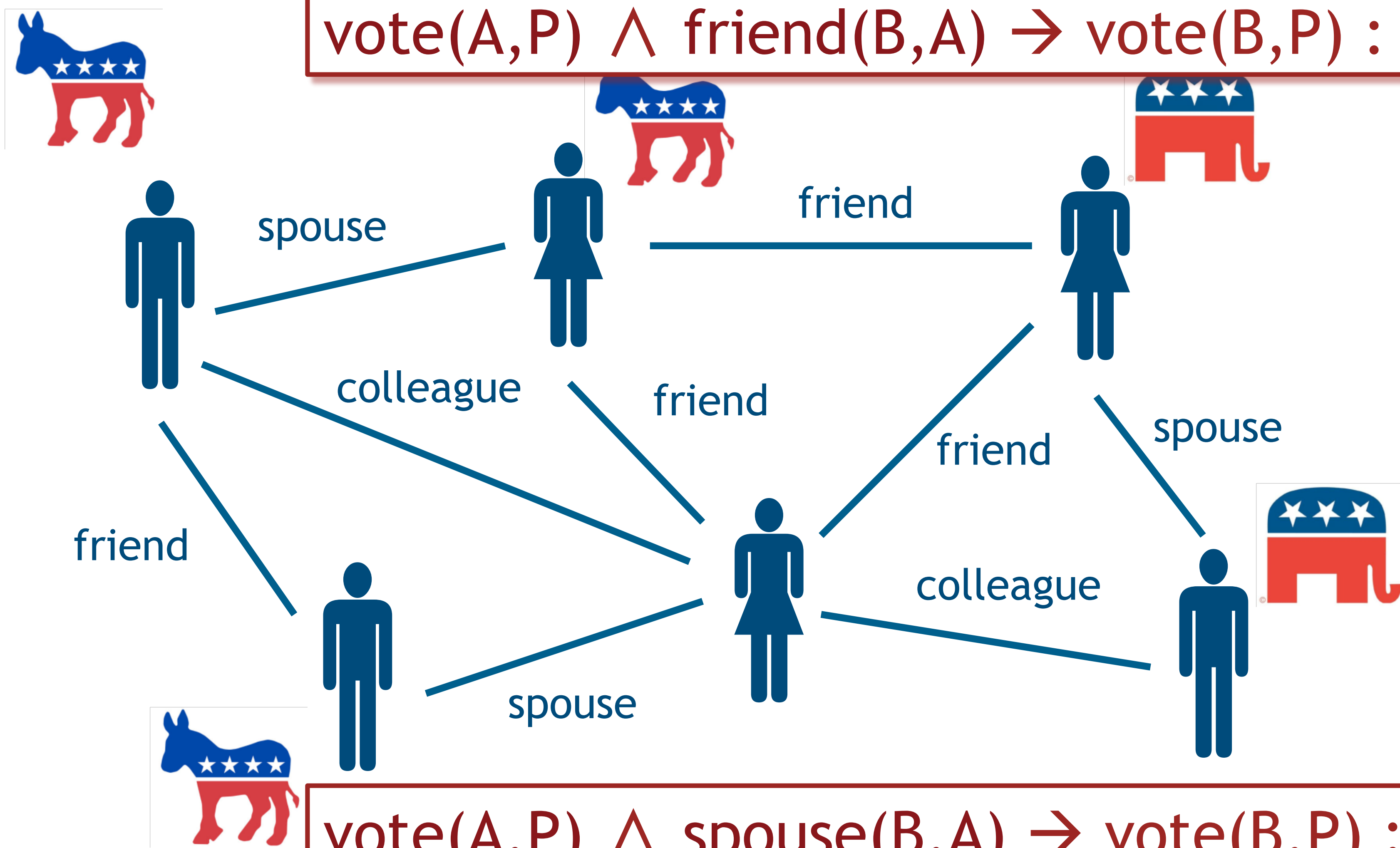
http://psl.umiacs.umd.edu

Voter Opinion Modeling



Voter Opinion Modeling

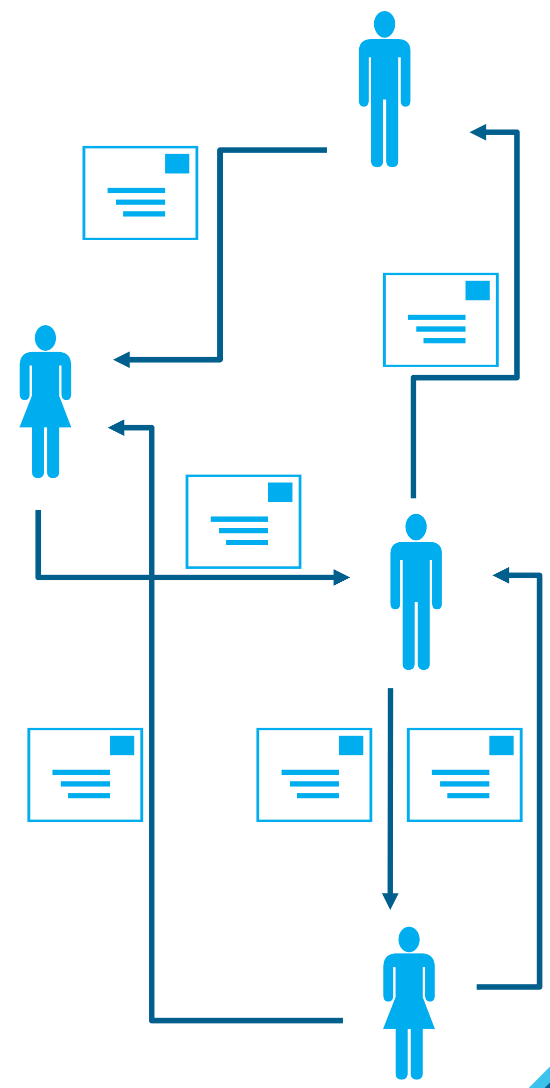
vote(A,P) ∧ friend(B,A) → vote(B,P) : 0.3



vote(A,P) ∧ spouse(B,A) → vote(B,P) : 0.8

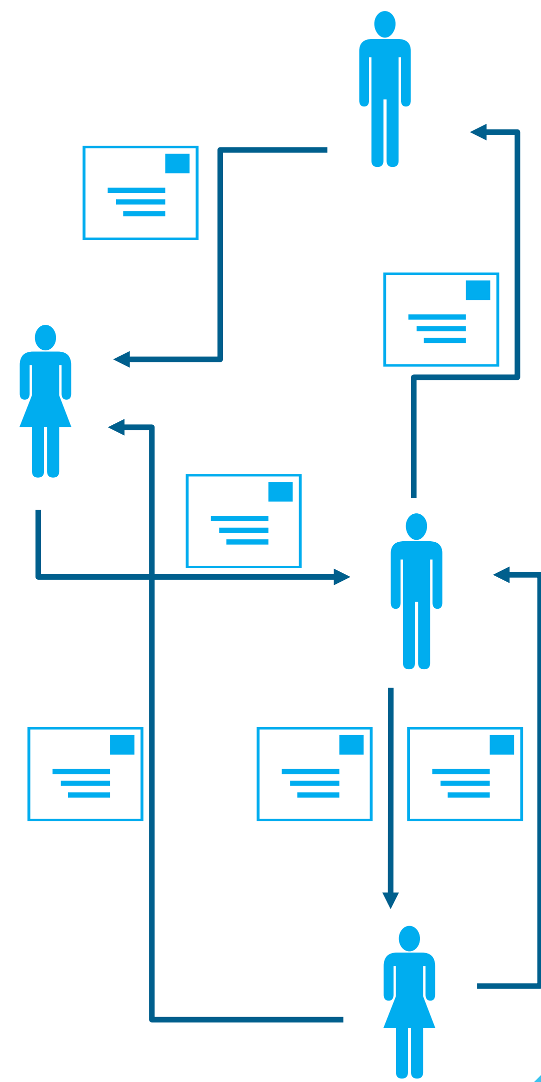
Link Prediction

- Entities
  - People, Emails
- Attributes
  - Words in emails
- Relationships
  - communication, work relationship
- Goal: Identify work relationships
  - Supervisor, subordinate, colleague



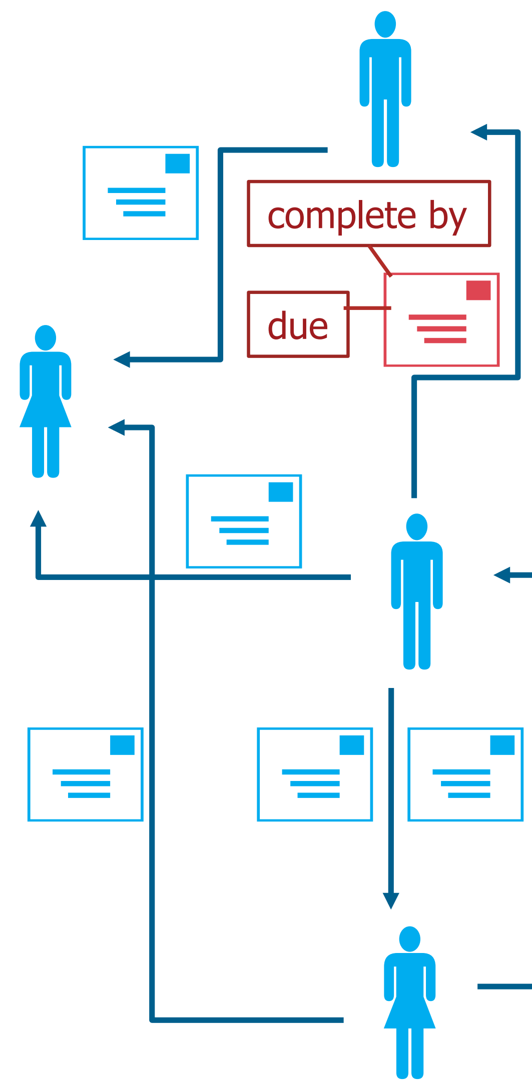
Link Prediction

- People, emails, words, communication, relations
- Use rules to express evidence
  - "If email content suggests type X, it is of type X"
  - "If A sends deadline emails to B, then A is the supervisor of B"
  - "If A is the supervisor of B, and A is the supervisor of C, then B and C are colleagues"



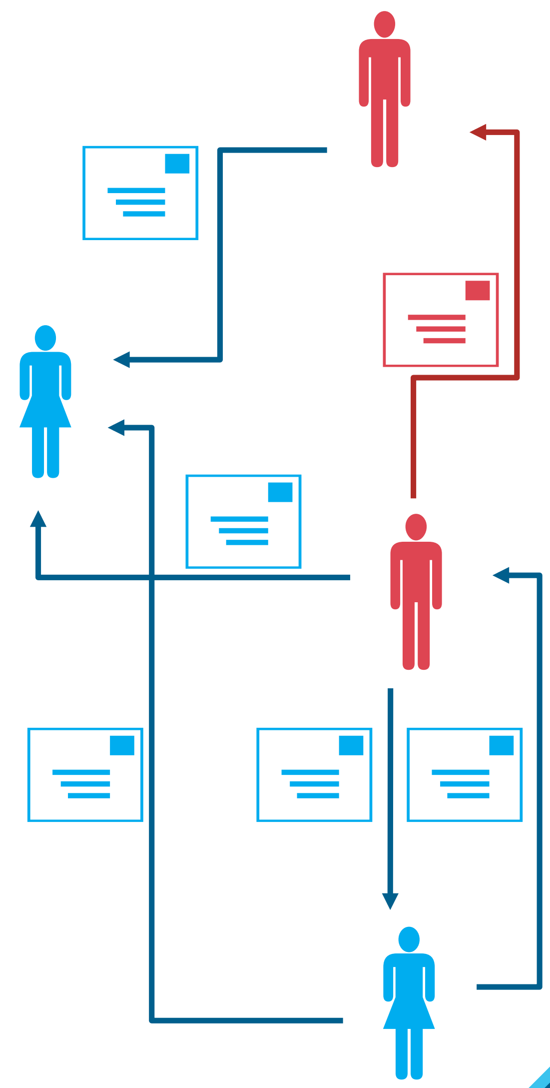
Link Prediction

- People, emails, words, communication, relations
- Use rules to express evidence
  - "If email content suggests type X, it is of type X"
  - "If A sends deadline emails to B, then A is the supervisor of B"
  - "If A is the supervisor of B, and A is the supervisor of C, then B and C are colleagues"



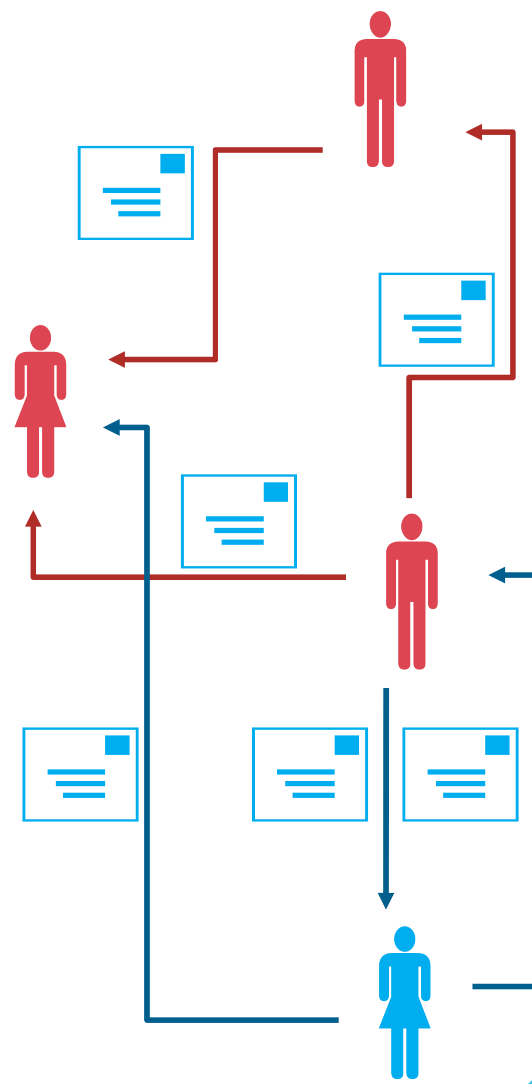
Link Prediction

- People, emails, words, communication, relations
- Use rules to express evidence
  - "If email content suggests type X, it is of type X"
  - "If A sends deadline emails to B, then A is the supervisor of B"
  - "If A is the supervisor of B, and A is the supervisor of C, then B and C are colleagues"



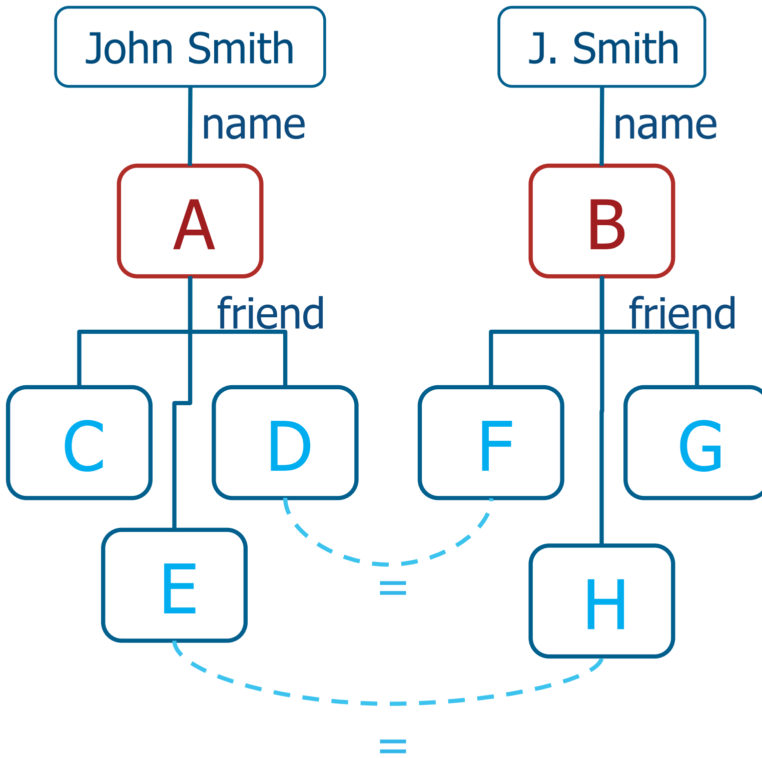
Link Prediction

- People, emails, words, communication, relations
- Use rules to express evidence
  - "If email content suggests type X, it is of type X"
  - "If A sends deadline emails to B, then A is the supervisor of B"
  - "If A is the supervisor of B, and A is the supervisor of C, then B and C are colleagues"



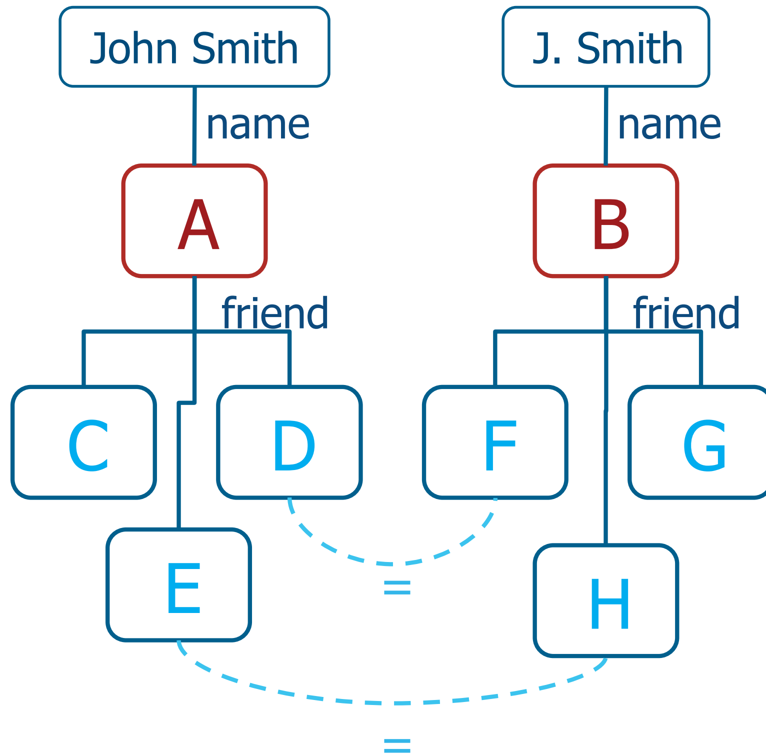
Entity Resolution

- Entities
  - People References
- Attributes
  - Name
- Relationships
  - Friendship
- Goal: Identify references that denote the same person



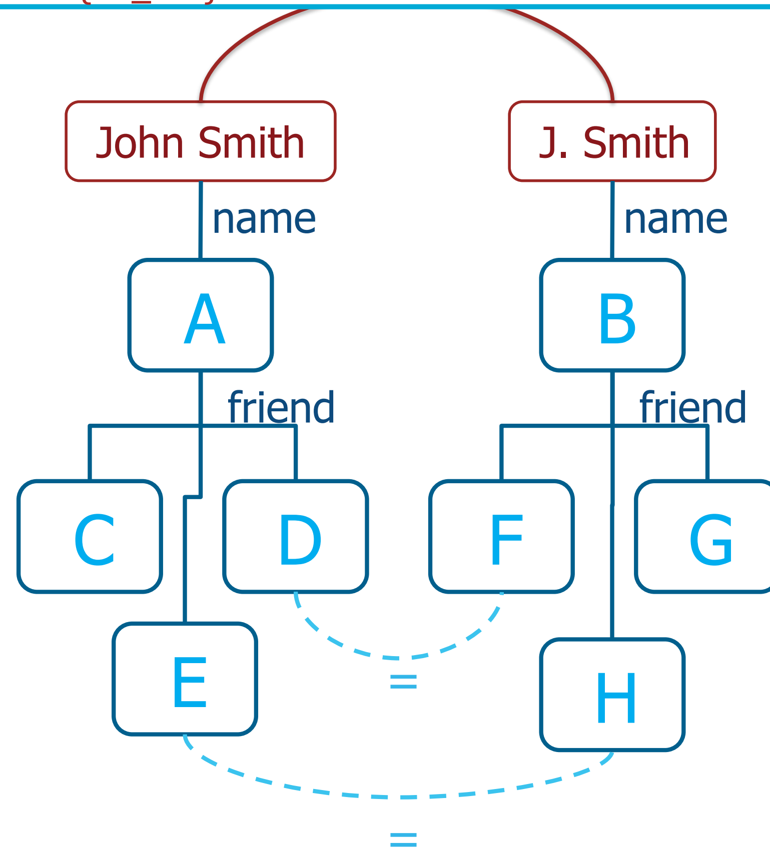
Entity Resolution

- References, names, friendships
- Use rules to express evidence
  - ‘’If two people have similar names, they are probably the same’’
  - ‘’If two people have similar friends, they are probably the same’’
  - ‘’If A=B and B=C, then A and C must also denote the same person’’



Entity Resolution

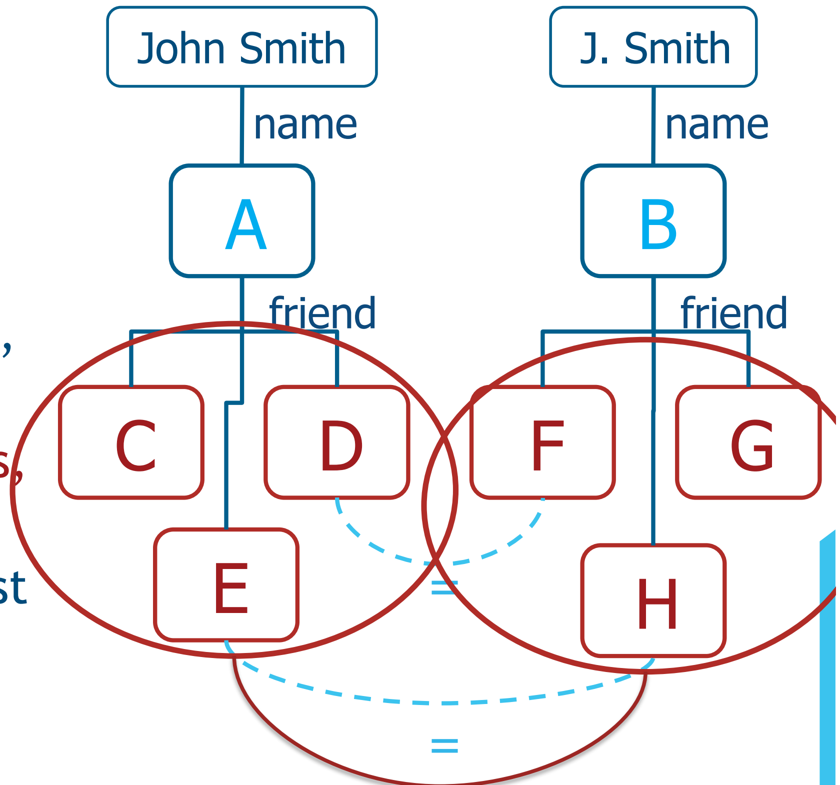
A.name $\approx_{\{str\_sim\}}$ B.name => A≈B : 0.8



- References, names, friendships
- Use rules to express evidence
  - ‘’If two people have similar names, they are probably the same’’
  - ‘’If two people have similar friends, they are probably the same’’
  - ‘’If A=B and B=C, then A and C must also denote the same person’’

Entity Resolution

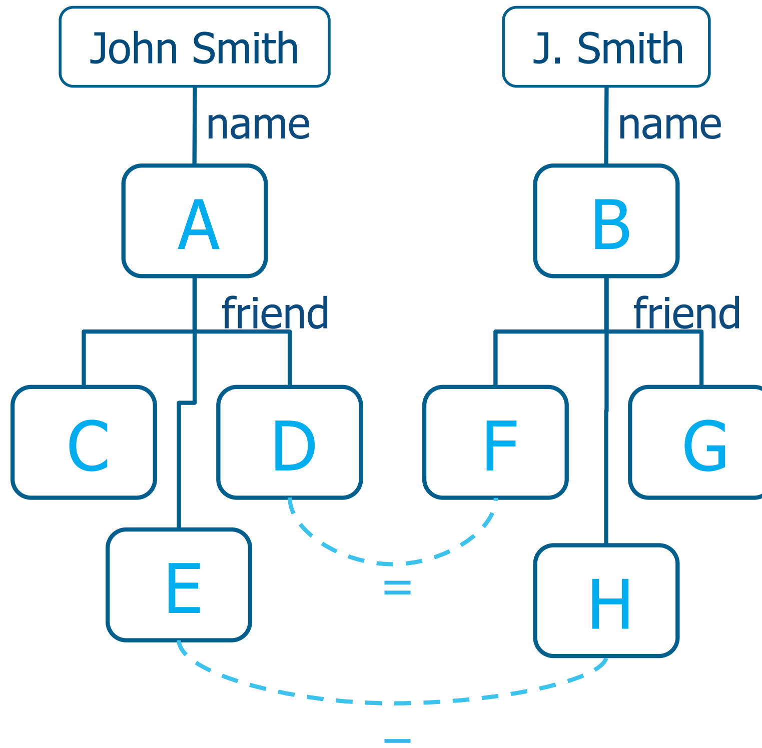
- References, names, friendships
- Use rules to express evidence
  - ‘‘If two people have similar names, they are probably the same’’
  - ‘‘If two people have similar friends, they are probably the same’’
  - ‘‘If A=B and B=C, then A and C must also denote the same person’’



$$\{A.friends\} \approx_{\{\}} \{B.friends\} \Rightarrow A \approx B : 0.6$$

Entity Resolution

- References, names, friendships
- Use rules to express evidence
  - ‘’ If two people have similar names, they are probably the same’’
  - ‘’ If two people have similar friends, they are probably the same’’
  - ‘’ If A=B and B=C, then A and C must also denote the same person’’



A≈B ^ B≈C => A≈C : ∞

Logic Foundation

http://psl.umiacs.umd.edu

[Broecheler, et al., UAI ‘10]

Rules

Ground Atoms

$$H_1 \lor \ldots H_m \leftarrow B_1 \land B_2 \land \ldots B_n$$

- Atoms are real valued
  - Interpretation I, atom A: I(A) $\in [0,1]$
  - We will omit the interpretation and write A $\in [0,1]$
- $\lor$, $\land$ are combination functions
  - T-norms: $[0,1]^n \rightarrow [0,1]$

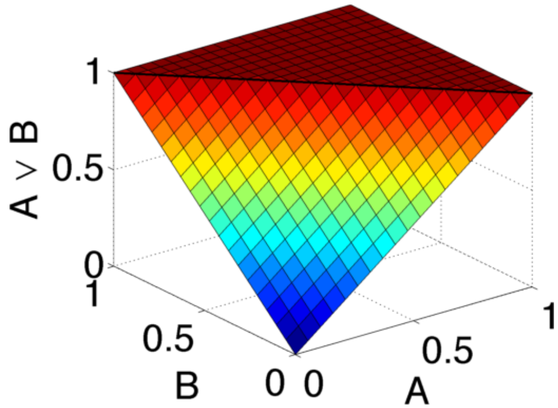
http://psl.umiacs.umd.edu

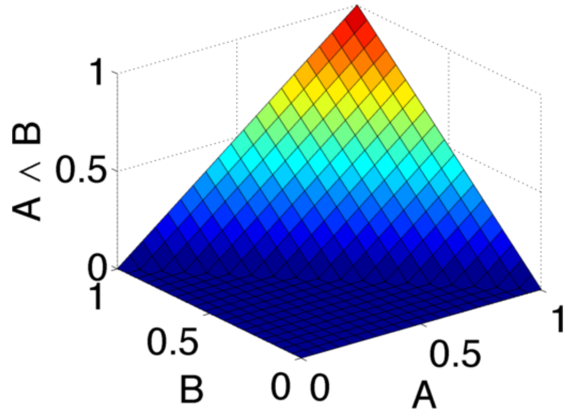
[Broecheler, et al., UAI ‘10]

Rules

$$H_1 \vee \ldots H_m \leftarrow B_1 \wedge B_2 \wedge \ldots B_n$$

- Combination functions (Lukasiewicz T-norm)
  - $A \vee B = \min(1, A + B)$
  - $A \wedge B = \max(0, A + B - 1)$





[Broecheler, et al., UAI '10]

Satisfaction

$$H_1 \vee \ldots H_m \leftarrow B_1 \wedge B_2 \wedge \ldots B_n$$

- Establish Satisfaction
  - $\vee (H_1,..,H_m) \leftarrow \wedge (B_1,..,B_n)$

$$\geq 0.5 \; H_1 \leftarrow B_1{:}0.7 \wedge B_2{:}0.8$$

[Broecheler, et al., UAI '10]

Distance to Satisfaction

$$H_1 \vee \ldots H_m \leftarrow B_1 \wedge B_2 \wedge \ldots B_n$$

- Distance to Satisfaction
  - $\max(\wedge (B_1,..,B_n) - \vee (H_1,..,H_m) , 0)$

$H_1{:}0.7 \leftarrow B_1{:}0.7 \wedge B_2{:}0.8$	0.0
$H_1{:}0.2 \leftarrow B_1{:}0.7 \wedge B_2{:}0.8$	0.3

http://psl.umiacs.umd.edu

[Broecheler, et al., UAI ‘10]

Rule Weights

$$W: H_1 \vee \ldots H_m \leftarrow B_1 \wedge B_2 \wedge \ldots B_n$$

- Weighted Distance to Satisfaction
  - $d(R,I) = W \bullet \max(\wedge(B_1,..,B_n) - \vee(H_1,..,H_m), 0)$

So far….

- Given a data set and a PSL program, we can construct a set of ground rules.
- Some of the atoms have fixed truth values and some have unknown truth values.
- For every assignment of truth values to the unknown atoms, we get a set of weighted distances from satisfaction.
- How to decide which is best?

Probabilistic Foundation

Probabilistic Model

Probability density over interpretation *I*

Ground rule's distance to satisfaction

$$d_r(I) = \max\{I_{r,\text{body}} - I_{r,\text{head}}, 0\}$$

$$P(I) = \frac{1}{Z} \exp\left[- \sum_{r \in R} w_r (d_r(I))^{p_r} \right]$$

Normalization constant

Ground rules

Rule weight

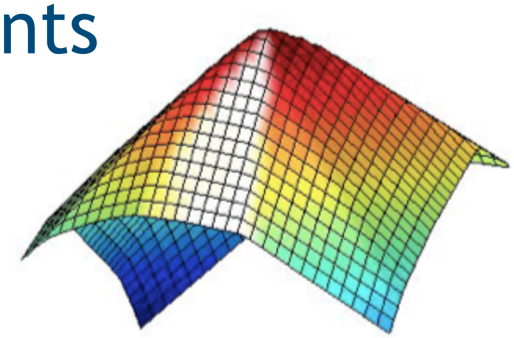
Distance exponent (in {1, 2})

Hinge-loss MRFs

Hinge-loss Markov Random Fields

$$P(\mathbf{Y} \mid \mathbf{X}) = \frac{1}{Z} \exp \left[-\sum_{j=1}^{m} w_j \max\{\ell_j(\mathbf{Y}, \mathbf{X}), 0\}^{p_j} \right]$$

- Continuous variables in [0,1]
- Potentials are hinge-loss functions
- Subject to arbitrary linear constraints
- Log-concave!



Inference as Convex Optimization

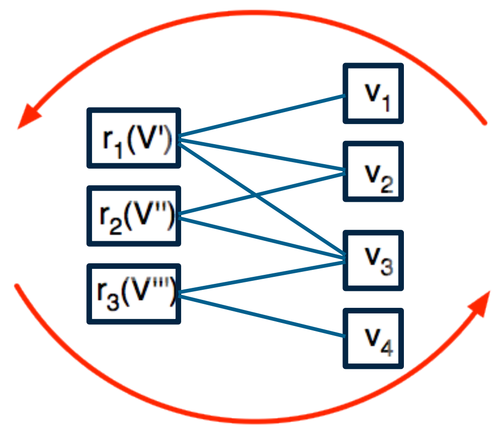
- Maximum Aposteriori Probability (MAP) Objective:

$$\arg\max_{\mathbf{Y}} P(\mathbf{Y} \mid \mathbf{X})$$
$$= \arg\min_{\mathbf{Y}} \sum_{j=1}^{m} w_j \max\{\ell_j(\mathbf{Y}, \mathbf{X}), 0\}^{p_j}$$

- This is convex!
- Can solve using off-the-shelf convex optimization packages
- … or custom solver

Consensus Optimization

- Idea: Decompose problem and solve sub-problems independently (in parallel), then merge results
  - Sub-problems are ground rules
  - Auxiliary variables enforce consensus across sub-problems



- Framework: ***Alternating direction method of multipliers*** (ADMM) [Boyd, 2011]

- Inference with ADMM is fast, scalable, and straightforward to implement [Bach et al., NIPS 2012, UAI 2013]

[Bach et al., UAI 2013; London et al., 2013]

Speed

Average running time

	Cora	Citeseer	Epinions	Activity
Discrete MRF	110.9 s	184.3 s	212.4 s	344.2 s
HL-MRF	**0.4 s**	**0.7 s**	**1.2 s**	**0.6 s**

- Inference in HL-MRFs is orders of magnitude faster than in discrete MRFs which use MCMC approximate inference
- In practice, scales linearly with the number of potentials

Compiling PSL → HL-MRF

- Ground out first-order rules
  - Variables: soft-truth values of atoms
  - Hinge-loss potentials: weighted ***distances to satisfaction*** of ground rules

- $$w : A \rightarrow B$$
  $$w : \neg A \vee B$$
  $$w \times (1 - \min\{1 - A + B, 1\})$$
  $$w \times \max\{A - B, 0\}$$

- The effect is assignments that satisfy weighted rules more are more probable

http://psl.umiacs.umd.edu

Inference Meta-Algorithm

Function: MAP-Inference

$I_0(\mathbf{y}) \leftarrow$ all zeros assignment
$R \leftarrow$ all grounded rules activated by $I(\mathbf{x}) \cup I_0(\mathbf{y})$
while *R has been updated* **do**
    $i \leftarrow$ current iteration
    $O \leftarrow \texttt{generateConvexProb}(R)$
    $I_i(\mathbf{y}) \leftarrow \texttt{optimize}(O)$
    foreach *Proposition* $y \in \mathbf{y}$ **do**
        if $I_i(y) > \theta(\theta = 0.01)$ **then**
            $R_y \leftarrow$ activated rules containing y $R \leftarrow R \cup R_y$
        end
    end
end

Each ground rule constitutes a linear or conic constraint, introducing a rule-specific "dissatisfaction" variable that is added to the objective function.

Inference Meta-Algorithm

Function: MAP-Inference

$I_0(\mathbf{y}) \leftarrow$ all zeros assignment
$R \leftarrow$ all grounded rules activated by $I(\mathbf{x}) \cup I_0(\mathbf{y})$
while *R has been updated* **do**
  $i \leftarrow$ current iteration
  $O \leftarrow \texttt{generateConvexProb}(R)$
  $I_i(\mathbf{y}) \leftarrow \texttt{optimize}(O)$
  foreach *Proposition* $y \in \mathbf{y}$ **do**
    if $I_i(y) > \theta(\theta = 0.01)$ **then**
      $R_y \leftarrow$ activated rules containing y $R \leftarrow R \cup R_y$
    end
  end
end

Find most probable assignment using consensus optimization (ADMM) subroutine

Inference Meta-Algorithm

Function: MAP-Inference

$I_0(\mathbf{y}) \leftarrow$ all zeros assignment
$R \leftarrow$ all grounded rules activated by $I(\mathbf{x}) \cup I_0(\mathbf{y})$
while *R has been updated* **do**
  $i \leftarrow$ current iteration
  $O \leftarrow \texttt{generateConvexProb}(R)$
  $I_i(\mathbf{y}) \leftarrow \texttt{optimize}(O)$
  foreach *Proposition* $y \in \mathbf{y}$ **do**
    if $I_i(y) > \theta(\theta = 0.01)$ **then**
      $R_y \leftarrow$ activated rules containing y $R \leftarrow R \cup R_y$
    end
  end
end

Conservative Grounding: Most rules trivially have satisfaction distance=0. Save time and space by not grounding them out in the first place.

Don't reason about it if you don't absolutely have to!

Miao, Liu, Huang, Getoor, IEEE Big Data 2013

Distributed MAP Inference

- ADMM consensus optimization problem can be implemented naturally in distributed setting
- For k+1 iteration, it consists three steps in which sub problems can run independently (1st and 2nd step):
  1. Update Lagrangian multiplier

$$y_j^{k+1} \leftarrow y_j^k + \rho (x_j^k - X_j^k)$$

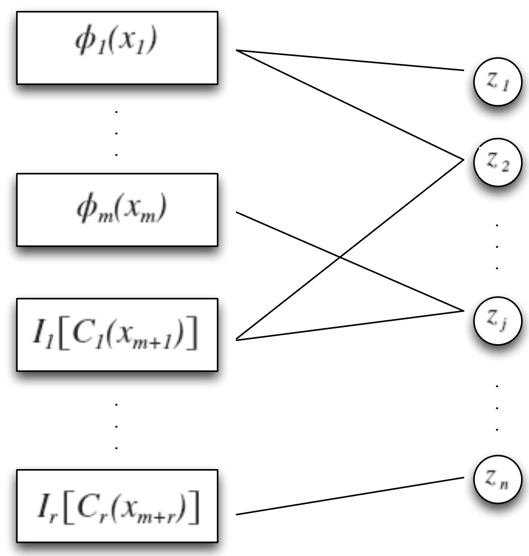
  2. Update each sub problem

$$x_j^{k+1} \leftarrow \arg\min_{x_j} \Lambda_j \phi_j(x_j) + \frac{\rho}{2} \left\| x_j - X_j^k + \frac{1}{\rho} y_j^{k+1} \right\|_2^2$$

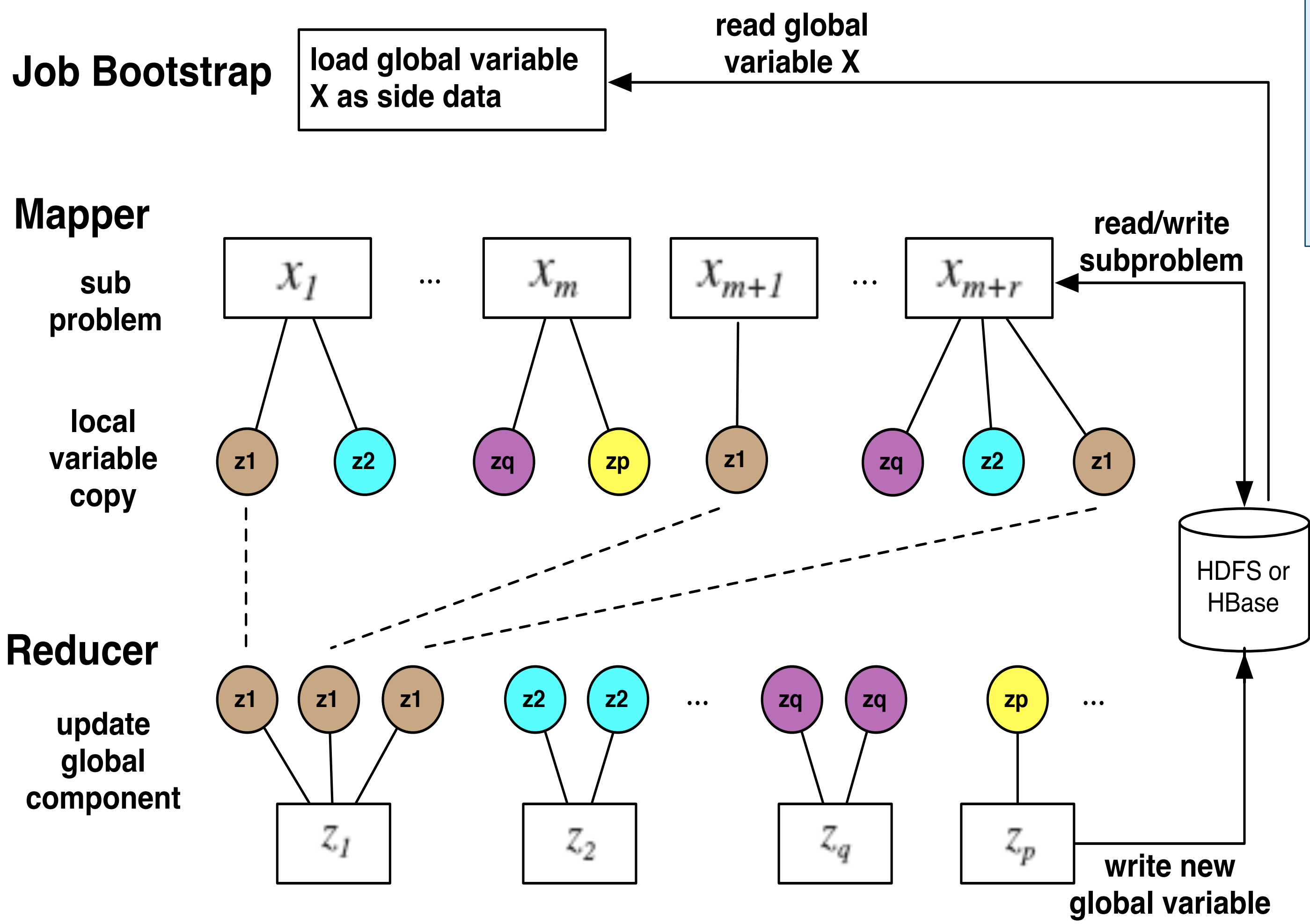
$$x_j^{k+1} \leftarrow \arg\min_{x_j} I_j[C_j(x_j)] + \frac{\rho}{2} \left\| x_j - X_j^k + \frac{1}{\rho} y_j^{k+1} \right\|_2^2$$

  3. Update the global variables

$$z_g^{k+1} \leftarrow \frac{1}{s_g} \sum_{G(i,j)=g} \left(x_i^{k+1} + \frac{y_i^{k+1}}{\rho} \right)_j$$



Distributed MAP: MapReduce



Pros:

- Straightforward Design

Cons:

- Job bootstrapping cost between iterations
- Difficult to schedule subset of nodes to run.

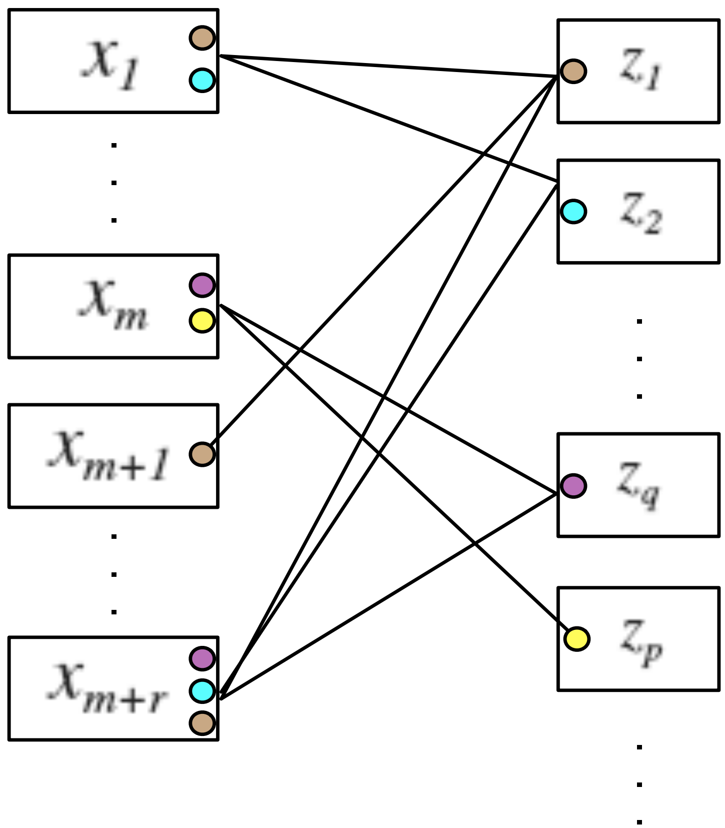
Distributed MAP: GraphLab

Advantages:

- No need to touch disk, no job bootstrap-ping cost
- Easy to express local convergence conditions to schedule only subset of nodes.

sub problem node

global variable compone



update i **update i+1**

```
gather
   get z

apply
   update y
   update x

scatter
   notify z
```

```
gather
   get local z,y

apply
   update z

scatter
   unless converge
       notify X
```

Experimental Results

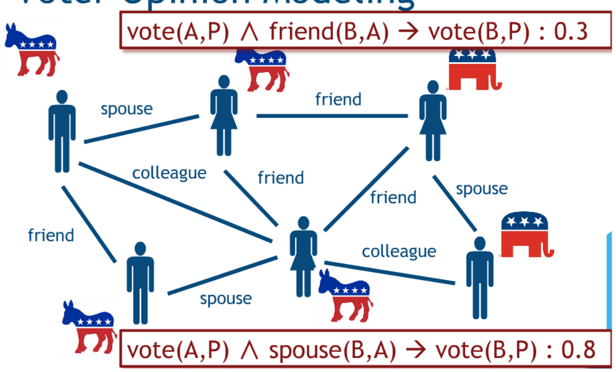
- Using PSL for knowledge graph cleaning task
  - 16M+ vertices, 22M+ edges, for small running instances
  - Takes 100 minutes to finish in Java single machine implementation using 40G+ memory
  - Distributed GraphLab implementation takes less than 15 minutes using 4 smaller machines
  - Possible to use commodity machines on large models!

Miao, Liu, Huang, Getoor, BigData ’13

Experimental Results

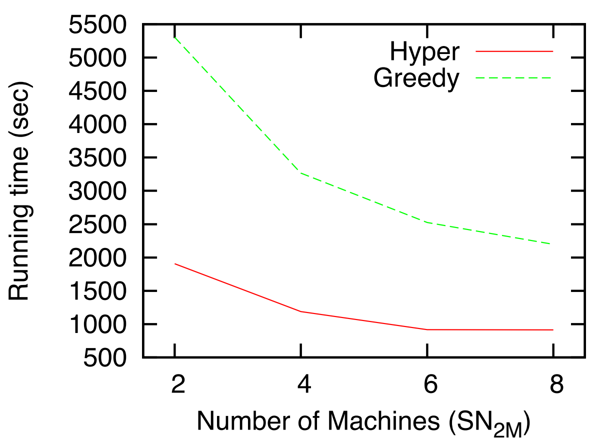
Voter model using commodity machines

Voter Opinion Modeling

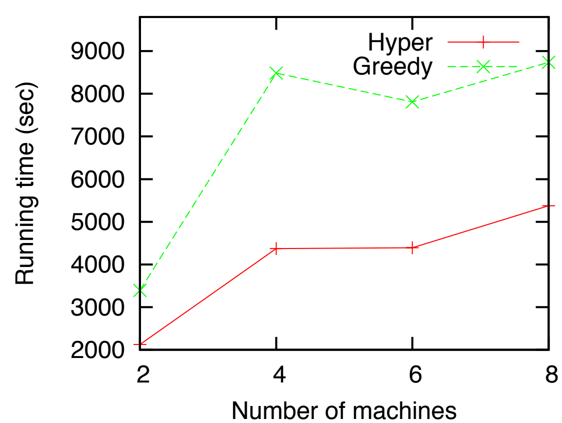


Name	|Subproblem|	|Consensus|	|Edge|	Fit in One Machine?	Run time (sec) |m| = 8
SN_{1M}	3.3M	1.1M	6M	Yes	2230
SN_{2M}	6.6M	2.1M	12M	No	3997
SN_{3M}	10M	3.1M	18M	No	4395
SN_{4M}	13M	4.2M	24M	No	5376

Machine: Intel Core2 Quad CPU 2.66GHz machines with 4GB RAM running Ubuntu 12.04 Linux



Strong scaling with fixed dataset



Weak scaling with increasing size

Weight Learning

Weight Learning

- Learn from training data
- No need to hand-code rule-weights
- Various methods:
  - approximate maximum likelihood

    Broecheler et al., UAI '10
  - maximum pseudo-likelihood
  - large-margin estimation

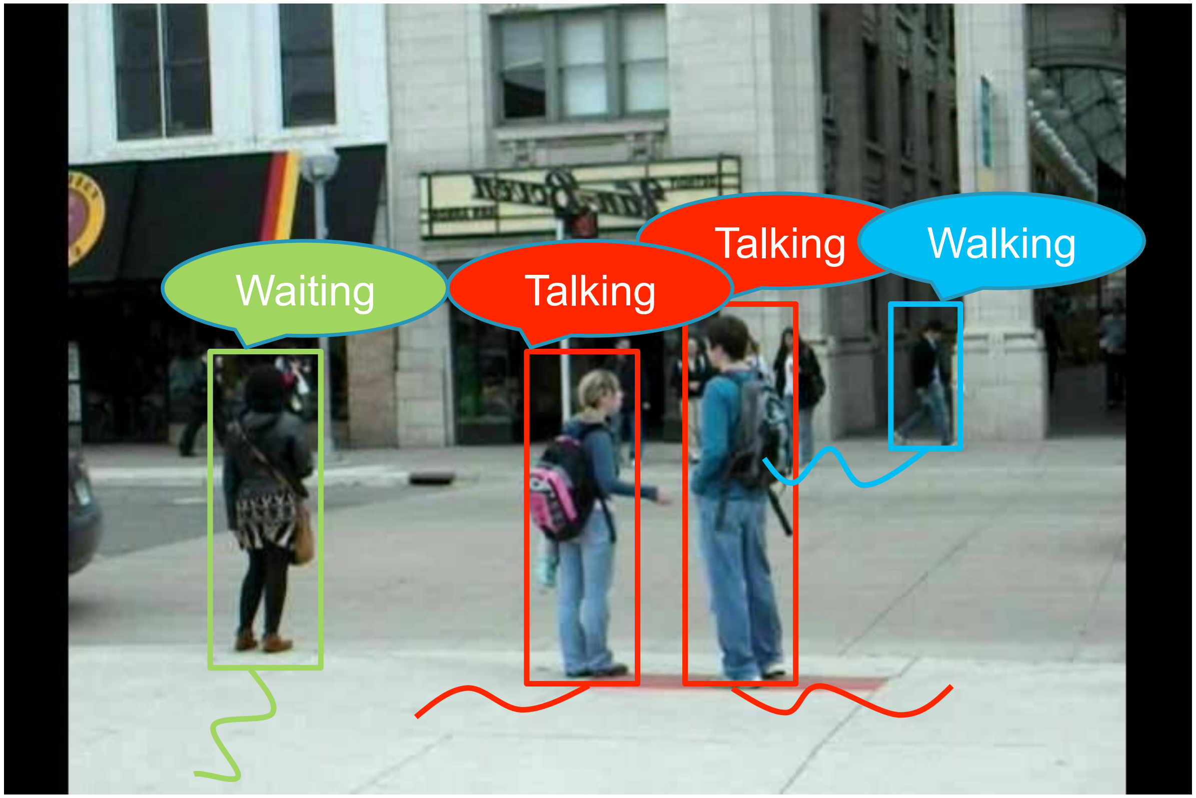
    Bach, et al., UAI 2013

Weight Learning

- State-of-the-art supervised-learning performance on
  - Collective classification
  - Social-trust prediction
  - Preference prediction
  - Image reconstruction

Example PSL Program

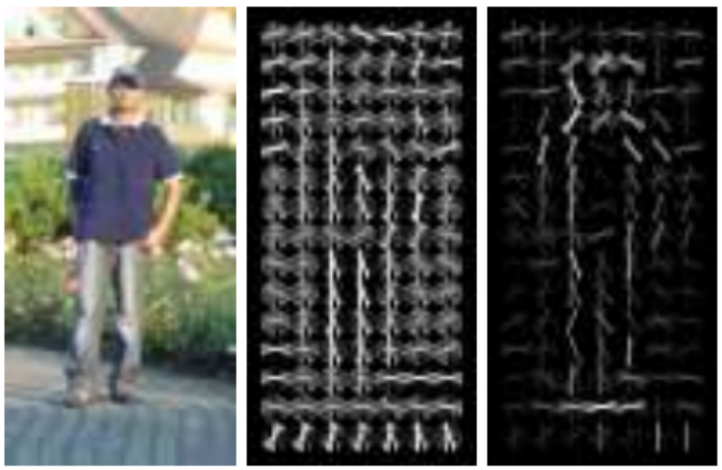
Collective Activity Detection



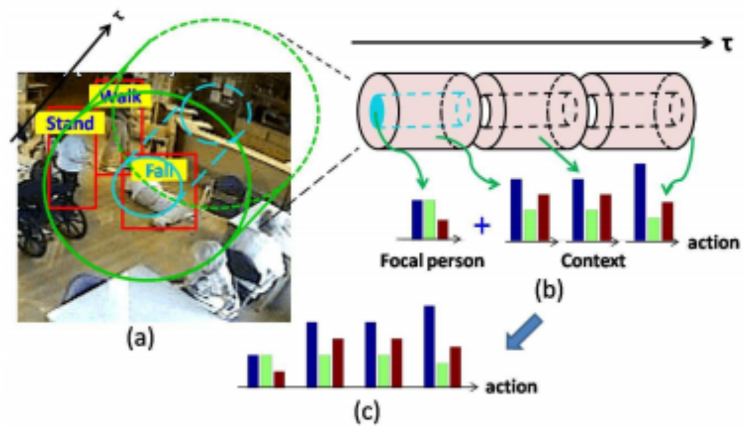
- Objective: Classify actions of individuals in a video sequence
  - Requires tracking the multiple targets, performing ID maintenance

Incorporate Low-level Detectors

Histogram of Oriented Gradients (HOG) [Dalal & Triggs, CVPR 2005]



Action Context Descriptors (ACD) [Lan et al., NIPS 2010]



For each action a, define PSL rule:

$$w_{local,a} : \text{Doing}(X, a) \leftarrow \text{Detector}(X, a)$$

e.g., $w_{local,\mathit{walking}} : \text{Doing}(X, \mathit{walking}) \leftarrow \text{Detector}(X, \mathit{walking})$

Easily Encode Intuitions

- Proximity: People that are close (in frame) are likely doing the same action

$$w_{prox,a} : \text{Doing}(X, a) \leftarrow \text{Close}(X, Y) \wedge \text{Doing}(Y, a)$$

  - Closeness is measured via a radial basis function

- Proximity: People are likely to continue doing the same action

$$w_{persist,a} : \text{Doing}(Y, a) \leftarrow \text{Same}(X, Y) \wedge \text{Doing}(X, a)$$

  - Requires tracking & ID maintenance rule:

$$w_{id} : \text{Same}(X,Y) \leftarrow \text{Sequential}(X,Y) \wedge \text{Close}(X,Y)$$

Other Rules

- Action transitions
- Frame/scene consistency
- Priors
- (Partial-)Functional Constraints

Collective Activity Detection Model

w_{id} : Same(X, Y) ← Sequential(X, Y) ∧ Close(X, Y)

$w_{idprior}$: ~SamePerson(X, Y)

For all actions a:

$w_{local,a}$: Doing(X, a) ← Detector(X, a)

$w_{frame,a}$: Doing(X, a) ← Frame(X, F) ∧ FrameAction(F, a)

$w_{prox,a}$: Doing(X, a) ← Close(X, Y) ∧ Doing(Y, a)

$w_{persist,a}$: Doing(Y, a) ← SamePerson(X, Y) ∧ Doing(X, a)

$w_{prior,a}$: ~Doing(X, a)

[London et al., 2013]

PSL Code

```
/*** MODEL DEFINITION ***/

PSLModel m = new PSLModel(this, data);

/* PREDICATES */

// target
m.add predicate: "doing", types: [ArgumentType.UniqueID,ArgumentType.Integer];
m.add predicate: "sameObj", types: [ArgumentType.UniqueID,ArgumentType.UniqueID];

// observed
m.add predicate: "inFrame", types: [ArgumentType.UniqueID,ArgumentType.Integer,ArgumentType.Integer];
m.add predicate: "inSameFrame", types: [ArgumentType.UniqueID,ArgumentType.UniqueID];
m.add predicate: "inSeqFrames", types: [ArgumentType.UniqueID,ArgumentType.UniqueID];
m.add predicate: "dims", types: [ArgumentType.UniqueID,ArgumentType.Integer,ArgumentType.Integer];
m.add predicate: "detector", types: [ArgumentType.UniqueID,ArgumentType.Integer];
m.add predicate: "frameAction", types: [ArgumentType.Integer,ArgumentType.Integer];

/* FUNCTIONAL PREDICATES */

m.add function: "close", implementation: new ClosenessFunction(0, 1e6, 0.1, true);
m.add function: "seqClose", implementation: new ClosenessFunction(100, 4.0, 0.7, true);
m.add function: "notMoved", implementation: new ClosenessFunction(10, 1.0, 0.0, false);
```

PSL Code

```
/* TRACKING RULES */

// ID maintenance
m.add rule: ( inSeqFrames(BB1,BB2) & dims(BB1,X1,Y1) & dims(BB2,X2,Y2)
               & seqClose(X1,X2,Y1,Y2) ) >> sameObj(BB1,BB2), weight: 1.0;

// Prior on sameObj
m.add rule: ~sameObj(BB1,BB2), weight: 0.01;

/* ACTION RULES */

def actions = ["crossing","standing","queueing","walking","talking"];
for (int a : actions) {

    // Local detectors
    m.add rule: detector(BB,a) >> doing(BB,a), weight: 1.0;

    // Frame consistency
    m.add rule: ( inFrame(BB,S,F) & frameLabel(F,a) ) >> doing(BB,a), weight: 0.1;

    // Persistence
    m.add rule: ( sameObj(BB1,BB2) & doing(BB1,a) ) >> doing(BB2,a), weight: 1.0;

    // Proximity
    m.add rule: ( inSameFrame(BB1,BB2) & doing(BB1,a) & dims(BB1,X1,Y1) & dims(BB2,X2,Y2)
                   & close(X1,X2,Y1,Y2) ) >> doing(BB2,a), weight: 0.1;

    // Prior on doing
    m.add rule: ~doing(BB,a), weight: 0.01;

}
```

PSL Code

```
/* FUNCTIONAL CONSTRAINTS */

// Functional constraint on doing means that it should sum to 1 for each BB
m.add PredicateConstraint.Functional, on: doing;

// (Inverse) Partial functional constraint on sameObj
m.add PredicateConstraint.PartialFunctional, on: sameObj;
m.add PredicateConstraint.PartialInverseFunctional, on: sameObj;
```

Foundations Summary

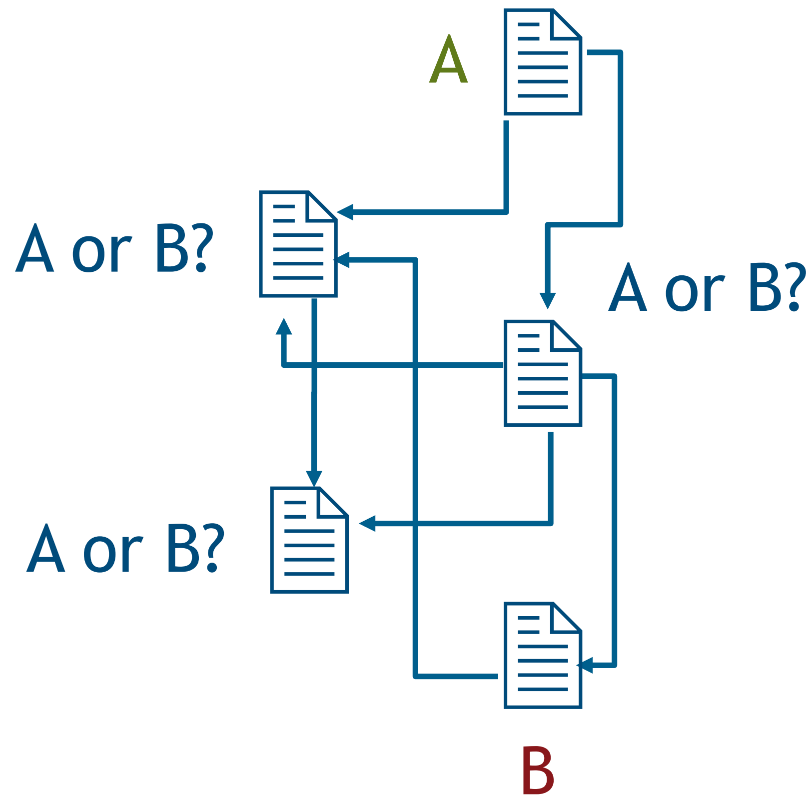
Foundations Summary

- Design probabilistic models using declarative language
  - Syntax based on first-order logic
- Inference of most-probable explanation is fast convex optimization (ADMM)
- Learning algorithms for training rule weights from labeled data

PSL Applications

Document Classification

- Given a networked collection of documents
- Observe some labels
- Predict remaining labels using
  - link direction
  - inferred class label



	Citeseer	Cora
HL-MRF-Q (MLE)	**0.729**	**0.816**
HL-MRF-Q (MPLE)	**0.729**	**0.818**
HL-MRF-Q (LME)	0.683	0.789
HL-MRF-L (MLE)	**0.724**	0.802
HL-MRF-L (MPLE)	**0.729**	**0.808**
HL-MRF-L (LME)	0.695	0.789
MLN (MLE)	0.686	0.756
MLN (MPLE)	0.715	0.797
MLN (LME)	0.687	0.783

Accuracy for collective classification. The label accuracy of the highest-scoring category for various HL-MRFs and MLNs. Scores statistically equivalent to the best scoring method are typed in bold.

Computer Vision Applications

- Low-level vision:
  - image reconstruction
- High-level vision:
  - activity recognition in videos

[Bach, et al., UAI 2013]

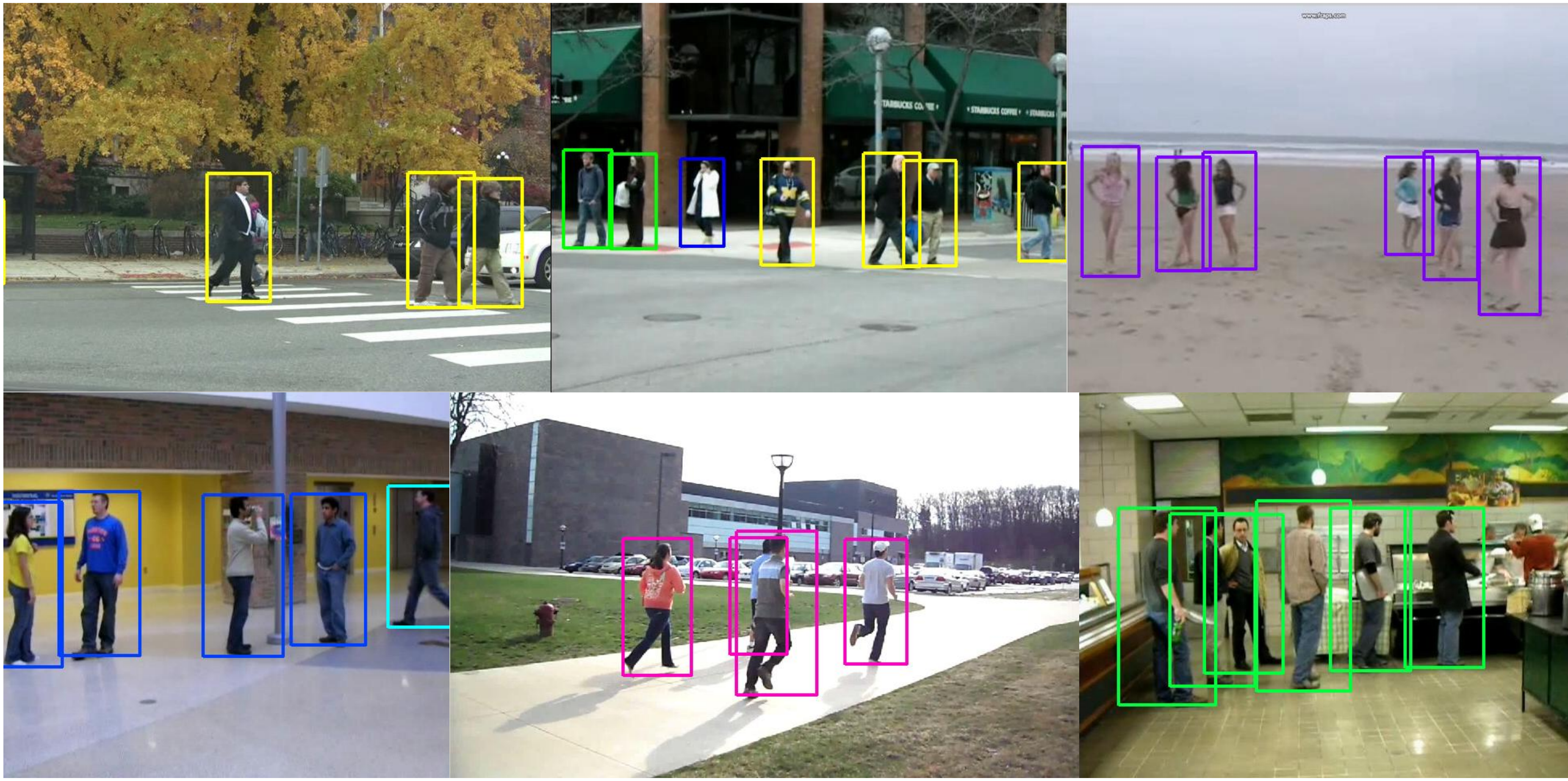
Image Reconstruction

	HL-MRF-Q (MLE)	SPN	DBM	DBN	PCA	NN
Caltech-Left	1751	1815	2998	4960	2851	2327
Caltech-Bottom	1863	1924	2656	3447	1944	2575
Olivetti-Left	932	942	1866	2386	1076	1527
Olivetti-Bottom	1202	918	2401	1931	1265	1793

RMSE reconstruction error

[London, et al., CVPR WS 2013]

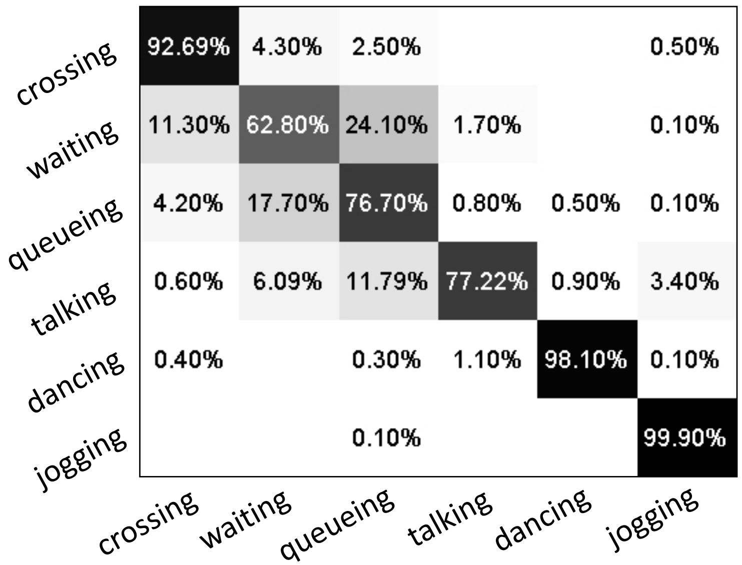
Activity Recognition in Videos



crossing waiting queueing walking talking dancing jogging

[London, et al., CVPR WS 2013]

Results on Activity Recognition



Recall matrix between different activity types

Method	5 Activities		6 Activities	
	Acc.	F1	Acc.	F1
HOG	.474	.481	.596	.582
HL-MRF + HOG	**.598**	**.603**	**.793**	**.789**
ACD	.675	.678	.835	.835
HL-MRF + ACD	**.692**	**.693**	**.860**	**.860**

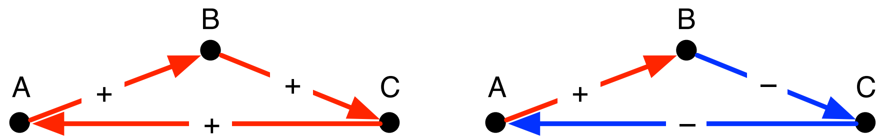
Accuracy metrics compared against baseline features

Social Trust Prediction

- Competing models from social psychology of strong ties
  - Structural balance [Granovetter ’73]
  - Social status [Cosmides et al., ’92]
- Effects of both models present in online social networks
  - [Leskovec, Huttenlocher, & Kleinberg, 2010]

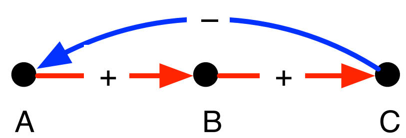
Structural Balance vs. Social Status

- Structural balance: strong ties are governed by tendency toward balanced triads



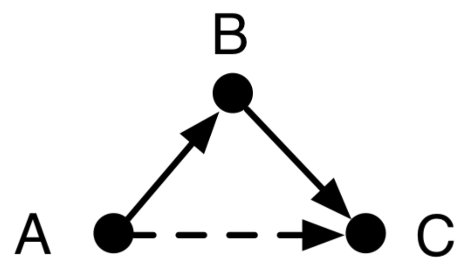
  - e.g., the enemy of my enemy...

- Social status: strong ties indicate unidirectional respect, “looking up to”, expertise status



  - e.g., patient-nurse-doctor, advisor-advisee

Structural Balance in PSL

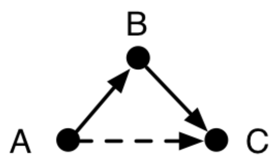


$$\text{Knows}(A, B) \wedge \text{Knows}(B, C) \wedge \text{Knows}(A, C) \\ \wedge \text{Trusts}(A, B) \wedge \text{Trusts}(B, C) \Rightarrow \text{Trusts}(A, C),$$

$$\begin{aligned} \text{Tr}(A, B) \wedge \text{Tr}(B, C) &\Rightarrow \text{Tr}(A, C), \\ \text{Tr}(A, B) \wedge \neg\text{Tr}(B, C) &\Rightarrow \neg\text{Tr}(A, C), \\ \neg\text{Tr}(A, B) \wedge \text{Tr}(B, C) &\Rightarrow \neg\text{Tr}(A, C), \\ \neg\text{Tr}(A, B) \wedge \neg\text{Tr}(B, C) &\Rightarrow \text{Tr}(A, C) \end{aligned}$$

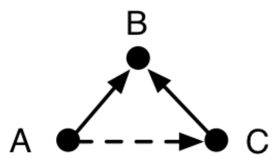
[Huang, et al., SBP '13]

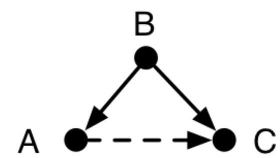
Structural Balance in PSL



$$
\begin{aligned}
\mathrm{Tr}(A,B) \wedge \mathrm{Tr}(B,C) &\Rightarrow \mathrm{Tr}(A,C),\\
\mathrm{Tr}(A,B) \wedge \neg\mathrm{Tr}(B,C) &\Rightarrow \neg\mathrm{Tr}(A,C),\\
\neg\mathrm{Tr}(A,B) \wedge \mathrm{Tr}(B,C) &\Rightarrow \neg\mathrm{Tr}(A,C),\\
\neg\mathrm{Tr}(A,B) \wedge \neg\mathrm{Tr}(B,C) &\Rightarrow \mathrm{Tr}(A,C),
\end{aligned}
$$

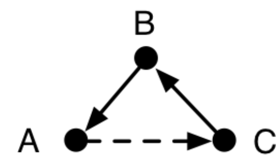
$$
\begin{aligned}
\mathrm{Tr}(A,B) \wedge \mathrm{Tr}(C,B) &\Rightarrow \mathrm{Tr}(A,C),\\
\mathrm{Tr}(A,B) \wedge \neg\mathrm{Tr}(C,B) &\Rightarrow \neg\mathrm{Tr}(A,C),\\
\neg\mathrm{Tr}(A,B) \wedge \mathrm{Tr}(C,B) &\Rightarrow \neg\mathrm{Tr}(A,C),\\
\neg\mathrm{Tr}(A,B) \wedge \neg\mathrm{Tr}(C,B) &\Rightarrow \mathrm{Tr}(A,C),
\end{aligned}
$$





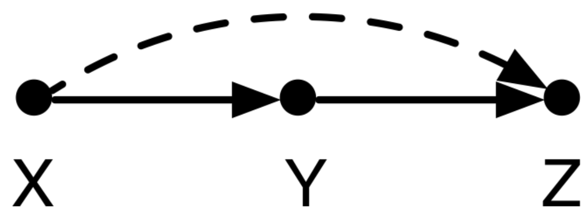
$$
\begin{aligned}
\mathrm{Tr}(B,A) \wedge \mathrm{Tr}(B,C) &\Rightarrow \mathrm{Tr}(A,C),\\
\mathrm{Tr}(B,A) \wedge \neg\mathrm{Tr}(B,C) &\Rightarrow \neg\mathrm{Tr}(A,C),\\
\neg\mathrm{Tr}(B,A) \wedge \mathrm{Tr}(B,C) &\Rightarrow \neg\mathrm{Tr}(A,C),\\
\neg\mathrm{Tr}(B,A) \wedge \neg\mathrm{Tr}(B,C) &\Rightarrow \mathrm{Tr}(A,C),
\end{aligned}
$$

$$
\begin{aligned}
\mathrm{Tr}(B,A) \wedge \mathrm{Tr}(C,B) &\Rightarrow \mathrm{Tr}(A,C),\\
\mathrm{Tr}(B,A) \wedge \neg\mathrm{Tr}(C,B) &\Rightarrow \neg\mathrm{Tr}(A,C),\\
\neg\mathrm{Tr}(B,A) \wedge \mathrm{Tr}(C,B) &\Rightarrow \neg\mathrm{Tr}(A,C),\\
\neg\mathrm{Tr}(B,A) \wedge \neg\mathrm{Tr}(C,B) &\Rightarrow \mathrm{Tr}(A,C)
\end{aligned}
$$



[Huang, et al., SBP '13]

Social Status in PSL

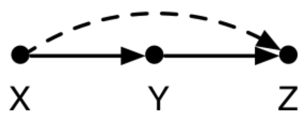


$$\mathrm{Tr}(X, Y) \wedge \mathrm{Tr}(Y, Z) \Rightarrow \mathrm{Tr}(X, Z)$$

$$\neg\mathrm{Tr}(X, Y) \wedge \neg\mathrm{Tr}(Y, Z) \Rightarrow \neg\mathrm{Tr}(X, Z)$$

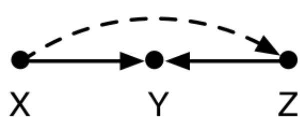
[Huang, et al., SBP '13]

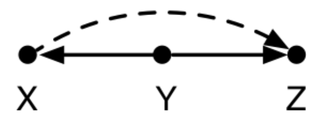
Social Status in PSL



$$\mathrm{Tr}(X, Y) \wedge \mathrm{Tr}(Y, Z) \Rightarrow \mathrm{Tr}(X, Z),$$
$$\neg\mathrm{Tr}(X, Y) \wedge \neg\mathrm{Tr}(Y, Z) \Rightarrow \neg\mathrm{Tr}(X, Z),$$

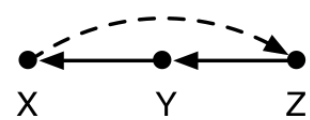
$$\mathrm{Tr}(X, Y) \wedge \neg\mathrm{Tr}(Z, Y) \Rightarrow \mathrm{Tr}(X, Z),$$
$$\neg\mathrm{Tr}(X, Y) \wedge \mathrm{Tr}(Z, Y) \Rightarrow \neg\mathrm{Tr}(X, Z),$$





$$\mathrm{Tr}(Y, X) \wedge \neg\mathrm{Tr}(Y, Z) \Rightarrow \neg\mathrm{Tr}(X, Z),$$
$$\neg\mathrm{Tr}(Y, X) \wedge \mathrm{Tr}(Y, Z) \Rightarrow \mathrm{Tr}(X, Z),$$

$$\mathrm{Tr}(Y, X) \wedge \mathrm{Tr}(Z, Y) \Rightarrow \neg\mathrm{Tr}(X, Z),$$
$$\neg\mathrm{Tr}(Y, X) \wedge \neg\mathrm{Tr}(Z, Y) \Rightarrow \mathrm{Tr}(X, Z)$$



[Huang, et al., SBP '13]

Evaluation

- User-user trust ratings from two different online social networks
- Observe some ratings, predict held-out
- Eight-fold cross validation on two data sets:
  - FilmTrust - movie review network, trust ratings from 1-10
  - Epinions - product review network, trust / distrust ratings {-1, 1}

FilmTrust Experiment

- Normalize [1,10] rating to [0,1]
- Prune network to largest connected-component
- 1,754 users, 2,055 relationships
- Compare mean average error, Spearman’s rank coefficient, and Kendall-tau distance

Method	MAE	τ	ρ	MAE*	τ*	ρ*
Average	**0.210**	n/a	n/a	n/a	n/a	n/a
EigenTrust	0.339	−0.054	−0.074	0.339	−0.054	−0.074
TidalTrust	0.229	0.059	0.078	0.236	0.089	0.117
PSL-Balance	**0.207**	**0.136**	**0.176**	**0.193**	**0.235**	**0.314**
PSL-Balance-Recip	**0.207**	**0.139**	**0.188**	**0.193**	**0.241**	**0.318**
PSL-Status	0.224	**0.112**	**0.144**	**0.230**	**0.205**	**0.277**
PSL-Status-Inv	0.224	0.065	0.085	0.238	**0.143**	**0.189**

* measured on only non-default predictions

[Huang, et al., SBP '13]

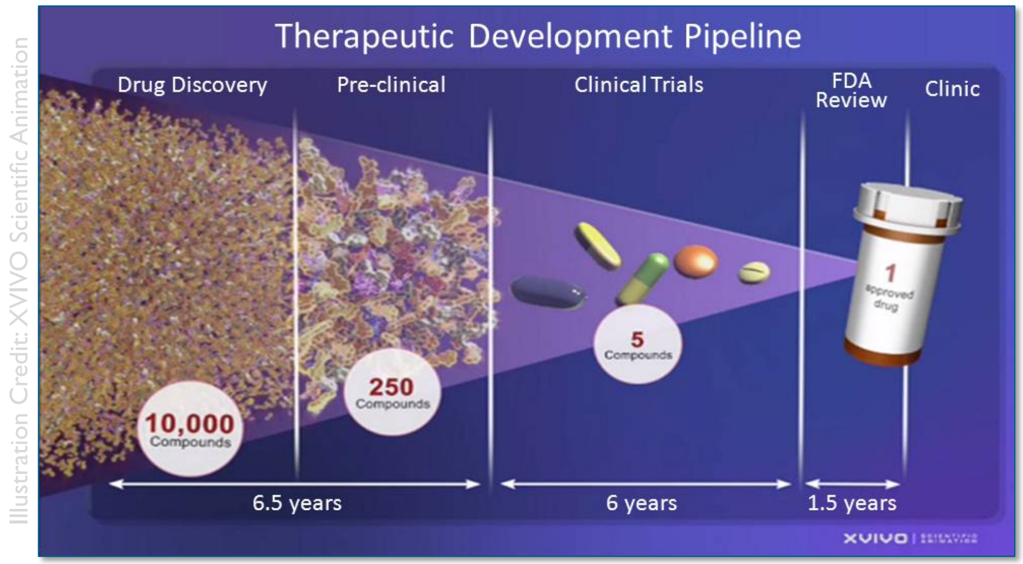
Epinions Experiment

- Snowball sample of 2,000 users from Epinions data set
- 8,675 trust scores normalized to {0,1}
- Measure area under precision-recall curve for distrust edges (rarer class)

Method	AUC
Average	0.070
PSL-Balance	**0.317**
PSL-Balance-Recip	**0.343**
PSL-Status	**0.297**
PSL-Status-Inv	0.280
EigenTrust	0.131
TidalTrust	0.130

[Fakhraei, et al., BioKDD'13]

Drug-Target Interaction Prediction



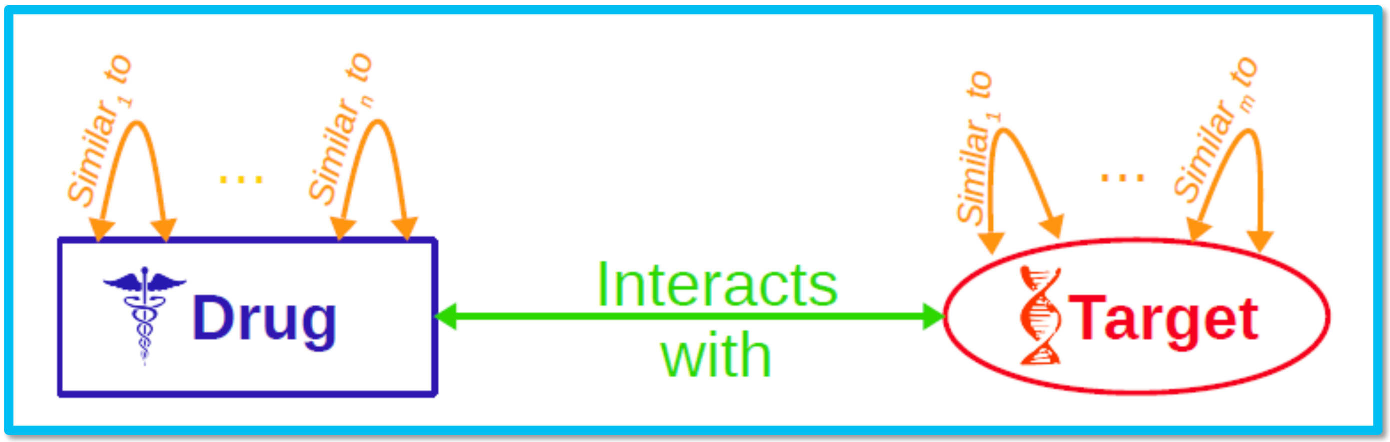
- New drugs take a decade to reach market.
- Development cost reaches 2 billion US dollars.
- Most novel drug candidates never get approved.

Drug repurposing:
Finding new uses for approved drugs

[Fakhraei, et al., BioKDD'13]

Drug-Target Interaction Prediction

Computational predictions focus biological investigations



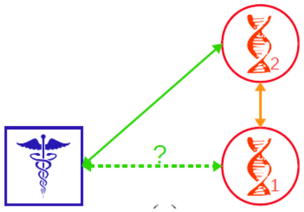
Data: drug-target (gene product) interaction network
+ drug-drug and target-target similarities

Task: link prediction

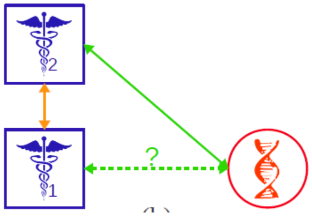
[Fakhraei, et al., BioKDD'13]

Drug-Target Interaction Prediction

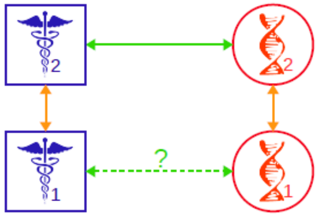
$$\mathrm{SimilarTarget}_\beta(T_1, T_2) \wedge Interacts(D, T_2) \rightarrow Interacts(D, T_1)$$



$$\mathrm{SimilarDrug}_\alpha(D_1, D_2) \wedge Interacts(D_2, T) \rightarrow Interacts(D_1, T)$$



$$\begin{aligned}\mathrm{SimilarDrug}_\alpha(D_1, D_2) &\wedge SimilarTarget_\beta(T_1, T_2)\\ &\wedge Interacts(D_2, T_2) \rightarrow Interacts(D_1, T_1)\end{aligned}$$

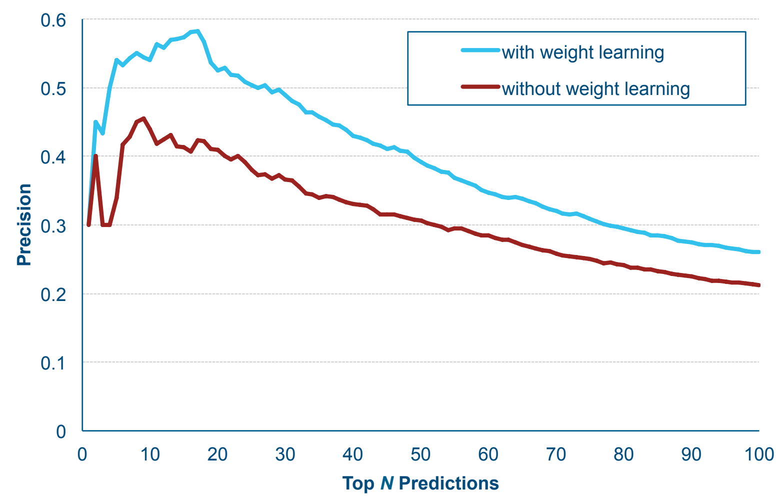


[Fakhraei, et al., BioKDD'13]

Drug-Target Interaction Prediction

- 315 Drugs, 250 Targets
- 78,750 possible interactions, 1,306 observed interactions
- 5 drug-drug similarities, 3 target-target similarities

Method	AUROC	Condition
PSL	0.931 ± 0.018	10-fold CV
Perlman, et al. 2011	0.935	with sampling
Yamanishi, et al. 2008	0.884	
Bleakley, et al. 2009	0.814	



[Bach, et al., ICML WS 2013]

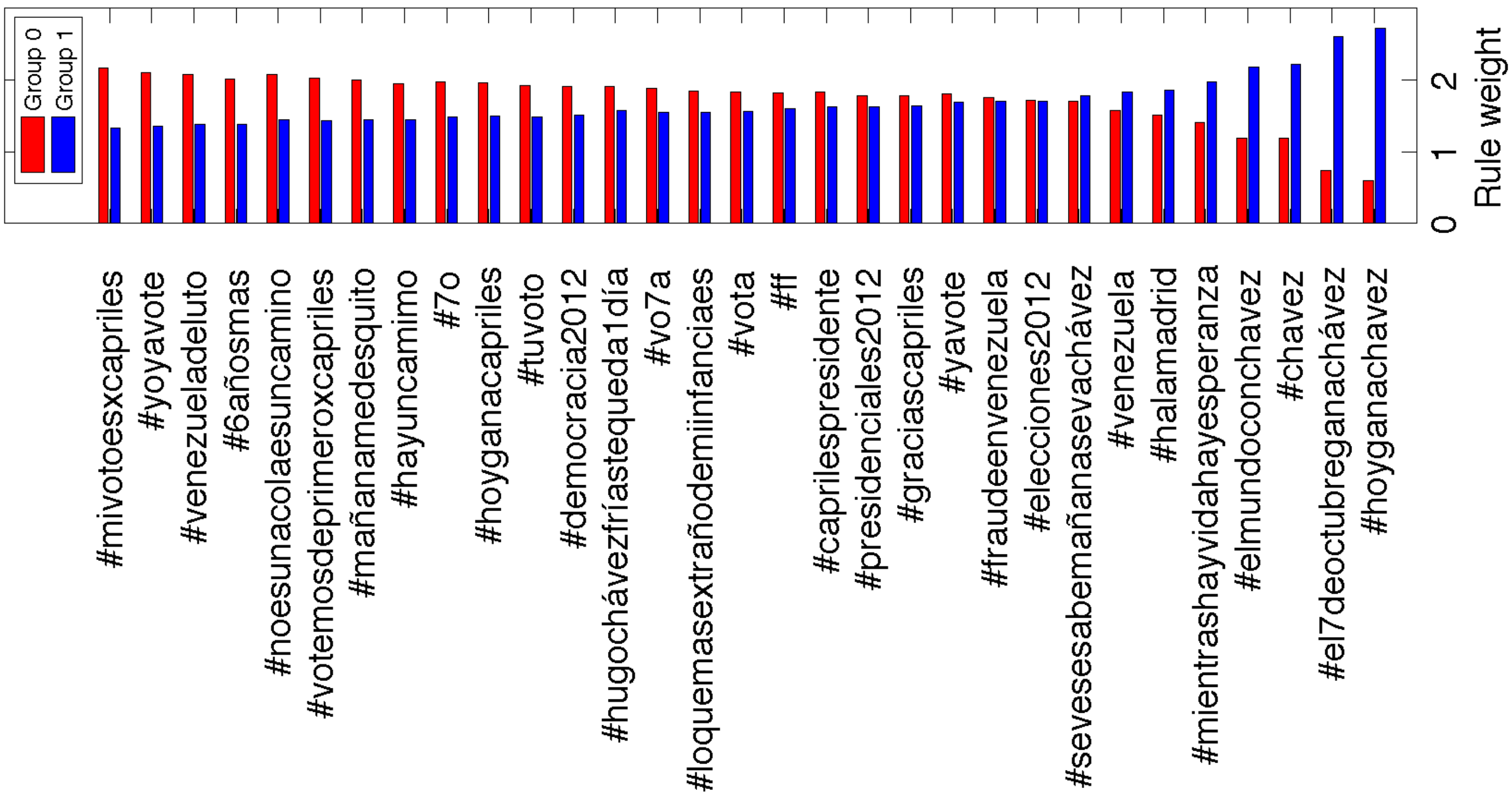
Learning Latent Groups

- Can we better understand political discourse in social media by learning groups of similar people?
- Case study: 2012 Venezuelan Presidential Election
  - Incumbent: Hugo Chávez
  - Challenger: Henrique Capriles

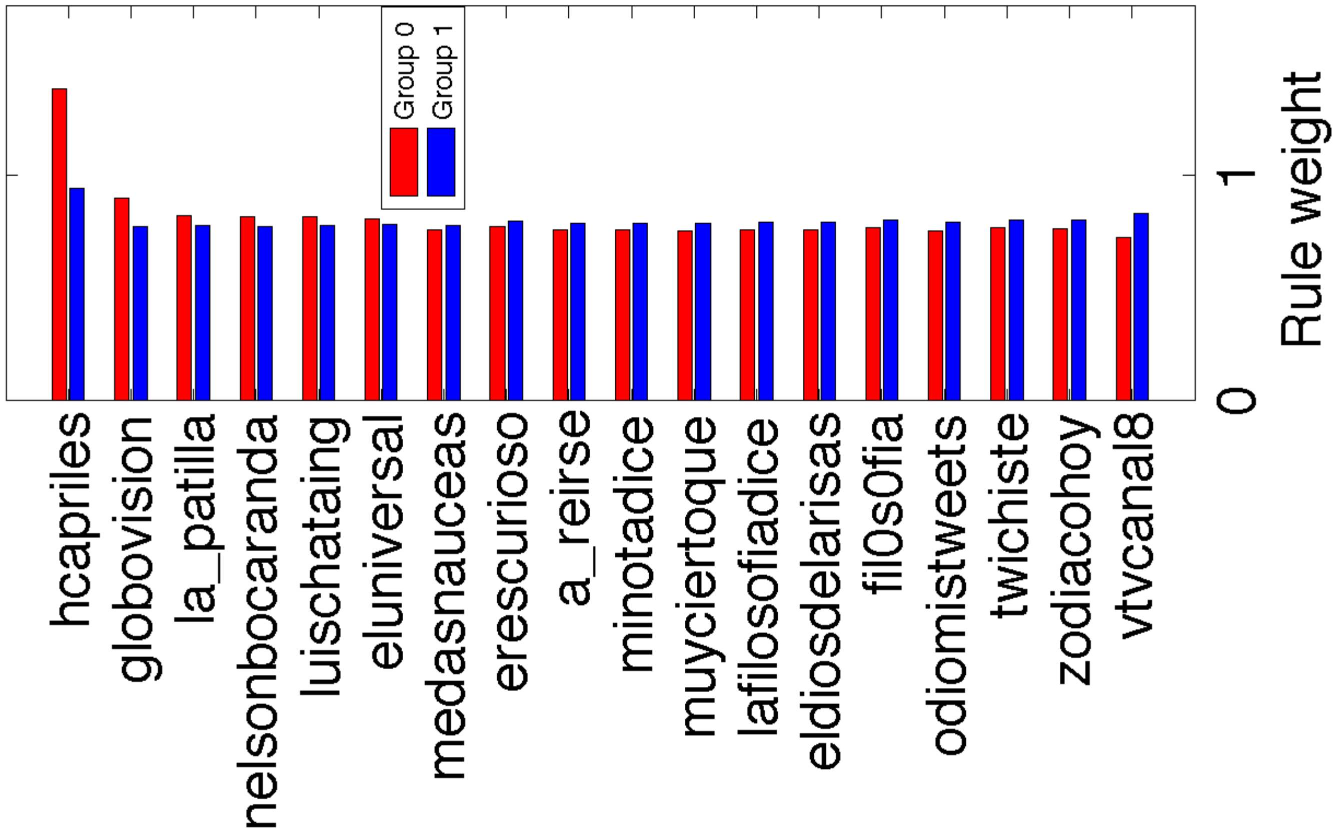
Learning Latent Groups

- South American tweets collected from 48-hour window around election.
- Selected 20 top users
  - Candidates, campaigns, media, and most retweeted
- 1,678 regular users interacted with at least one top user *and* used at least one hashtag in another tweet
- Those regular users had 8,784 interactions with non-top users

Learning Latent Groups



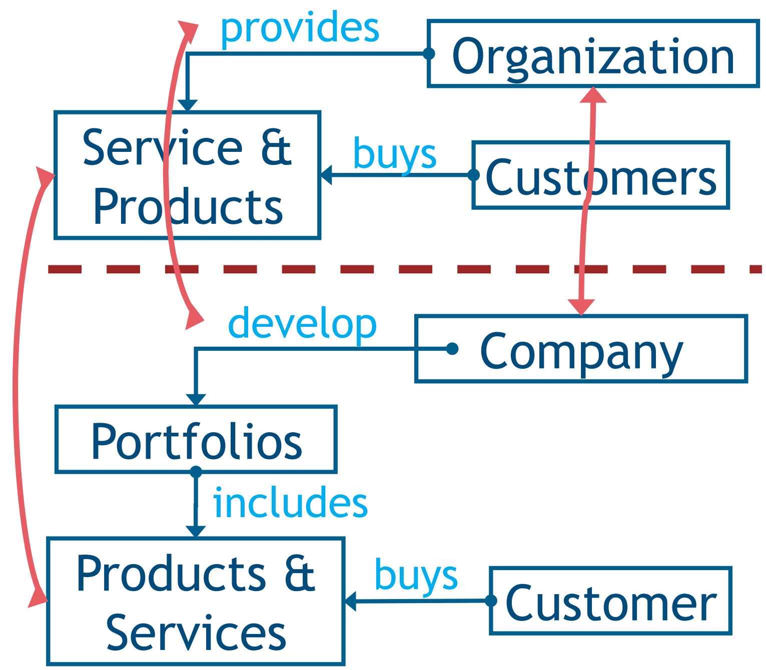
Learning Latent Groups



[Memory, Kimmig, Getoor, in prep]

Schema Matching

- Correspondences between source and target schemas
- Matching rules
  - ‘’ If two concepts are the same, they should have similar subconcepts’’
  - ‘’ If the domains of two attributes are similar, they may be the same’’

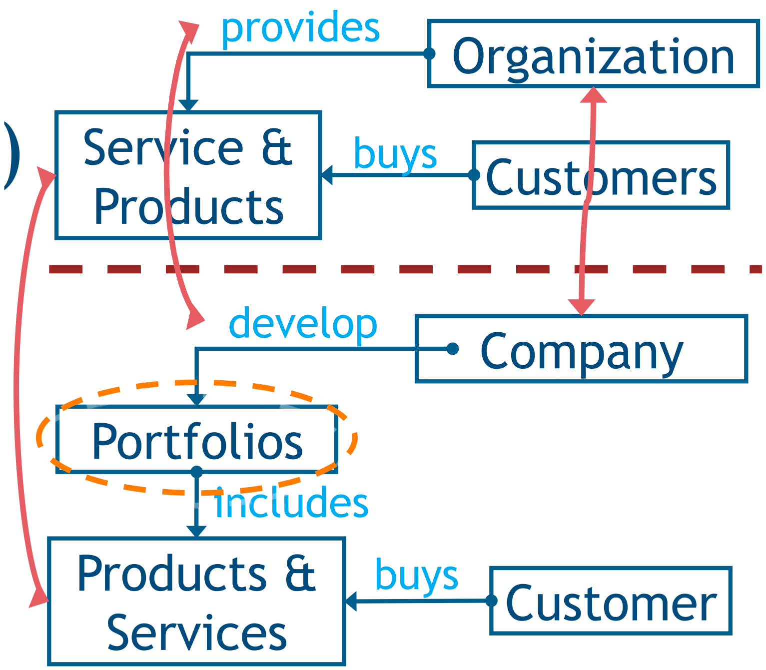


develop(A, B) <= provides(A, B)
Company(A) <= Organization(A)
Products&Services(B) <= Service&Products(B)

[Memory, Kimmig, Getoor, in prep]

Schema Mapping

- Input: Schema matches
- Output: S-T query pairs (TGD) for exchange or mediation
- Mapping rules
  - "Every matched attribute should participate in some TGD."
  - "The solutions to the queries in TGDs should be similar."



$\exists$Portfolio P, develop(A, P) $\wedge$ includes(P, B) <= provides(A, B) . . .

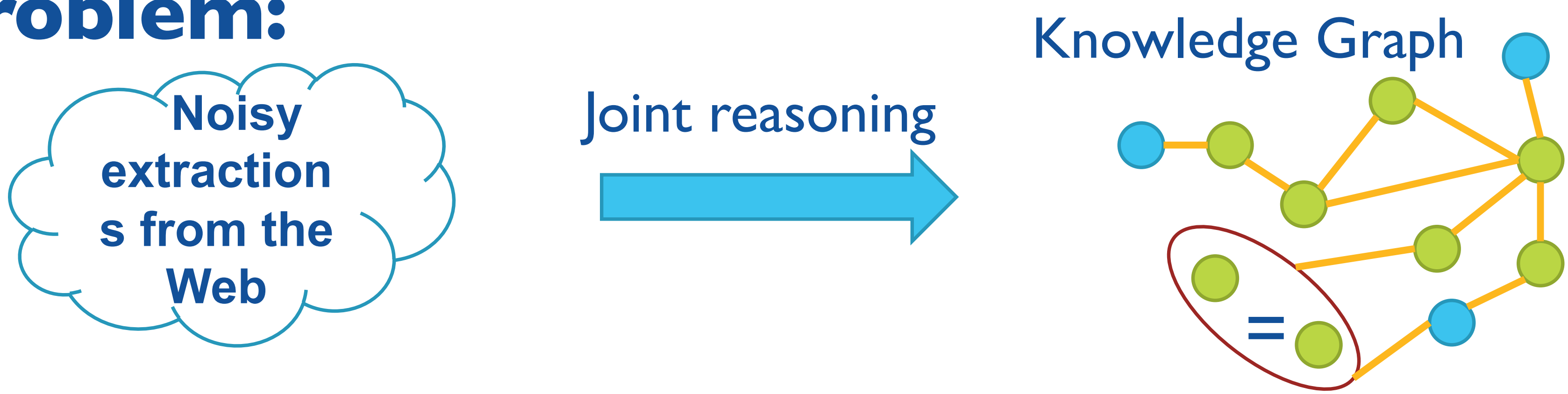
Pujara, Miao, Getoor, Cohen, ISWC 2013

Knowledge Graph Identification

- **Problem**: Collectively reason about noisy, inter-related fact extractions
- **Task**: NELL fact-promotion (web-scale IE)
  - Millions of extractions, with entity ambiguity and confidence scores
  - Rich ontology: Domain, Range, Inverse, Mutex, Subsumption
- **Goal**: Determine which facts to include in NELL's knowledge base

Knowledge Graph Identification

Problem:



Solution: *Knowledge Graph Identification* (KGI)

- Performs *graph identification*:
  - entity resolution
  - collective classification
  - link prediction
- Enforces *ontological constraints*
- Incorporates *multiple uncertain sources*

Graph Identification in KGI

Noisy Extractions:

$$\textsc{CandRel}_T(E_1, E_2, R) \overset{w_{CR_T}}{\Longrightarrow} \textsc{Rel}(E_1, E_2, R)$$

$$\textsc{CandLbl}_T(E, L) \overset{w_{CL_T}}{\Longrightarrow} \textsc{Lbl}(E, L)$$

$$\textsc{SameEnt}(E_1, E_2) \tilde{\wedge} \textsc{Lbl}(E_1, L) \Rightarrow \textsc{Lbl}(E_2, L)$$

$$\textsc{SameEnt}(E_1, E_2) \tilde{\wedge} \textsc{Rel}(E_1, E, R) \Rightarrow \textsc{Rel}(E_2, E, R)$$

$$\textsc{SameEnt}(E_1, E_2) \tilde{\wedge} \textsc{Rel}(E, E_1, R) \Rightarrow \textsc{Rel}(E, E_2, R)$$

KGI Representation of Ontological Rules

$$\textsc{Dom}(R, L) \,\tilde{\wedge}\, \textsc{Rel}(E_1, E_2, R) \implies \textsc{Lbl}(E_1, L)$$

$$\textsc{Rng}(R, L) \,\tilde{\wedge}\, \textsc{Rel}(E_1, E_2, R) \implies \textsc{Lbl}(E_2, L)$$

$$\textsc{Inv}(R, S) \,\tilde{\wedge}\, \textsc{Rel}(E_1, E_2, R) \implies \textsc{Rel}(E_2, E_1, R)$$

$$\textsc{Sub}(L, P) \,\tilde{\wedge}\, \textsc{Lbl}(E, L) \implies \textsc{Lbl}(E, P)$$

$$R\textsc{Sub}(R, S) \,\tilde{\wedge}\, \textsc{Rel}(E_1, E_2, R) \implies \textsc{Rel}(E_1, E_2, S)$$

$$\textsc{Mut}(L_1, L_2) \,\tilde{\wedge}\, \textsc{Lbl}(E, L_1) \implies \neg\textsc{Lbl}(E, L_2)$$

$$R\textsc{Mut}(R_1, R_2) \,\tilde{\wedge}\, \textsc{Rel}(E_1, E_2, R) \implies \neg\textsc{Rel}(E_1, E_2, R_2)$$

Adapted from Jiang et al., ICDM 2012

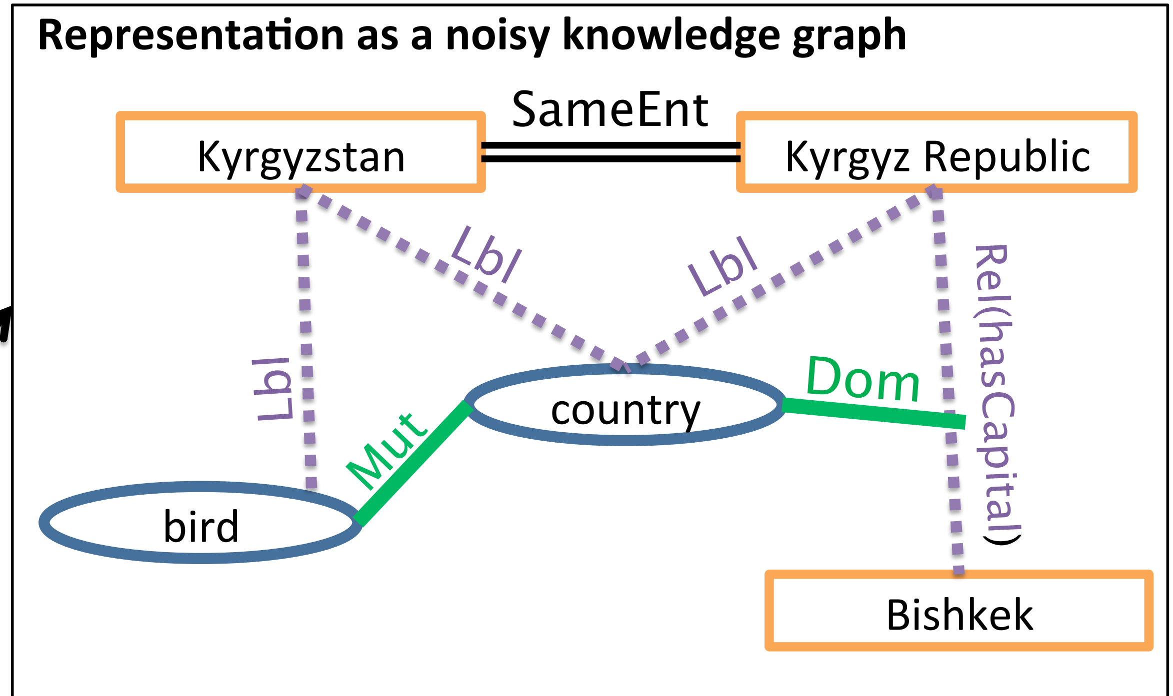
Illustration of KGI

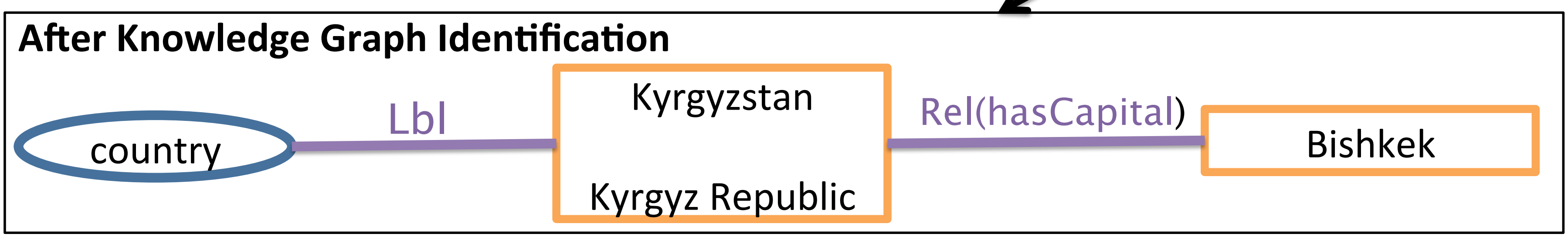
Extractions:
Lbl(Kyrgyzstan, bird)
Lbl(Kyrgyzstan, country)
Lbl(Kyrgyz Republic, country)

Rel(Kyrgyz Republic, Bishkek, hasCapital)

Ontology:
Dom(hasCapital, country)
Mut(country, bird)

Entity Resolution:
SameEnt(Kyrgyz Republic, Kyrgyzstan)





Pujara, Miao, Getoor, Cohen, ISWC 2013

Datasets & Results

- Evaluation on NELL dataset from iteration 165:
  - 1.7M candidate facts
  - 70K ontological constraints
- Predictions on 25K facts from a 2-hop neighborhood around test data
- Beats other methods, runs in just 10 seconds!
- Also supports lazy inference of complete knowledge graph (100 minutes)

	F1	AUC
Baseline	.828	.873
NELL	.673	.765
MLN (Jiang, 12)	.836	.899
KGI-PSL	**.853**	**.904**

Conclusion

Closing Comments

- Great opportunities to do good work and do useful things in the current era of big data, information overload and network science – ‘entity-oriented data science’
- Statistical relational learning provides some of the tools, much work still needed, developing theoretical bounds for relational learning, scalability, etc.
- Compelling applications abound!

Looking for students & postdocs

Probabilistic
Soft
Logic

psl.umiacs.umd.edu