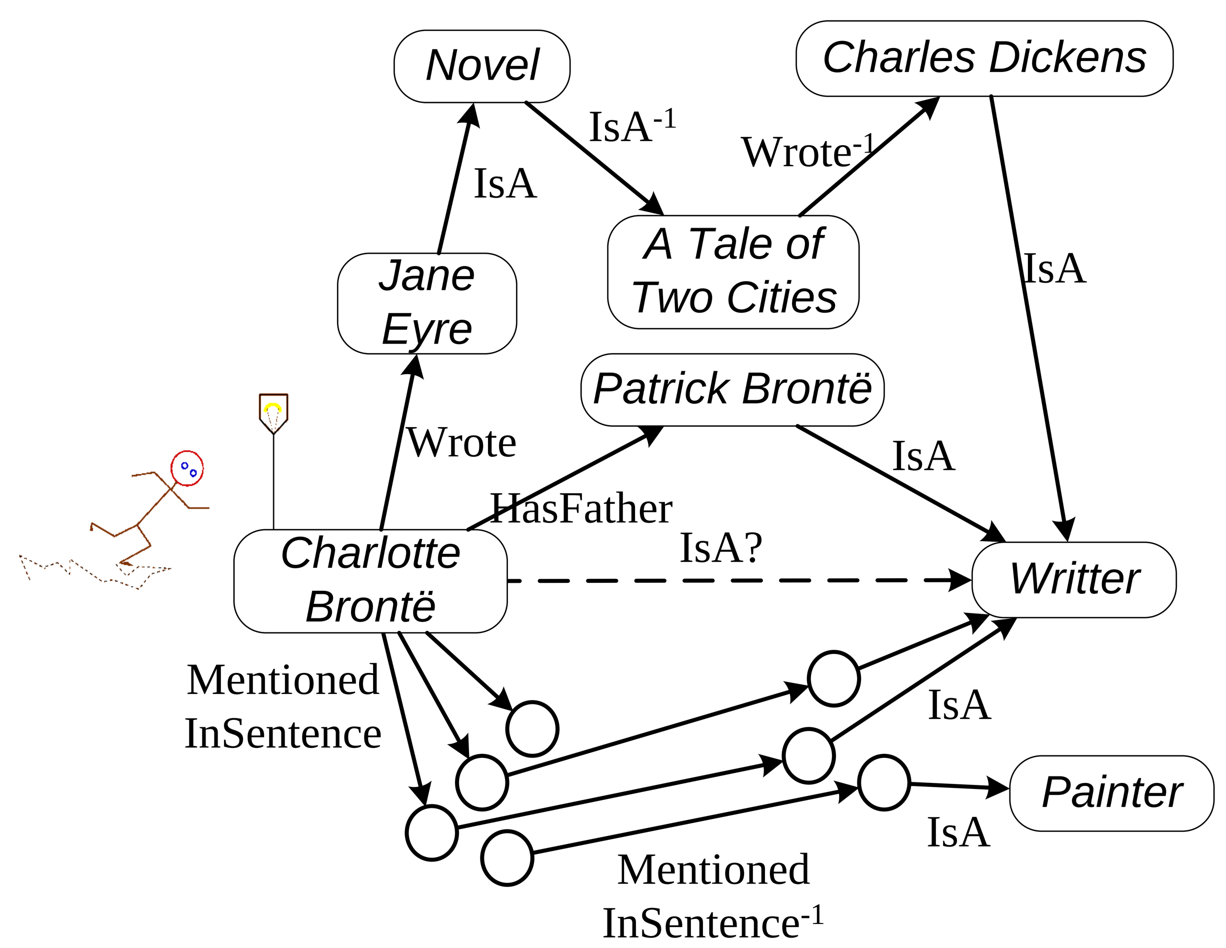
Scalable Inference over Large Knowledge Bases

Motivation

Relational Learning

Prediction with rich meta-data has great potentials and challenges

The world consists of objects and relations among them (e.g. family/friends, behavior, literature), but statistical learning tools (e.g. SVM) expect samples and their feature values



Random walk with restart

One-parameter-per-edge label is limited because the context of an edge label is ignored, e.g. Prob(Charlotte → Writer)

First Order Inductive Learner

Learn Horn clauses in first order logic (FOIL , 1993), e.g.

HasFather(a, b) ^ isa(b,y) → isa(a; y)

InSentence(a, j) ^ InSentence(b, j) ^ isa(b,y) → isa(a; y)

Lexicalized rules: HasFather(x, a) ^ isa(a,writer) → isa(x; writer)

Quantifier: ∃ i, Write(x, i) → isa(x; writer)

Horn clauses are costly to discover, inference is generally slow

Combine rules with disjunctions, therefore cannot leverage low accuracy rules

Random Walk Inference

Features generated by Path Constrained Random Walks

Prob(Charlotte → Writer| HasFather, isa)

Prob(Charlotte → Writer| Write, isa, isa^{-1}, Write, isa)

Prob(Charlotte → Writer| InSentence, InSentence^{-1} , isa)

Path Ranking Algorithm (PRA)

Model (Lao & Cohen, ECML 2010)

Scores source-target pairs by a linear function of their path features

$$score(s,t) = \sum_{P \in \mathbf{P}} \text{Prob}(s \to t \mid P)\theta_P$$

P is the set of all relation paths with length $\leq l$

Training

For a relation R and a set of node pairs $\{(s_i, t_i)\}$, create dataset D $=\{(x_i, y_i)\}$,

e.g. s_i →Charlotte, t_i →painter/writer

x_i is a vector of all the path features for (s_i, t_i), and

y_i indicates whether $R(s_i, t_i)$ is true or not

θ is estimated using L1,L2-regularized logistic regression

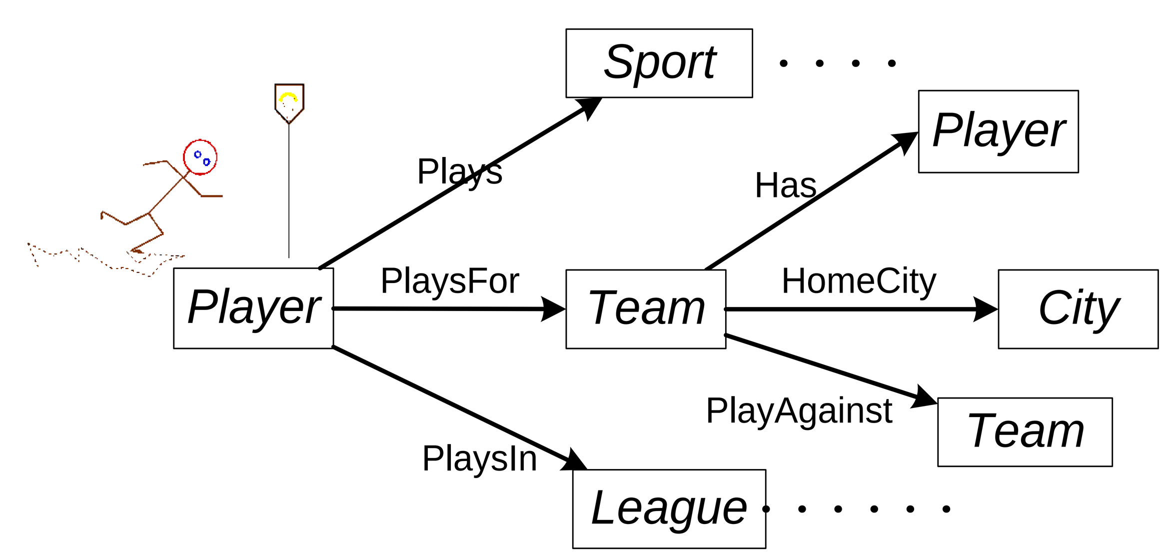
Path Finding & Feature Selection (L.M.C., EMNLP 2011)

Impractical to enumerate all possible edge sequences $O(|V|^L)$

Constraint 1: paths to instantiate in at least K(=5) training queries

Constraint 2: Prob(s→t| path, s→any node) > α (=0.2)

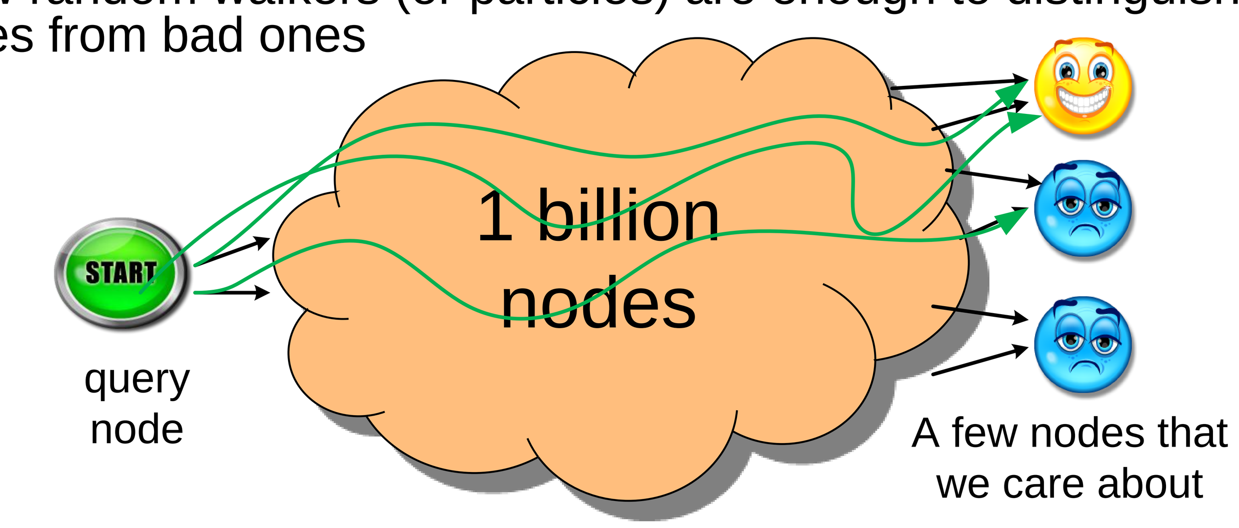
Depth first search up to length l: starts from a set of training queries, expand a node if the instantiation constraint is satisfied



Efficient Inference (Lao & Cohen, KDD 2010)

Exact calculation of random walk distributions results in non-zero probabilities for many internal nodes in the graph

A few random walkers (or particles) are enough to distinguish good target nodes from bad ones



Ni Lao, Tom Mitchell, William W. Cohen

{nlao, tom.mitchell, wcohen}@cs.cmu.edu

The NELL Case Study

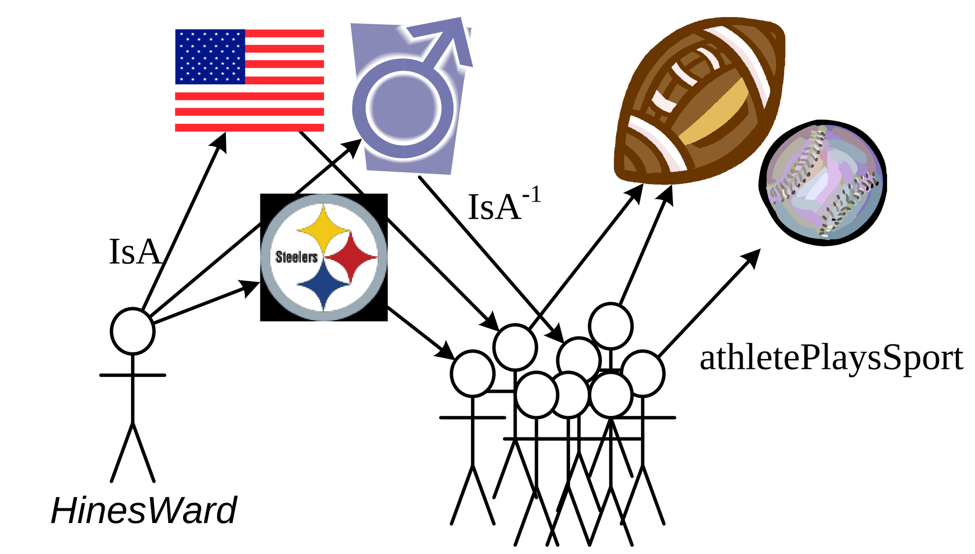
Never-Ending Language Learning

Combines multiple strategies: morphological patterns, textual context, html patterns, logical inference, etc.

Our task

Learn to infer new instances of 96 relations for which NELL database has more than 100 instances. Closed world assumption for training: the actual nodes y known to satisfy R(x; y) are treated as labeled positive examples, and all other nodes are treated as negative examples

Example features:



athletePlaysSport

$c \xrightarrow{isa} c \xrightarrow{isa^{-1}} c \xrightarrow{athletePlaysSport} c$

$c \xrightarrow{athletePlaysInLeague} c \xrightarrow{superpartOfOrganization} c \xrightarrow{teamPlaysSport} c$

Cross Validation on Training Queries

Table 3: Compare PRA with RWR models. MRRs and training times are averaged over 96 tasks.

	l=2		l=3	
	MRR	Training	MRR	Training
RWR(no train)	0.271		0.456	
RWR	0.280	3.7s	0.471	9.2s
PRA	0.307	5.7s	0.516	15.4s

Evaluation by Mechanical Turk

Evaluate the top ranked result for each query. Sort the predictions for each predicate by scores, and evaluate precisions at top 10, 100 and 1000 queries

	Functional Predicates				Non-functional Predicates			
	#Rules	p@10	p@100	p@1000	#Rules	p@10	p@100	p@1000
N-FOIL	2.1(+37)	0.76	0.380	0.007	----	----	----	----
PRA	43	0.79	**0.668**	**0.615**	92	**0.65**	**0.620**	**0.615**

Future Work

Efficiently discover long paths

Discover lexicalized paths (contains constant nodes)

Generalize relation paths to trees/networks

Carnegie Mellon