Machine Learning 10-601

Tom M. Mitchell
Machine Learning Department
Carnegie Mellon University

November 20, 2017

Today:

- Semi-supervised learning
- Co-Training
- Never ending learning

Recommended reading:

- See final slide

When can Unlabeled Data improve Supervised learning?

Important question! In many cases, unlabeled data is plentiful, labeled data expensive

- Medical outcomes (x=<symptoms,treatment>, y=outcome)
- Text classification (x=document, y=relevance)
- Customer modeling (x=user actions, y=user intent)
- Sensor interpretation (x=<video,audio>, y=who's there)

When can Unlabeled Data help supervised learning?

Problem setting (the PAC learning setting):

- Set *X* of instances drawn from unknown distribution *P(X)*
- Wish to learn target function *f: X*→ *Y* (or, P(Y|X))
- Given a set *H* of possible hypotheses for *f*

Given:

- i.i.d. labeled examples $L = \{\langle x_1, y_1 \rangle \ldots \langle x_m, y_m \rangle\}$
- i.i.d. unlabeled examples $U = \{x_{m+1}, \ldots x_{m+n}\}$

Wish to find hypothesis with lowest true error:

$$\hat{f} \leftarrow \arg\min_{h \in H} \Pr_{x \in P(X)} [h(x) \neq f(x)]$$

Note unlabeled data helps us estimate P(X)

Idea 1: Use U to reweight labeled examples

- Most learning algorithms minimize *errors over labeled examples*

- But we really want to minimize *true error*

true error

$$\hat{f} \leftarrow \arg\min_{h \in H} \Pr_{x \in P(X)} [h(x) \neq f(x)]$$

- If we know the underlying distribution P(X), we could weight each labeled training example <x,y> by its probability according to P(X=x)

- Unlabeled data allows us to estimate P(X)

Idea 1: Use U to reweight labeled examples L

Use $U \rightarrow \hat{P}(X)$ to alter the loss function

- Wish to minimize true error:

$$\hat{f} \leftarrow \underset{h \in H}{\operatorname{argmin}} \sum_{x \in X} \delta(h(x) \neq f(x)) P(x)$$

δ(): if its argument is true, then 1, else 0

- Usually we approximate this by training error:

$$\hat{f} \leftarrow \underset{h \in H}{\operatorname{argmin}} \frac{1}{|L|} \sum_{\langle x,y \rangle \in L} \delta(h(x) \neq y)$$

Which equals:

$$\hat{f} \leftarrow \underset{h \in H}{\operatorname{argmin}} \sum_{\langle x,y \rangle \in L} \delta(h(x) \neq y) \left[\frac{n(x,L)}{|L|} \right]$$

n(x,L) = number of times x occurs in L

empirical distr.

- U allows producing a better approximation to P(x):

$$\hat{f} \leftarrow \underset{h \in H}{\operatorname{argmin}} \sum_{\langle x,y \rangle \in L} \delta(h(x) \neq y) \left[\frac{n(x,L) + n(x,U)}{|L| + |U|} \right]$$

Reweighting Labeled Examples

- Wish to find

$$\hat{f} \leftarrow \underset{h \in H}{\operatorname{argmin}} \sum_{\langle x,y \rangle \in L} \delta(h(x) \neq y) \left[\frac{n(x,L) + n(x,U)}{|L| + |U|} \right]$$

- Already have algorithm (e.g., decision tree learner) to find

$$\hat{f} \leftarrow \underset{h \in H}{\operatorname{argmin}} \sum_{\langle x,y \rangle \in L} \delta(h(x) \neq y)$$

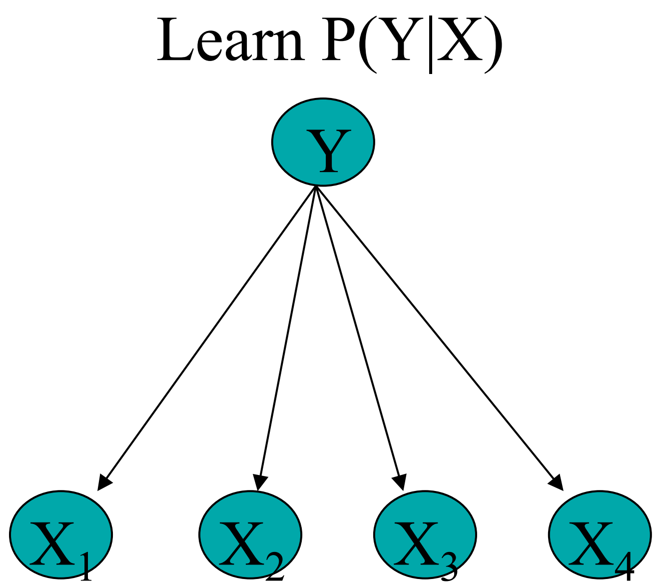
- Just reweight each <x,y> in L by $\left[\frac{n(x,L) + n(x,U)}{|L| + |U|} \right]$

- if X is continuous, may want to estimate p(X) in different way, still using L+U (e.g., density estimation)

$$\hat{f} \leftarrow \arg\min_{h \in H} \cancel{\frac{1}{L}} \sum_{\langle x,y \rangle \in L} \delta(h(x) \neq y)\, \widehat{p}(x)$$

Idea 2: Use Labeled and Unlabeled Data to Train Bayes Net for P(X,Y)

Idea 2: Use Labeled and Unlabeled Data to Train Bayes Net for P(X,Y), then infer P(Y|X)



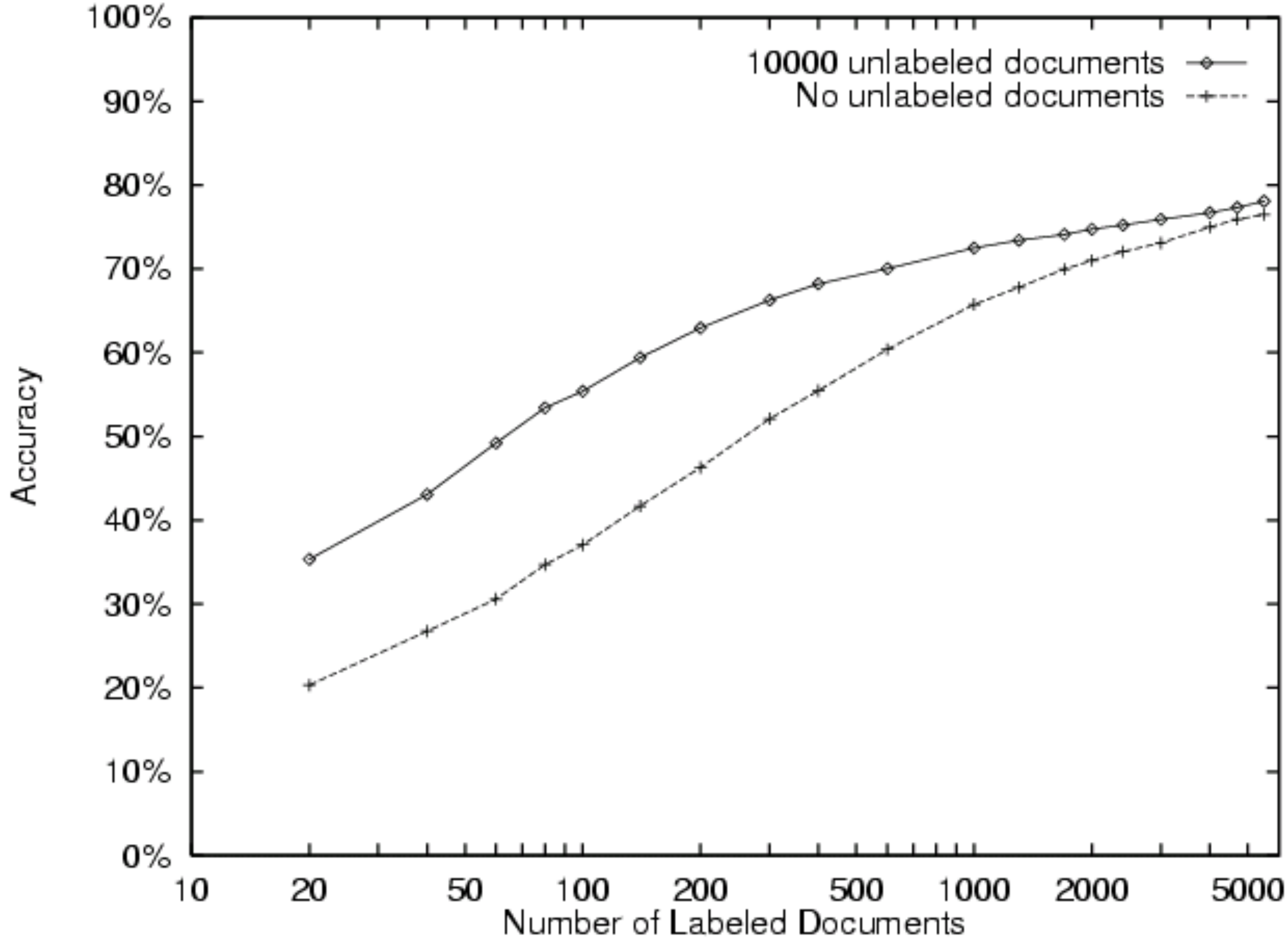
Y	X1	X2	X3	X4
1	0	0	1	1
0	0	1	0	0
0	0	0	1	0
?	0	1	1	0
?	0	1	0	1

EM: Train hypothesis h by repeating until convergence

E step: Apply h to assign probabilistic labels to unlabeled data

M step: Use observed plus probabilistic labels to train classifier h

20 Newsgroups



Summary : Semisupervised Learning with EM and Naïve Bayes Model

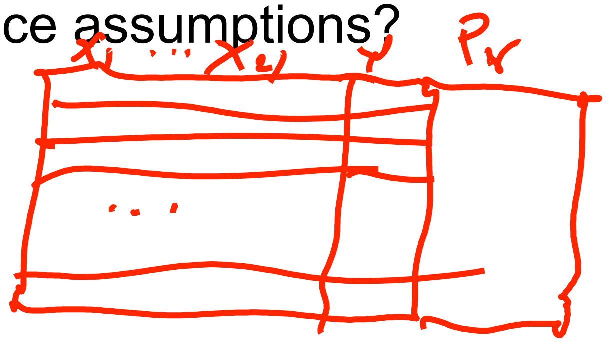
- If all data unlabeled, corresponds to unsupervised, mixture-of-multinomial clustering

$$P(x) = P(x \mid Y=0)\,P(Y=0) + P(x \mid Y=1)\,P(Y=1)$$

- If both labeled and unlabeled data, then unlabeled data helps if the Bayes net modeling assumptions are correct (e.g., P(X) is a mixture of class-conditional multinomials with conditionally independent X_i)

- Of course we could use Bayes net models other than Naïve Bayes

- Can unlabeled data be useful even if Bayes net makes no conditional independence assumptions?

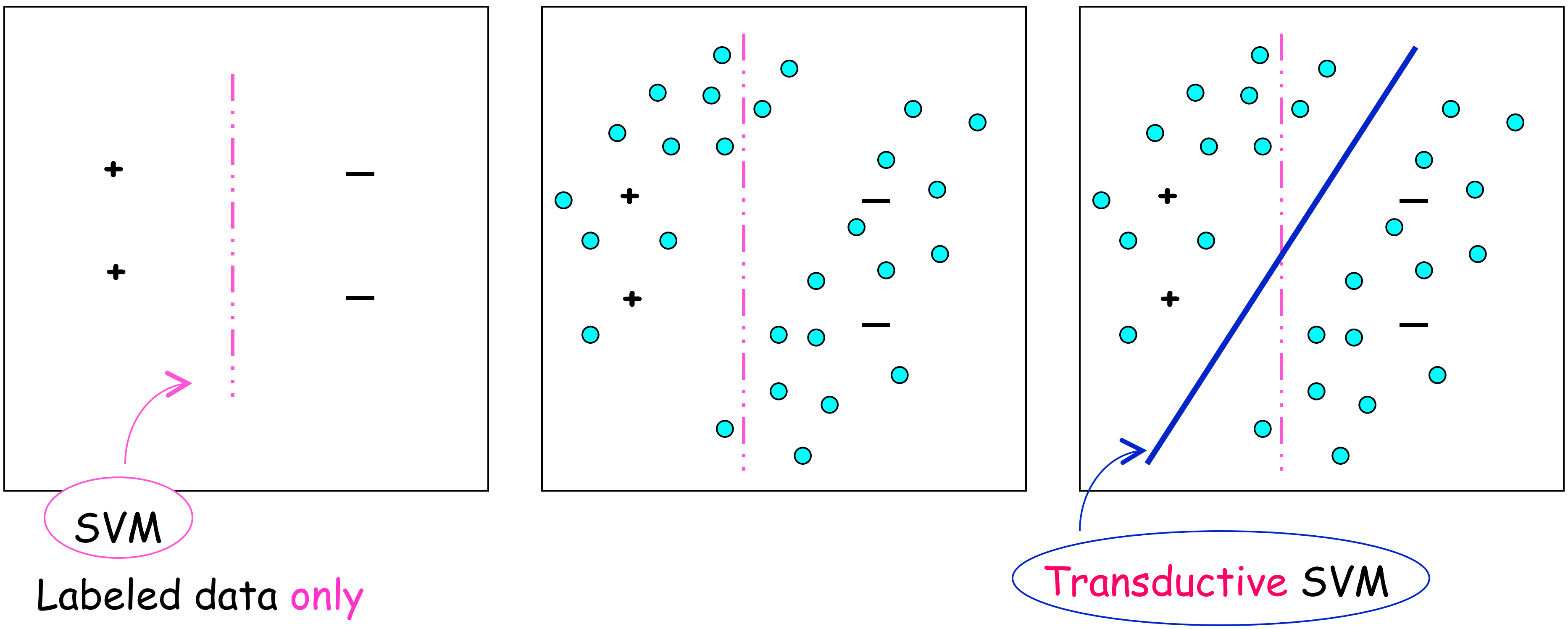


Idea 2.5: Similarly Use Labeled and Unlabeled Data to Train Support Vector Machine

Margins based regularity

Target goes through low density regions (large margin).

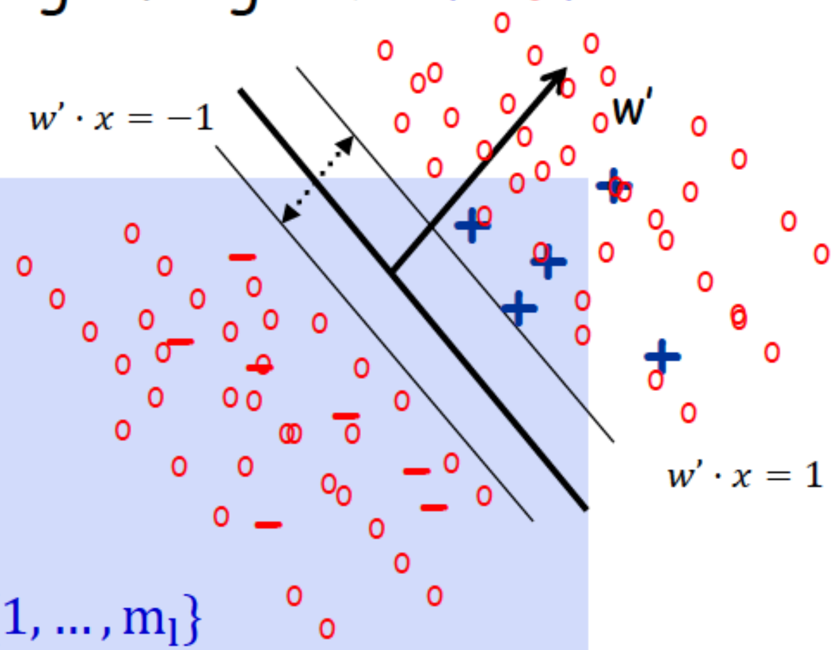
- assume we are looking for linear separator
- belief: should exist one with large separation



[courtesy of Maria-Florina Balcan]

Transductive Support Vector Machines

Optimize for the separator with large margin wrt labeled and unlabeled data. [Joachims '99]



Input: $S_l=\{(x_1,y_1), \dots,(x_{m_l},y_{m_l})\}$

$S_u=\{x_1, \dots,x_{m_u}\}$

$\mathrm{argmin}_w \ ||w||^2$ s.t.:

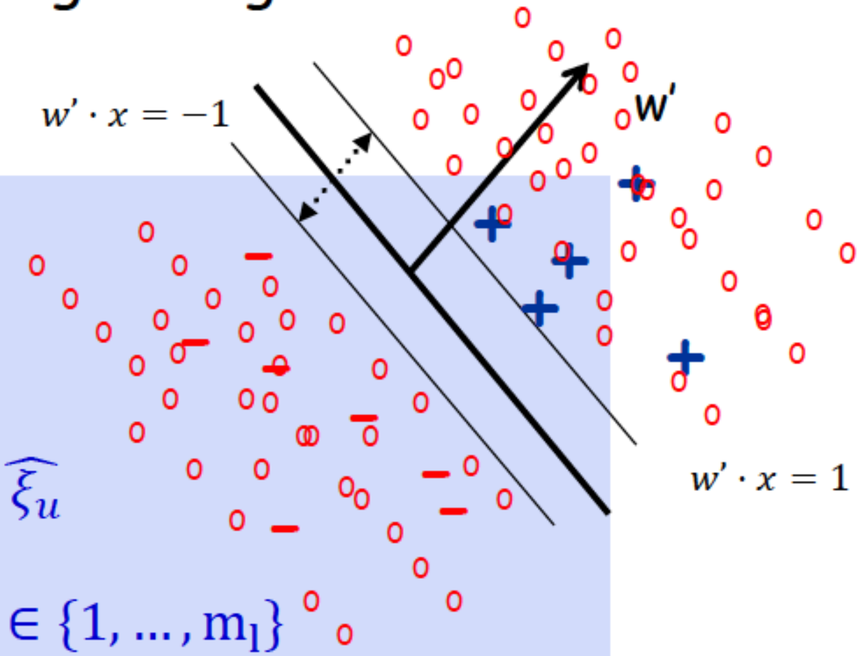
- $y_i \, w \cdot x_i \geq 1$, for all $i \in \{1, \dots, m_l\}$
- $\widehat{y_u} w \cdot x_u \geq 1$, for all $u \in \{1, \dots, m_u\}$
- $\widehat{y_u} \in \{-1, 1\}$ for all $u \in \{1, \dots, m_u\}$

Find a labeling of the unlabeled sample and w s.t. w separates both labeled and unlabeled data with maximum margin.

[courtesy Maria-Florina Balcan]

Transductive Support Vector Machines

Optimize for the separator with large margin wrt labeled and unlabeled data. [Joachims '99]



<u>Input</u>: $S_l=\{(x_1, y_1), \ldots, (x_{m_l}, y_{m_l})\}$

$S_u=\{x_1, \ldots, x_{m_u}\}$

$$\mathrm{argmin}_w \; ||w||^2 + C \sum_i \xi_i + C \sum_u \widehat{\xi_u}$$

- $y_i \, w \cdot x_i \geq 1 - \xi_i$, for all $i \in \{1, \ldots, m_l\}$
- $\widehat{y_u} w \cdot x_u \geq 1 - \widehat{\xi_u}$, for all $u \in \{1, \ldots, m_u\}$
- $\widehat{y_u} \in \{-1, 1\}$ for all $u \in \{1, \ldots, m_u\}$

Find a labeling of the unlabeled sample and w s.t. w separates both labeled and unlabeled data with maximum margin.

[Courtesy Maria-Florina Balcan]

Transductive Support Vector Machines

Optimize for the separator with large margin wrt labeled and unlabeled data.

<u>Heuristic (Joachims) high level idea:</u>

- First maximize margin over the labeled points
- Use this to give initial labels to unlabeled points based on this separator.
- Try flipping labels of unlabeled points to see if doing so can increase margin

Keep going until no more improvements. Finds a locally-optimal solution.

Idea 3: CoTraining, Coupled Training

- When learning f: X → Y, sometimes available features of X are redundantly sufficient to predict Y.   We can then train two classifiers based on disjoint subsets of X

- Of course these two classifiers should agree on the classification for each unlabeled example

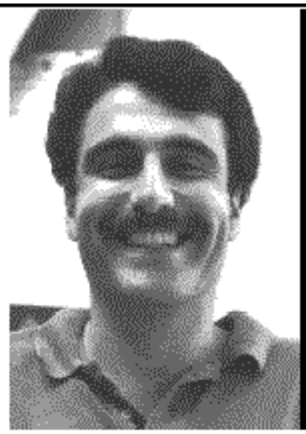
- Therefore, we can use the unlabeled data to constrain joint training of both classifiers

Redundantly Sufficient Features

hyperlink

Professor Faloutsos

my advisor



U.S. mail address:
Department of Computer Science
University of Maryland
College Park, MD 20742
(97-99: on leave at CMU)
Office: 3227 A.V. Williams Bldg.
Phone: (301) 405-2695
Fax: (301) 405-6707
Email: christos@cs.umd.edu

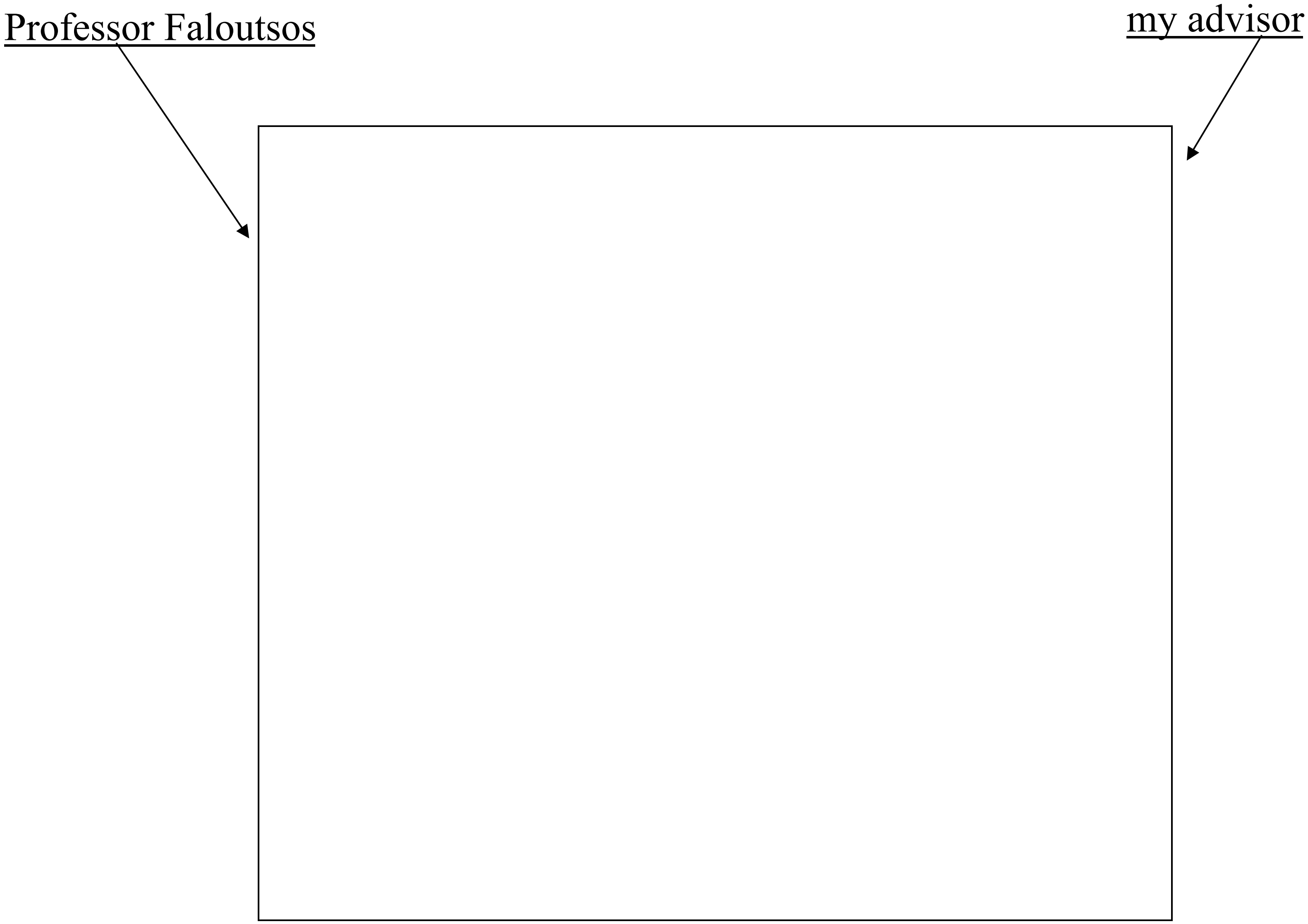
Christos Faloutsos

Current Position: Assoc. Professor of Computer Science. (97-98: on leave at CMU)
Join Appointment: Institute for Systems Research (ISR).
Academic Degrees: Ph.D. and M.Sc. (University of Toronto.); B.Sc. (Nat. Tech. U. Ath

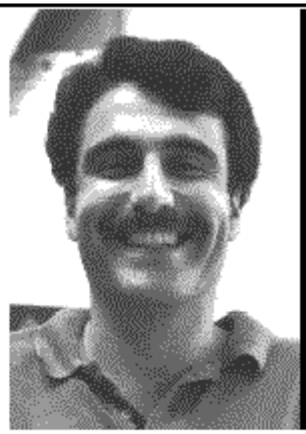
Research Interests:

- Query by content in multimedia databases;
- Fractals for clustering and spatial access methods;
- Data mining;

Redundantly Sufficient Features



Redundantly Sufficient Features



U.S. mail address:
Department of Computer Science
University of Maryland
College Park, MD 20742
(97-99: on leave at CMU)
Office: 3227 A.V. Williams Bldg.
Phone: (301) 405-2695
Fax: (301) 405-6707
Email: christos@cs.umd.edu

Christos Faloutsos

Current Position: Assoc. Professor of Computer Science. (97-98: on leave at CMU)
Join Appointment: Institute for Systems Research (ISR).
Academic Degrees: Ph.D. and M.Sc. (University of Toronto.); B.Sc. (Nat. Tech. U. Ath

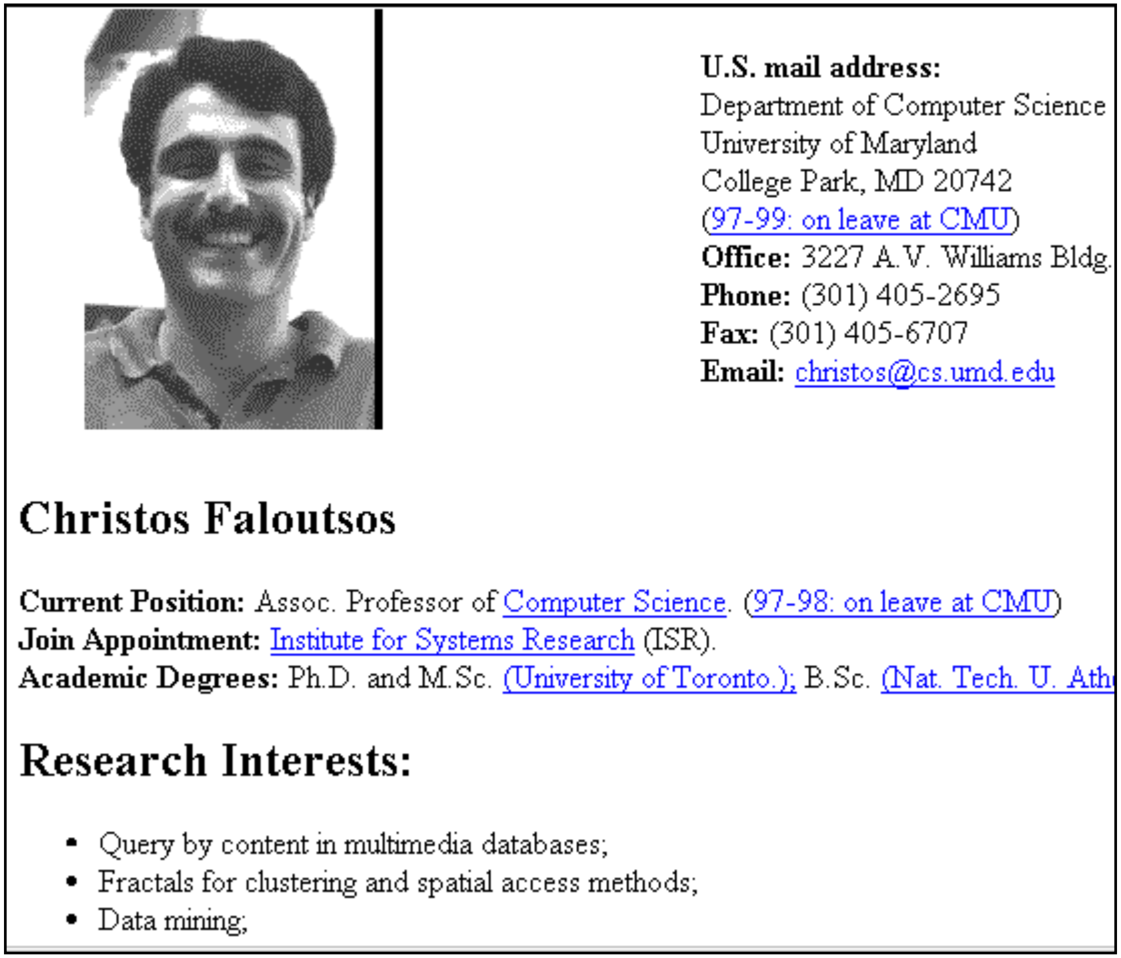
Research Interests:

- Query by content in multimedia databases;
- Fractals for clustering and spatial access methods;
- Data mining;

Redundantly Sufficient Features

Professor Faloutsos

my advisor



U.S. mail address:
Department of Computer Science
University of Maryland
College Park, MD 20742
(97-99: on leave at CMU)
Office: 3227 A.V. Williams Bldg.
Phone: (301) 405-2695
Fax: (301) 405-6707
Email: christos@cs.umd.edu

Christos Faloutsos

Current Position: Assoc. Professor of Computer Science. (97-98: on leave at CMU)
Join Appointment: Institute for Systems Research (ISR).
Academic Degrees: Ph.D. and M.Sc. (University of Toronto.); B.Sc. (Nat. Tech. U. Ath

Research Interests:

- Query by content in multimedia databases;
- Fractals for clustering and spatial access methods;
- Data mining;

CoTraining Algorithm #1

[Blum&Mitchell, 1998]

Given: labeled data L,

unlabeled data U

Loop:

Train g1 (hyperlink classifier) using L

Train g2 (page classifier) using L

Allow g1 to label *p* positive, *n* negative examps from U

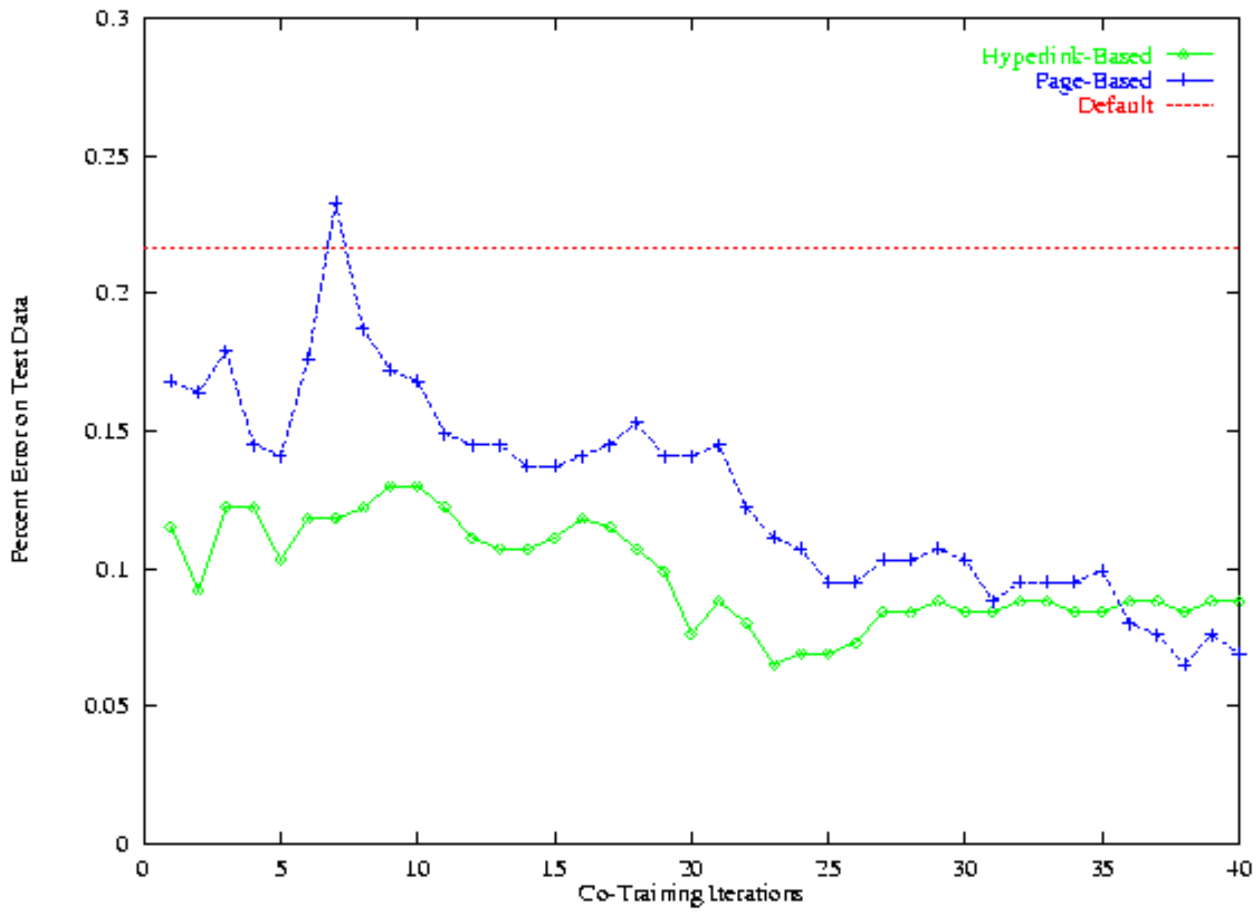
Allow g2 to label *p* positive, *n* negative examps from U

Add these self-labeled examples to L

CoTraining: Experimental Results

- begin with 12 labeled web pages (academic course pages)
- provide 1,000 additional unlabeled web pages
- average error: learning from labeled data 11.1%;
- average error: cotraining 5.0%

Typical run:



CoTraining setting:

- wish to learn f: X → Y, given L and U drawn from P(X)
- features describing X can be partitioned (X = X1 x X2)
  such that f can be computed from either X1 or X2

hyp

page itself

Classifier with accuracy > 0.5

One theoretical result [Blum&Mitchell 1998]:

- If
  - X1 and X2 are conditionally independent given Y
  - f is PAC learnable from polynomial number of noisy *labeled* examples
- Then
  - f is PAC learnable from <u>weak</u> initial classifier plus polynomial number of *unlabeled* examples

CoTraining Summary

- Unlabeled data improves supervised learning when example features are redundantly sufficient
  - Family of algorithms that train multiple classifiers

- Theoretical results
  - If X1,X2 conditionally independent given Y, Then
    - PAC learnable from weak initial classifier plus unlabeled data
    - disagreement between g1(x1) and g2(x2) bounds final classifier error

- Many real-world problems of this type
  - Semantic lexicon generation [Riloff, Jones 99], [Collins, Singer 99]
  - Web page classification [Blum, Mitchell 98]
  - Word sense disambiguation [Yarowsky 95]
  - Speech recognition [de Sa, Ballard 98]
  - Visual classification of cars [Levin, Viola, Freund 03]

What you should know

1. Using unlabeled data to reweight labeled examples gives better approximation to true error
   - If we assume examples drawn from fixed P(X)

2. Unlabeled can help EM learn Bayes nets for P(X,Y), and thus P(Y|X)
   - If we assume the Bayes net structure reflects cond. independencies

2.5. Transductive SVM's
   - If we assume maximizing margin captures relationship between P(X) and f: $X \rightarrow Y$

3. Jointly train multiple classifiers, coupled by consistency constraints that can be evaluated using unlabeled data
   - optimize both the fit to labeled examples, and satisfaction of the consistency constraints

Never Ending Language Learning

http://rtw.ml.cmu.edu

NELL: Never-Ending Language Learner

The task:

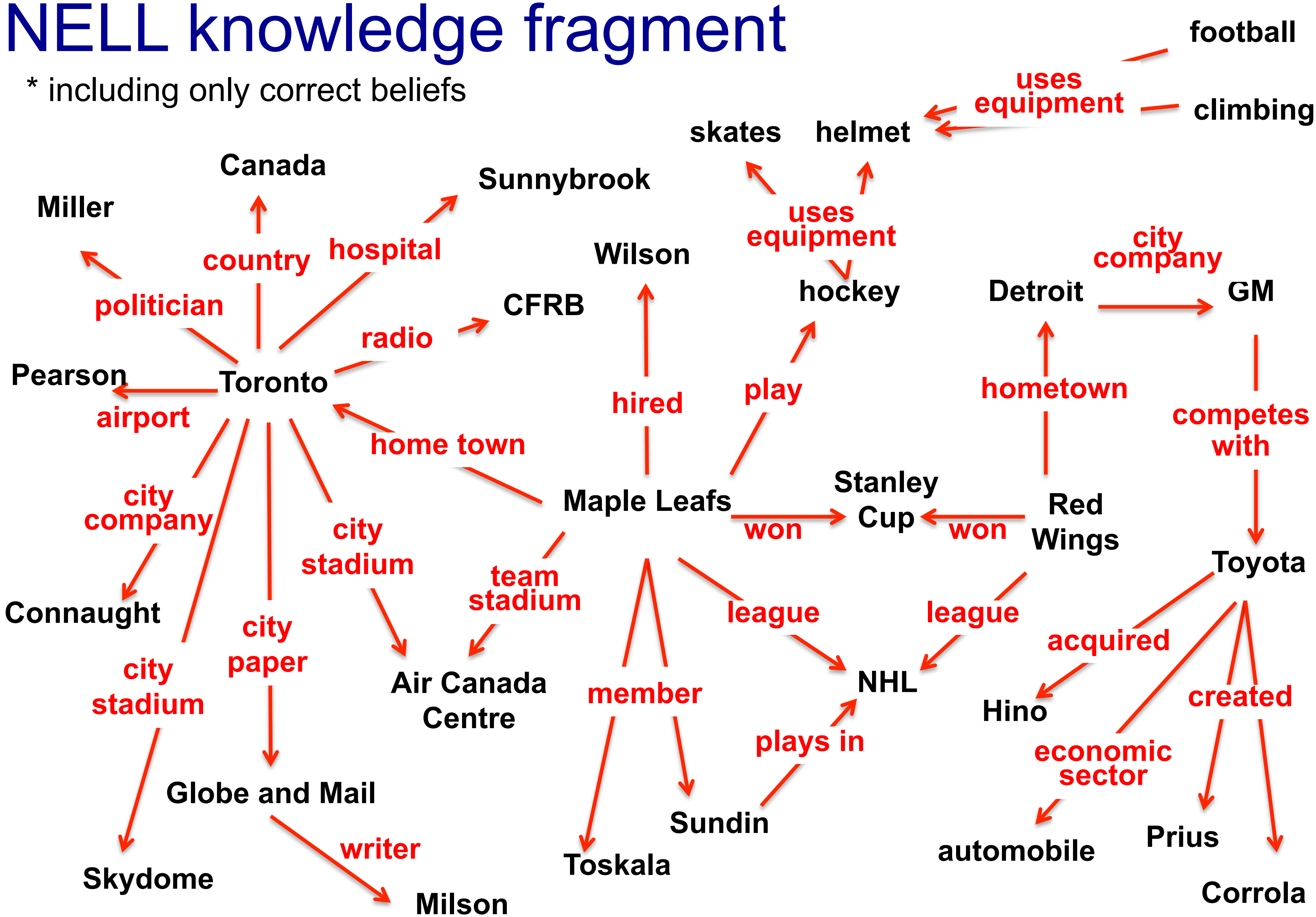
- run 24x7, forever
- each day:
    1. extract more facts from the web to populate the ontology
    2. learn to read (perform #1) better than yesterday

Inputs:

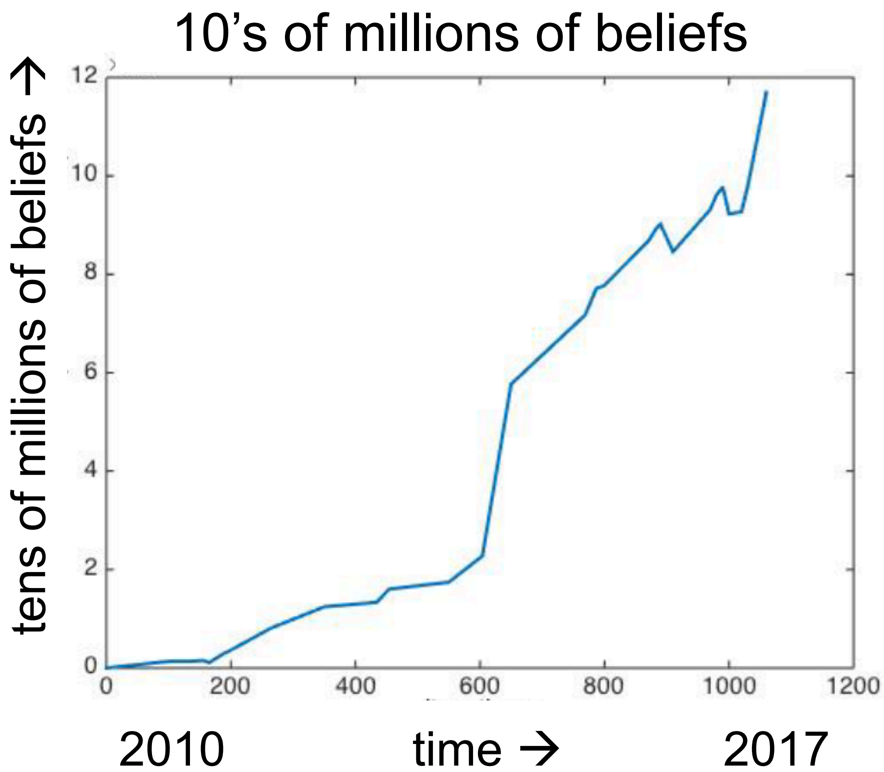
- initial ontology (categories and relations)
- dozen examples of each ontology predicate
- the web
- occasional interaction with human trainers

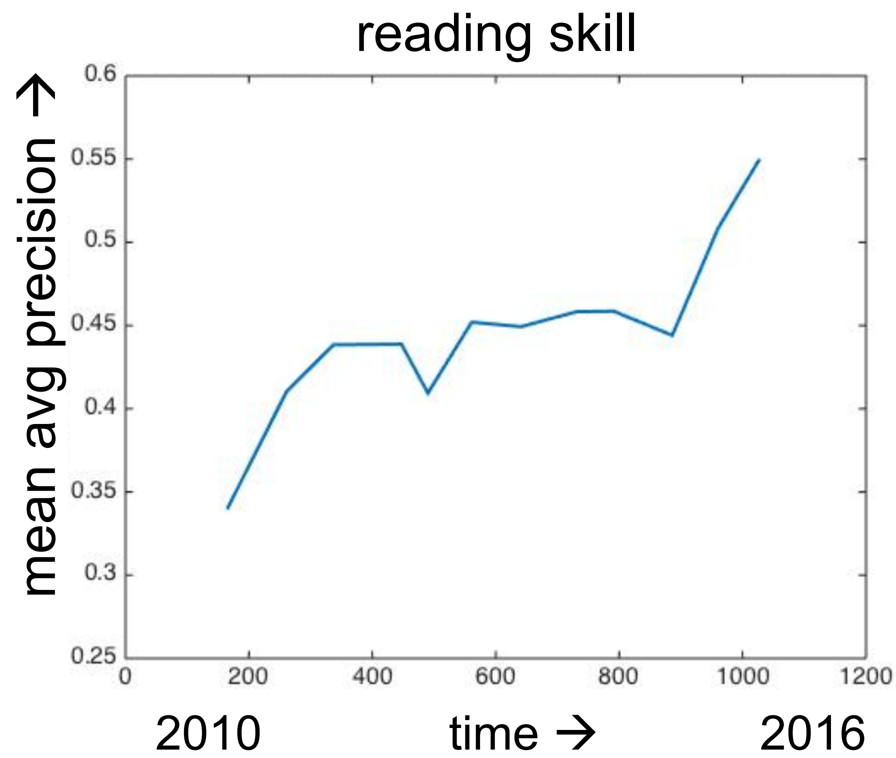
NELL knowledge fragment

* including only correct beliefs



Improving Over Time
Never Ending Language Learner





[Mitchell et al., CACM 2017]

Learning from Unlabeled Data in NELL

Coupled training of thousands of functions

Semi-Supervised Bootstrap Learning

it's underconstrained!!

Extract cities:

Paris
Pittsburgh
Seattle
Cupertino

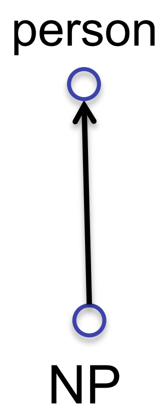
San Francisco
Austin
denial

anxiety
selfishness
Berlin

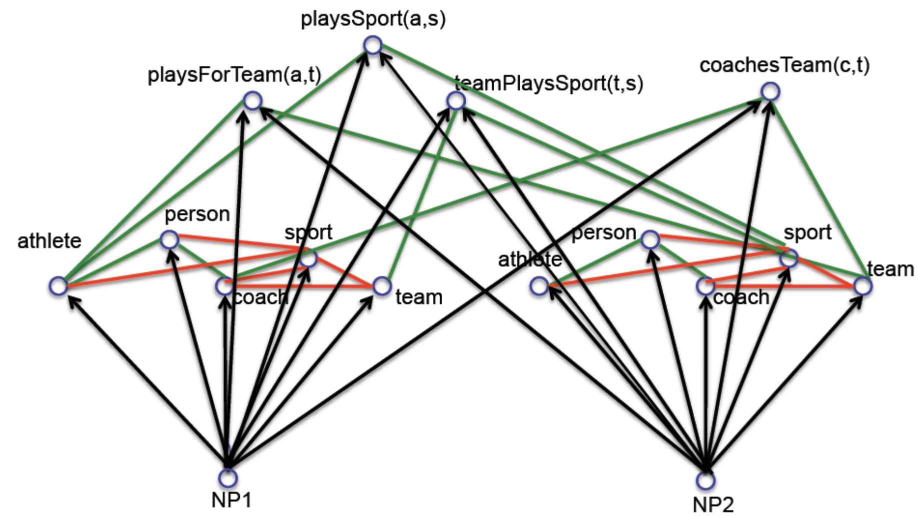
mayor of arg1
live in arg1

arg1 is home of
traits such as arg1

Key Idea 1: Coupled semi-supervised training of 1000's of functions



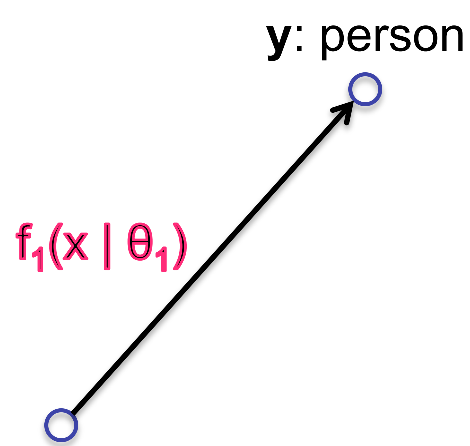
hard
(underconstrained)
semi-supervised
learning problem



much easier (more constrained)
semi-supervised learning problem

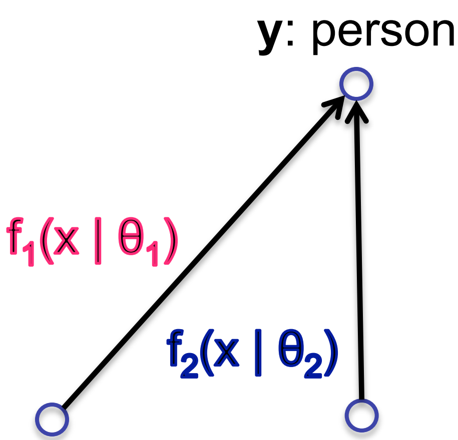
Supervised training of 1 function:

$$\theta_1 = \arg\min_{\theta_1} \sum_{\langle x,y \rangle \in \; labeled \; data} |f_1(x|\theta_1) - y|$$



x: NP context distribution

__ is a friend
rang the __
…
__ walked in



x: NP context distribution | NP morphology

__ is a friend
rang the __
…
__ walked in

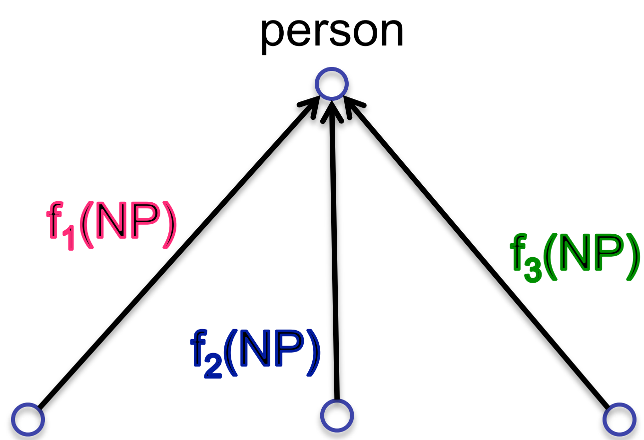
capitalized?
ends with '...ski'?
…
contains "univ."?

Coupled training of 2 functions:

$$\theta_1, \theta_2 = \arg\min_{\theta_1,\theta_2} \sum_{\langle x,y\rangle \,\in\, labeled\ data} |f_1(x|\theta_1) - y| + \sum_{\langle x,y\rangle \,\in\, labeled\ data} |f_2(x|\theta_2) - y| + \sum_{x \,\in\, unlabeled\ data} |f_1(x|\theta_1) - f_2(x|\theta_2)|$$

Type 1 Coupling: Co-Training, Multi-View Learning

[Blum & Mitchell; 98]
[Dasgupta et al; 01]
[Ganchev et al., 08]
[Sridharan & Kakade, 08]
[Wang & Zhou, ICML10]



NP:

NP text context distribution	NP morphology	NP HTML contexts
__ is a friend	*capitalized?*	*www.celebrities.com:*
rang the __	*ends with '...ski'?*	* __ *
...	...	
__ walked in	*contains "univ."?*	...

NELL Learned Contexts for “Hotel” (~1% of total)

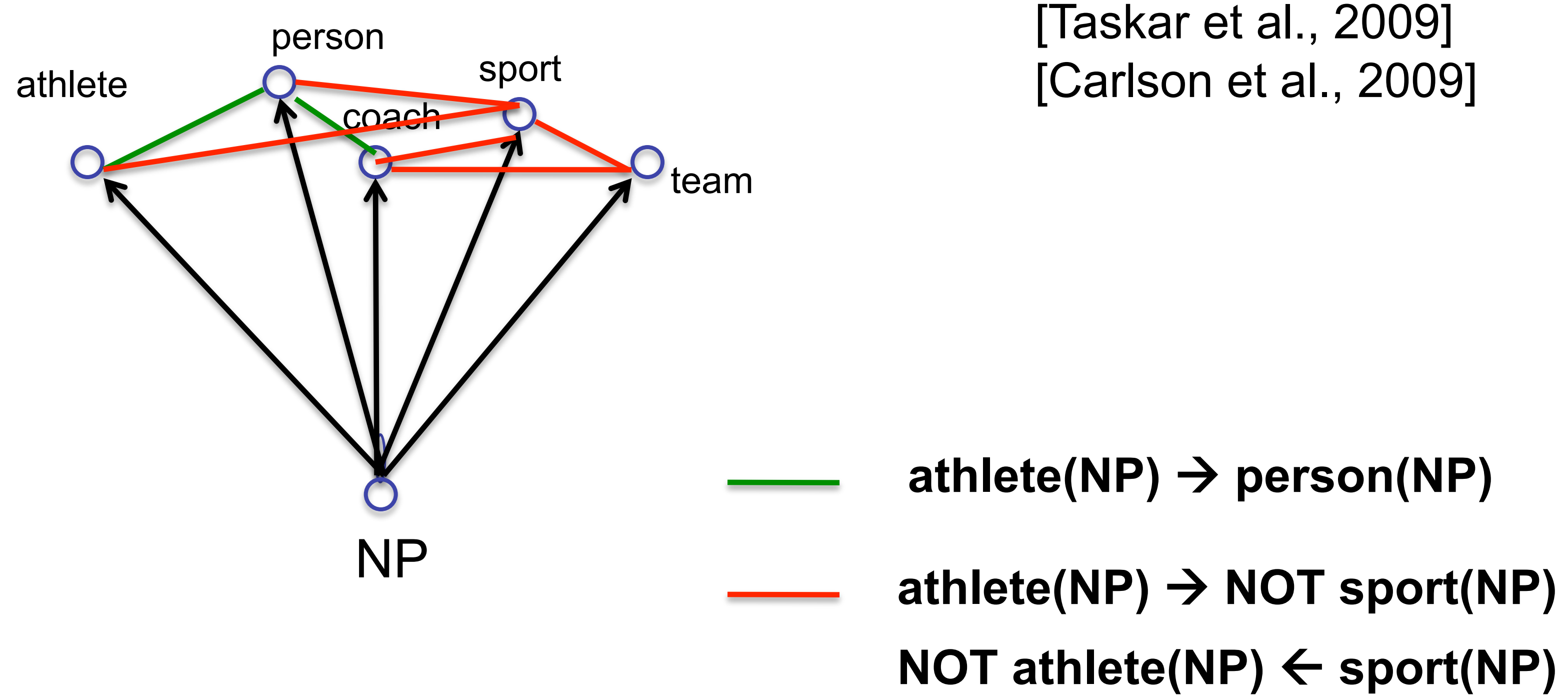
"_ is the only five-star hotel” "_ is the only hotel” "_ is the perfect accommodation" "_ is the perfect address" "_ is the perfect central location” "_ is the perfect extended stay hotel” "_ is the perfect headquarters” "_ is the perfect home base” "_ is the perfect lodging choice" "_ is the perfect lodging” "_ is the sister hotel” "_ is the ultimate hotel" "_ is the value choice” "_ is uniquely situated in” "_ is Walking Distance” "_ is wonderfully situated in” "_ las vegas hotel” "_ los angeles hotels" "_ maintains all ownership rights" "_ Make an online hotel reservation” "_ makes a great home-base” "_ mentions Downtown” "_ mette a disposizione” "_ miami south beach” "_ minded traveler” "_ mucha prague Map Hotel” "_ n'est qu'quelques minutes” "_ naturally has a pool” "_ north reddington beach” "_ now offer guests" "_ now offers guests” "_ occupies a privileged location” "_ occupies an ideal location" "_ offer a king bed” "_ offer a large bedroom” "_ offer a master bedroom” "_ offer a refrigerator” "_ offer a separate living area" "_ offer a separate living room” "_ offer comfortable rooms” "_ offer complimentary shuttle service” "_ offer deluxe accommodations” "_ offer family rooms” "_ offer secure online reservations” "_ offer upscale amenities" "_ offering a complimentary continental breakfast” "_ offering comfortable rooms” "_ offering convenient access" "_ offering great lodging" "_ offering luxury accommodation” "_ offering world class facilities” "_ offers a business center" "_ offers a business centre” "_ offers a casual elegance” "_ offers a central location” “_ surrounds travelers” …

NELL Highest Weighted string fragments: “Hotel”

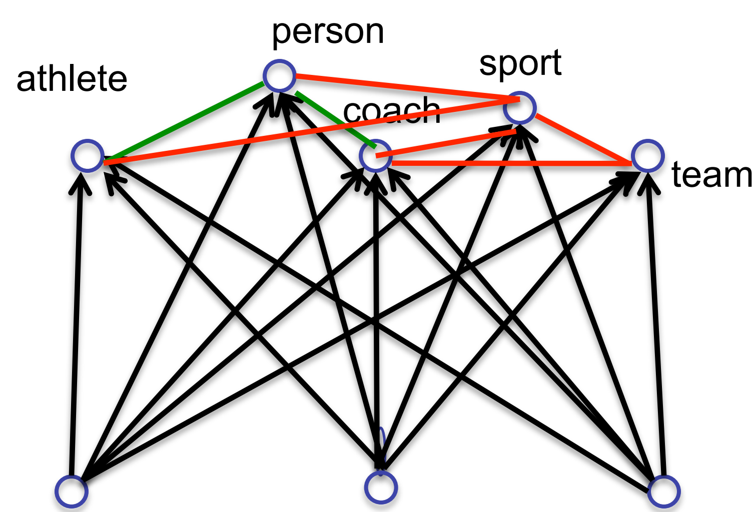
0.87944 SUFFIX=iott
0.88023 PREFIX=west
0.88297 SUFFIX=riott
0.92353 SUFFIX=yatt
0.93224 PREFIX=hyat
0.95354 PREFIX=marri
0.95574 PREFIX=marr
0.95585 FIRST_WORD=le
0.97019 SUFFIX=ites
1.00765 FIRST_WORD=the
1.02291 SUFFIX=ort
1.04229 PREFIX=resor
1.04476 FIRST_WORD=hilton
1.04524 SUFFIX=uites
1.06683 SUFFIX=odge
1.08925 PREFIX=hot
1.12714 PREFIX=hote
1.12796 PREFIX=in
1.43756 LAST_WORD=inn
1.81727 SUFFIX=otel
1.82307 SUFFIX=tel

Type 2 Coupling: Multi-task, Structured Outputs

[Daume, 2008]
[Bakhir et al., eds. 2007]
[Roth et al., 2008]
[Taskar et al., 2009]
[Carlson et al., 2009]

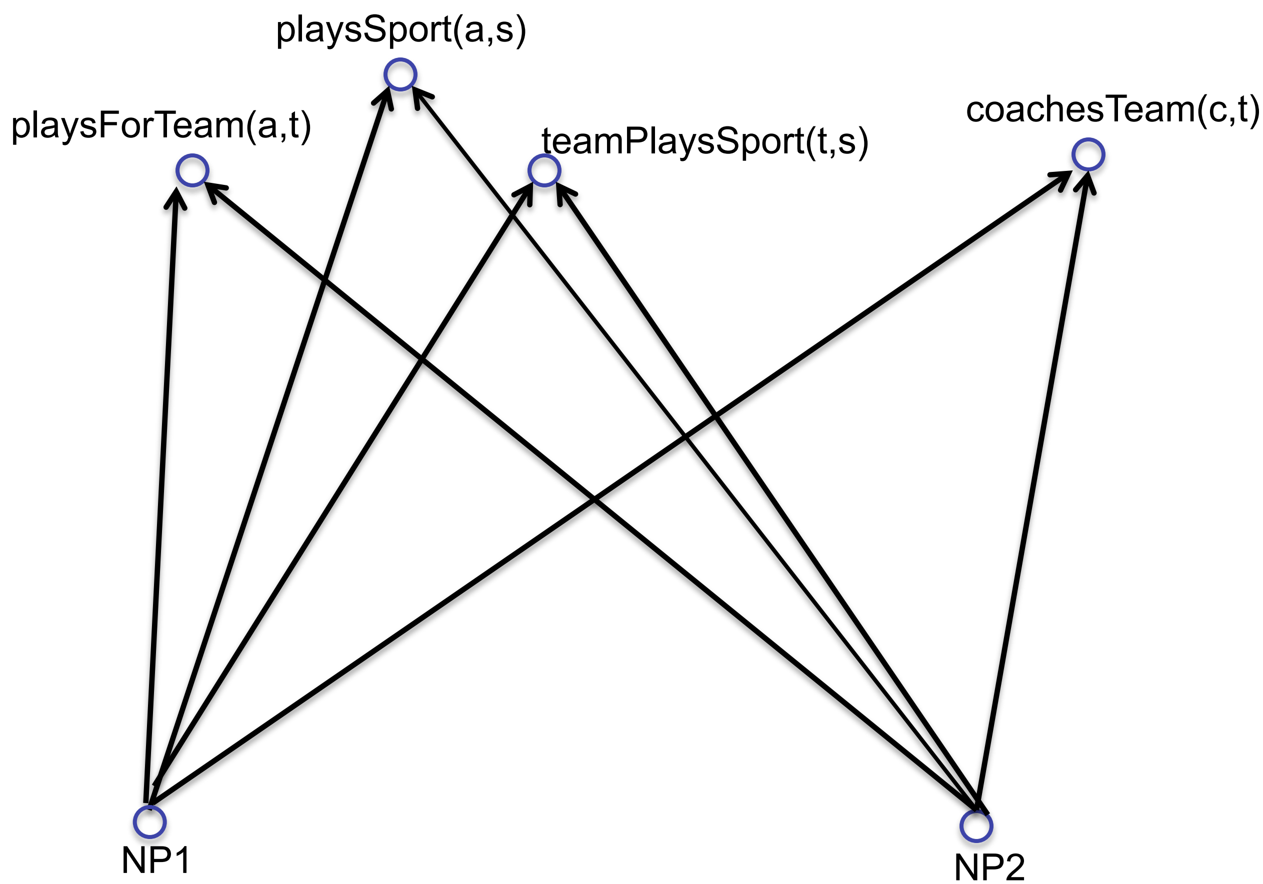


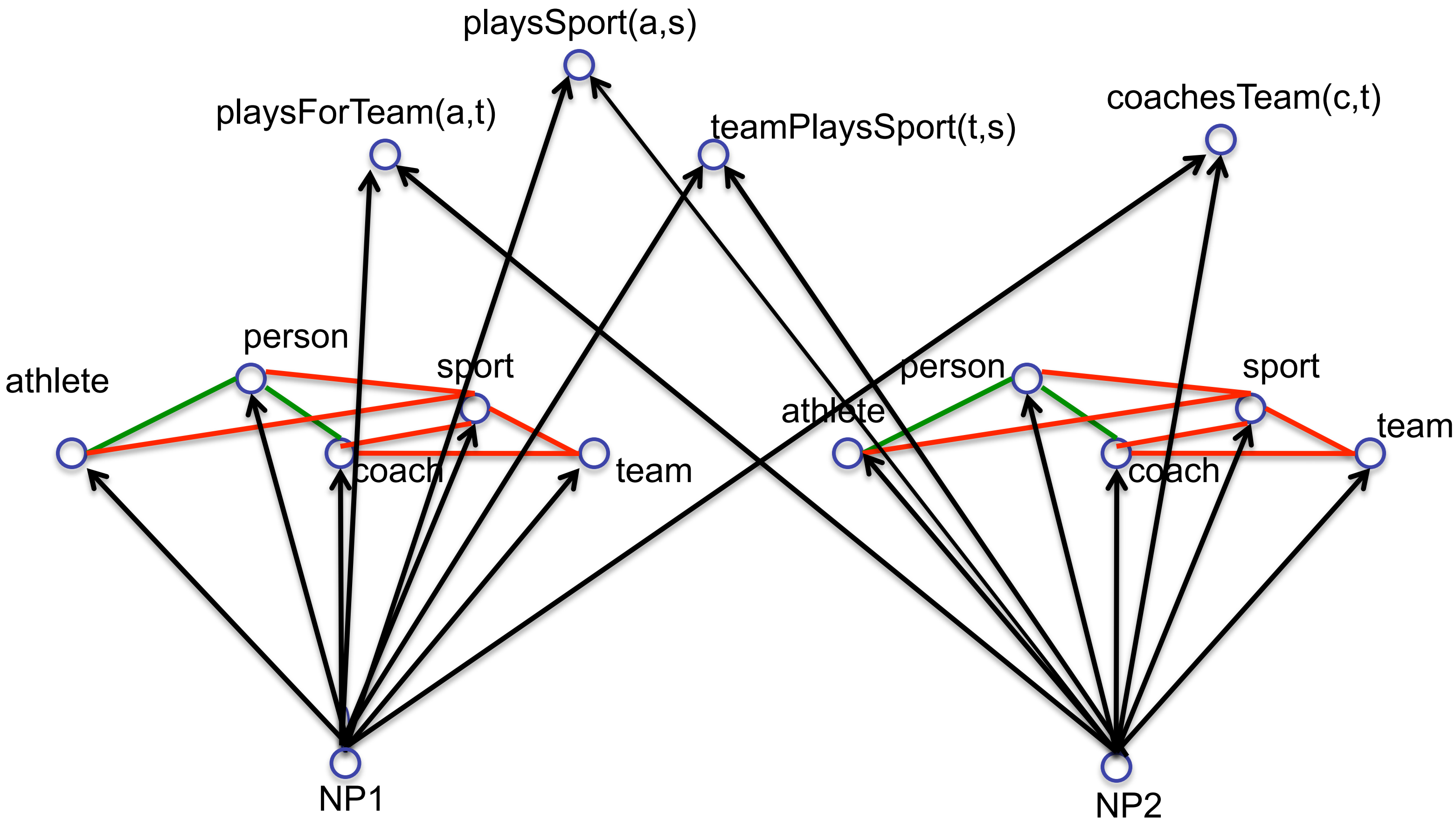
Multi-view, Multi-Task Coupling



NP: NP text context distribution | NP morphology | NP HTML contexts

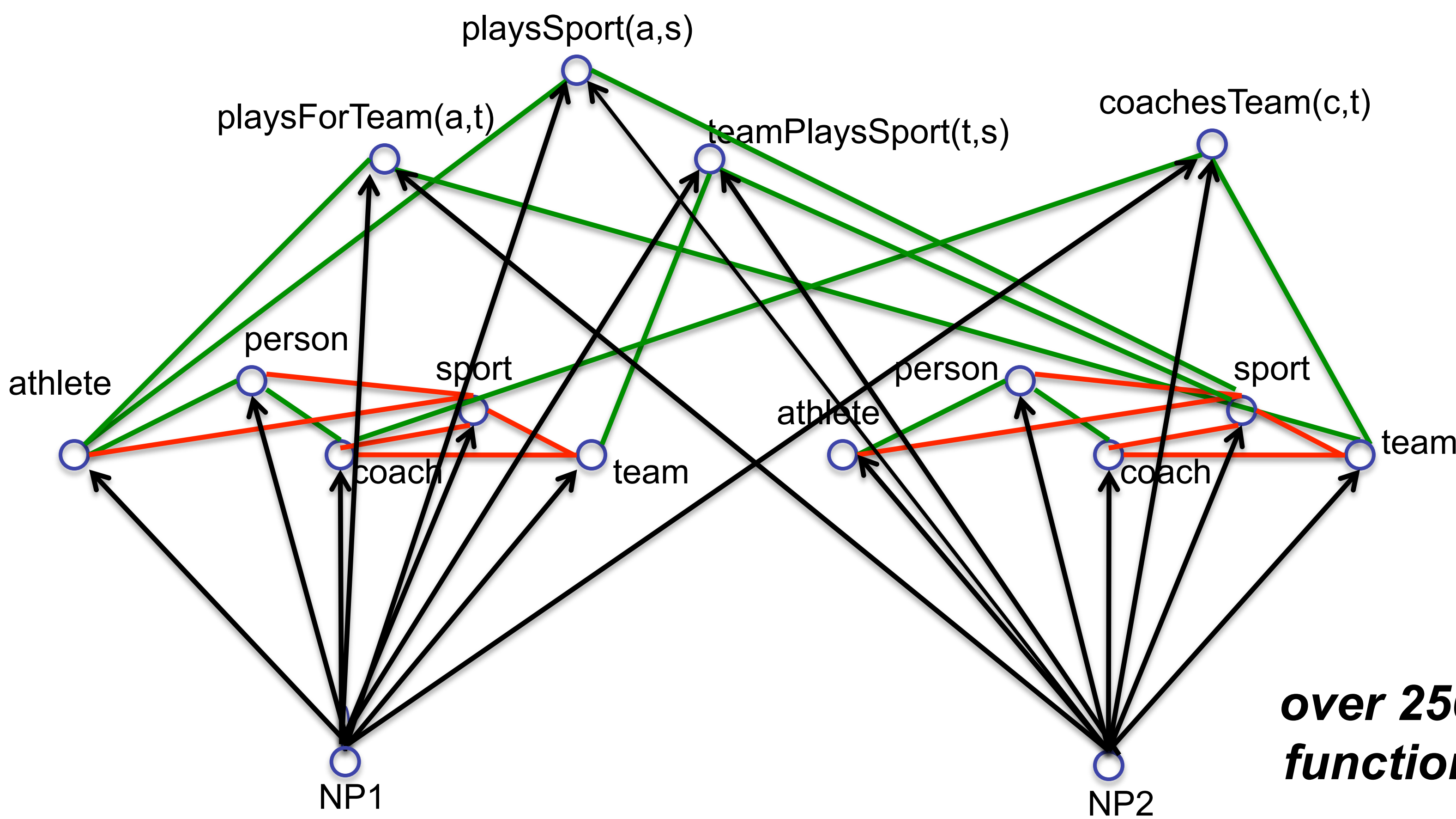
Learning Relations between NP's




playsSport(a,s)
playsForTeam(a,t)
teamPlaysSport(t,s)
coachesTeam(c,t)
person
athlete
sport
coach
team
NP1
person
sport
athlete
team
coach
NP2

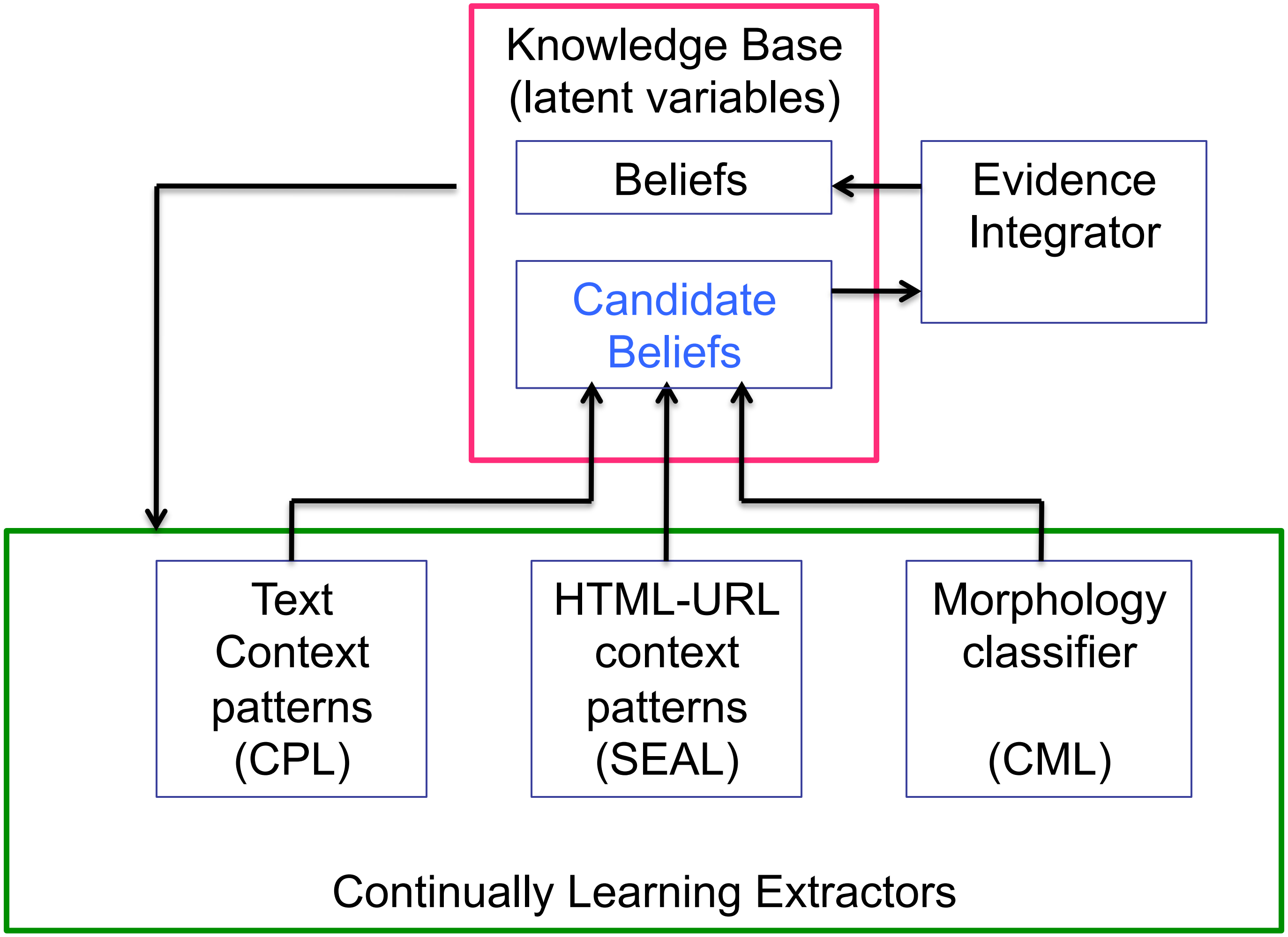
Type 3 Coupling: Argument Types

playsSport(NP1,NP2) → athlete(NP1), sport(NP2)



over 2500 coupled functions in NELL

Initial Core NELL Architecture



NELL: Learned reading strategies

Plays_Sport(arg1,arg2):

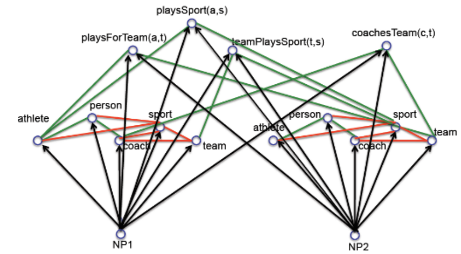
arg1_was_playing_arg2   arg2_megas
arg2_player_named_arg1   arg2_prod
arg1_is_the_tiger_woods_of_arg2   ar
arg2_greats_as_arg1   arg1_plays_ar
arg2_legends_arg1   arg1_announced
arg2_operations_chief_arg1   arg2_pla
arg2_and_golfing_personalities_incluc
arg2_greats_like_arg1   arg2_players_
arg2_great_arg1   arg2_champ_arg1
arg2_professionals_such_as_arg1   ar
arg2_icon_arg1   arg2_stars_like_arg1
arg1_retires_from_arg2   arg2_phenor
arg2_architects_robert_trent_jones_ar
arg2_pros_arg1   arg2_stars_venus_a
arg2_superstar_arg1   arg2_legend_a
arg2_players_is_arg1   arg2_pro_arg1
arg2_god_arg1   arg2_idol_arg1   arg1

Predicate	Feature	Weight
mountain	LAST=peak	1.791
mountain	LAST=mountain	1.093
mountain	FIRST=mountain	-0.875
musicArtist	LAST=band	1.853
musicArtist	POS=DT_NNS	1.412
musicArtist	POS=DT_JJ_NN	-0.807
newspaper	LAST=sun	1.330
newspaper	LAST=university	-0.318
newspaper	POS=NN_NNS	-0.798
university	LAST=college	2.076
university	PREFIX=uc	1.999
university	LAST=state	1.992
university	LAST=university	1.745
university	FIRST=college	-1.381
visualArtMovement	SUFFIX=ism	1.282

Predicate	Web URL	Extraction Template
academicField	http://scholendow.ais.msu.edu/student/ScholSearch.Asp	[X] -
athlete	http://www.quotes-search.com/d_occupation.aspx?o=+athlete	<a href='d_author.aspx?a=[X]'>-
bird	http://www.michaelforsberg.com/stock.html	<option>[X]</option>
bookAuthor	http://lifebehindthecurve.com/	[X] by [Y] –

If coupled learning is the key,
how can we get new coupling constraints?

Key Idea 2:



Discover New Coupling Constraints

- first order, probabilistic horn clause constraints:

0.93 athletePlaysSport(?x,?y) ←	athletePlaysForTeam(?x,?z)
	teamPlaysSport(?z,?y)

  - connects previously uncoupled relation predicates
  - infers new beliefs for KB

Example Learned Horn Clauses

0.95 athletePlaysSport(?x,basketball) ← athleteInLeague(?x,NBA)

0.93 athletePlaysSport(?x,?y) ← athletePlaysForTeam(?x,?z)
teamPlaysSport(?z,?y)

0.91 teamPlaysInLeague(?x,NHL) ← teamWonTrophy(?x,Stanley_Cup)

0.90 athleteInLeague(?x,?y) ←athletePlaysForTeam(?x,?z),
teamPlaysInLeague(?z,?y)

0.88 cityInState(?x,?y) ← cityCapitalOfState(?x,?y), cityInCountry(?y,USA)

0.62* newspaperInCity(?x,New_York) ← companyEconomicSector(?x,media)
generalizations(?x,blog)

Some rejected learned rules

teamPlaysInLeague{?x nba} ← teamPlaysSport{?x basketball}

0.94 [35 0 35] [positive negative unlabeled]

cityCapitalOfState{?x ?y} ← cityLocatedInState{?x ?y}, teamPlaysInLeague{?y nba}

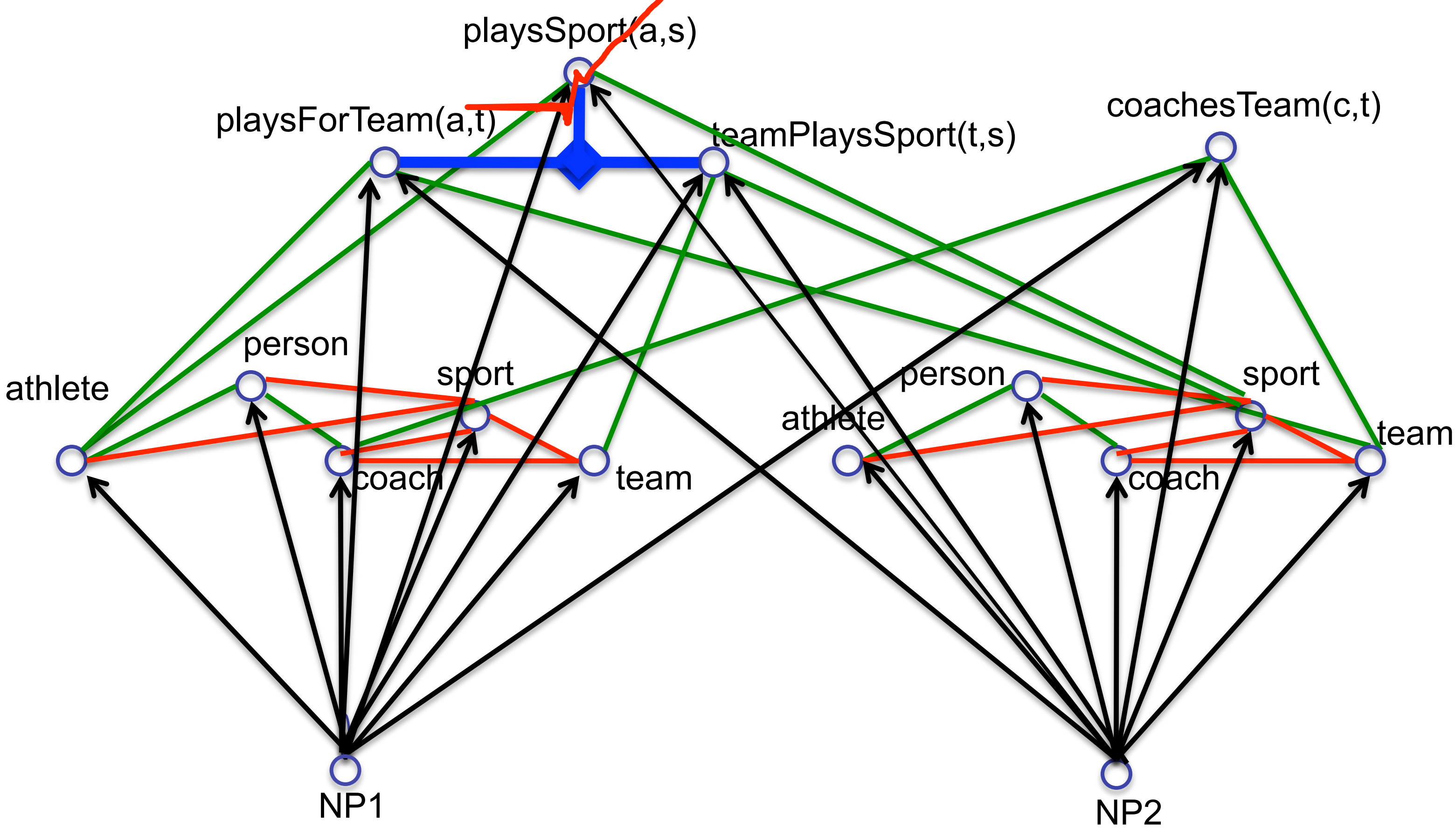
0.80 [16 2 23]

teamplayssport{?x, basketball} ← generalizations{?x, university}

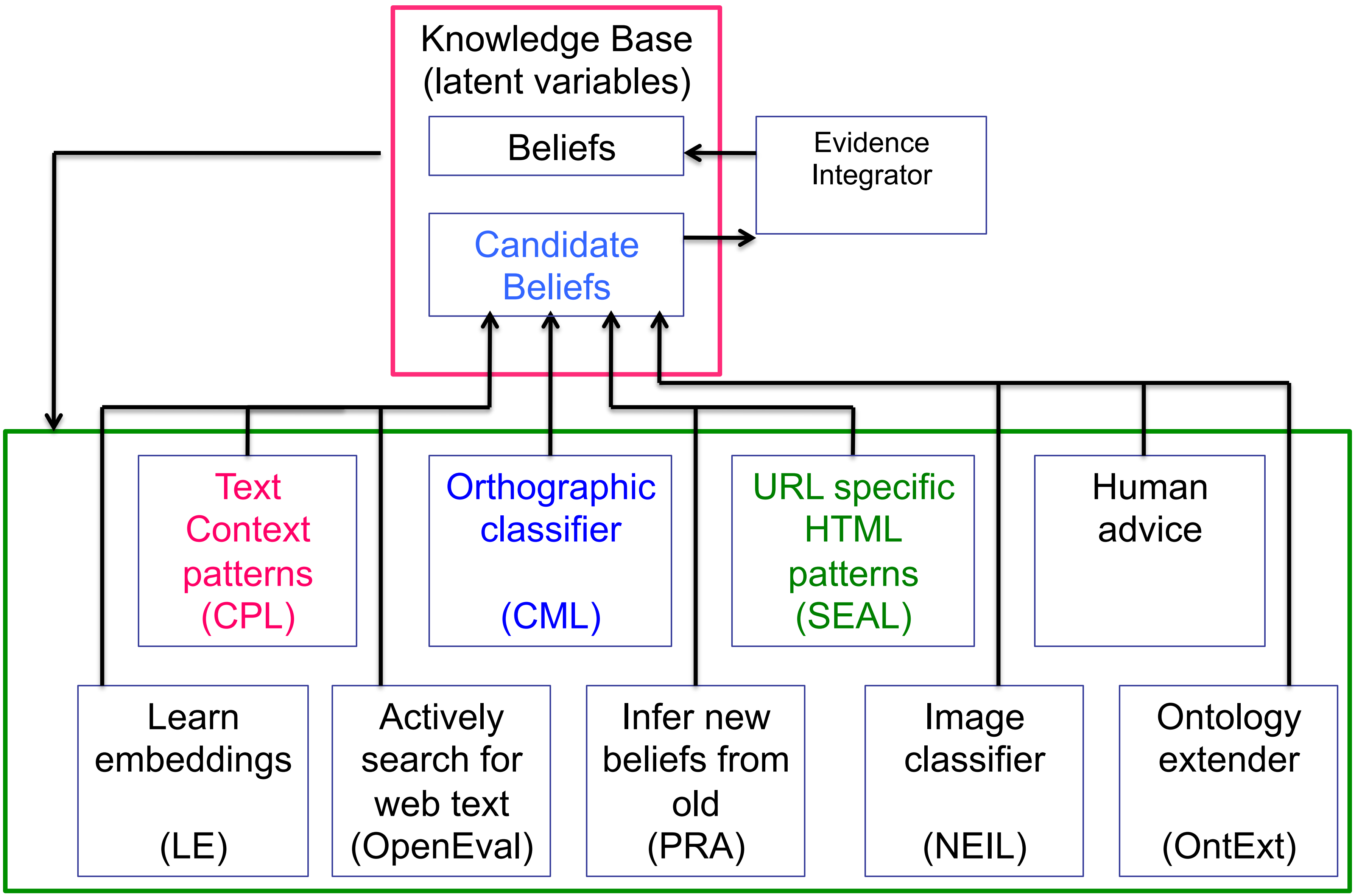
0.61 [246 124 3063]

Learned Probabilistic Horn Clause Rules

0.93 playsSport(?x,?y) ← playsForTeam(?x,?z), teamPlaysSport(?z,?y)



NELL Architecture



Research questions

How can we architect system so that acquiring one skill improves ability to learn others?

What parts of agent should be fixed, vs. plastic?

How to learn from mostly unsupervised training?

How to avoid “learning plateaus”?

What self-reflection and self-modification?

What theoretical guarantees?

Cumulative, Staged Learning in NELL

Learning X improves ability to learn Y

1. Classify noun phrases (NP's) by category
2. Classify NP pairs by relation
3. Discover rules to predict new relation instances
4. Learn which NP's (co)refer to which latent concepts
5. Discover new relations to extend ontology
6. Learn to infer relation instances via targeted random walks
7. Learn to microread single sentences, paragraphs
8. Vision: connect NELL and NEIL
9. Learn in multiple languages
10. Goal-driven reading: predict, then read to corroborate/correct
11. Make NELL a conversational agent on Twitter
12. Add a robot body to NELL

NELL is here

Further Reading

- Semi-Supervised Learning, O. Chapelle, B. Sholkopf, and A. Zien (eds.), MIT Press, 2006. (book)
- Semi-Supervised Learning. Encyclopedia of Machine Learning. Jerry Zhu, 2010
- EM for Naïve Bayes classifiers: K.Nigam, et al., 2000. "Text Classification from Labeled and Unlabeled Documents using EM", *Machine Learning,* 39, pp.103—134.
- CoTraining: A. Blum and T. Mitchell, 1998. "Combining Labeled and Unlabeled Data with Co-Training," *Proceedings of the 11th Annual Conference on Computational Learning Theory (COLT-98).*
- Never Ending Learning, T. Mitchell et al., CACM, Dec. 2017.
- Model selection: D. Schuurmans and F. Southey, 2002. "Metric-Based methods for Adaptive Model Selection and Regularization," Machine Learning, 48, 51—84.