Never Ending Language Learning

Tom M. Mitchell
Machine Learning Department
Carnegie Mellon University

Thesis:

We will never really understand learning until we build machines that

- learn many different things,
- from years of diverse experience,
- in a staged, curricular fashion,
- and become better <u>learners</u> over time.

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 today

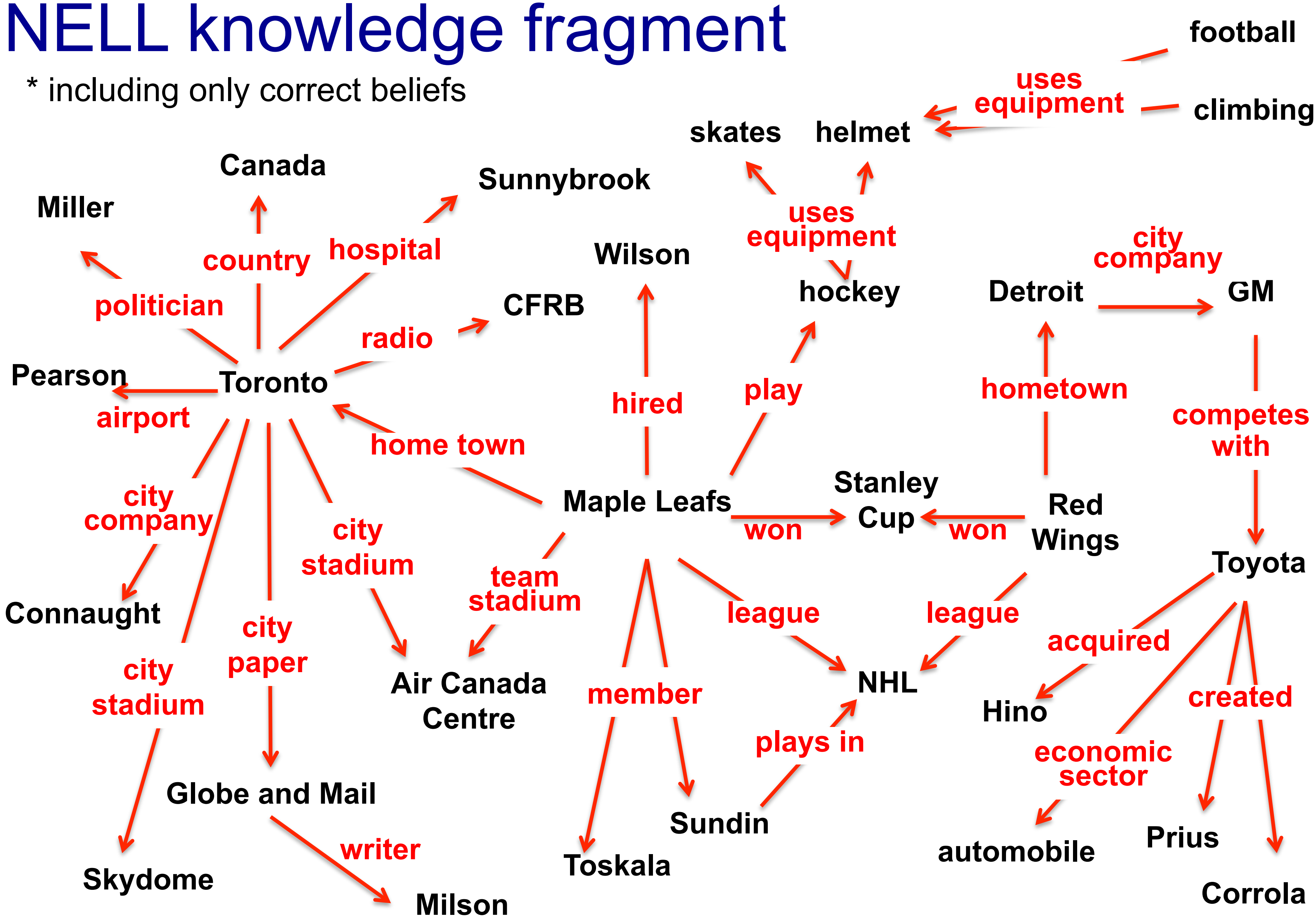
Running 24x7, since January, 12, 2010

Result:

- KB with ~120 million confidence-weighted beliefs
- learning to read
- learning to reason
- extending ontology

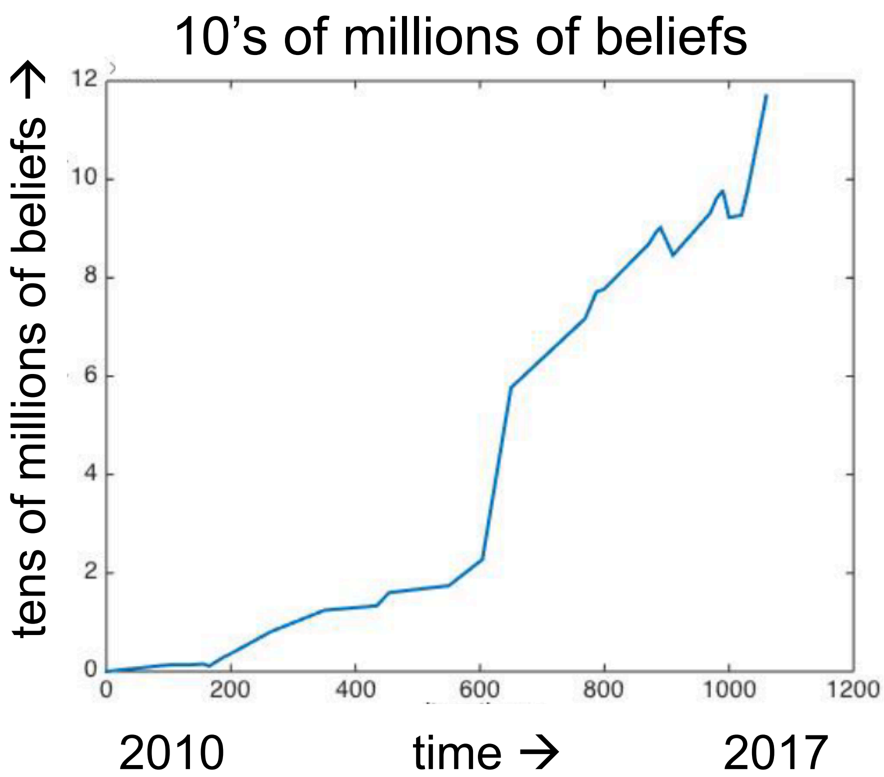
NELL knowledge fragment

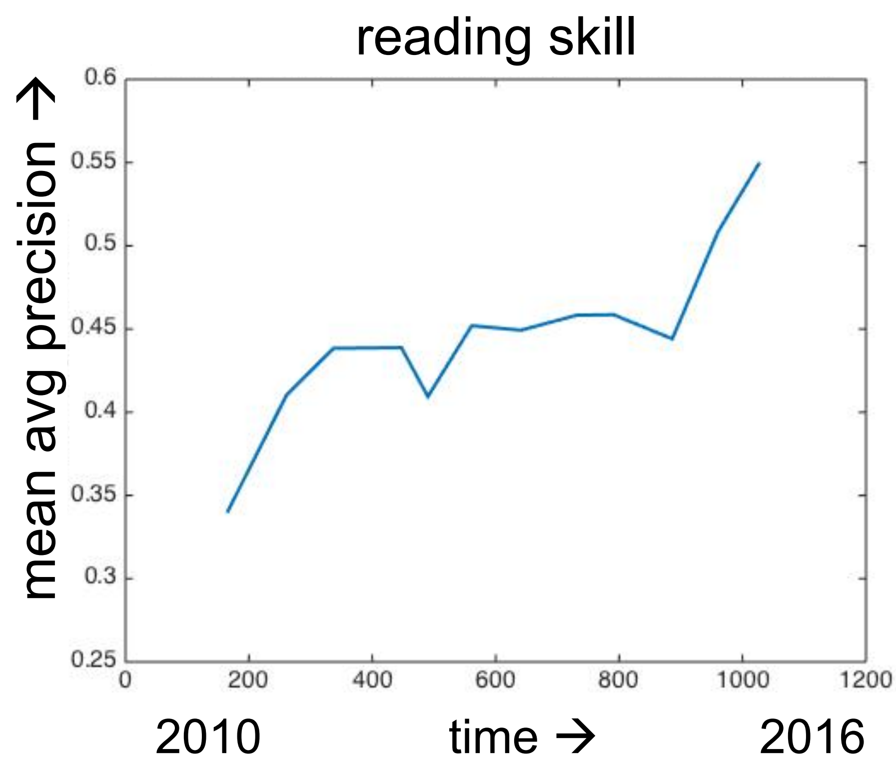
* including only correct beliefs



Improving Over Time
Never Ending Language Learner

[Mitchell et al., CACM 2017]





Semi-Supervised Bootstrap Learning

it's underconstrained!!

Learn which noun phrases are cities:

Paris
Pittsburgh
Seattle
Montpelier

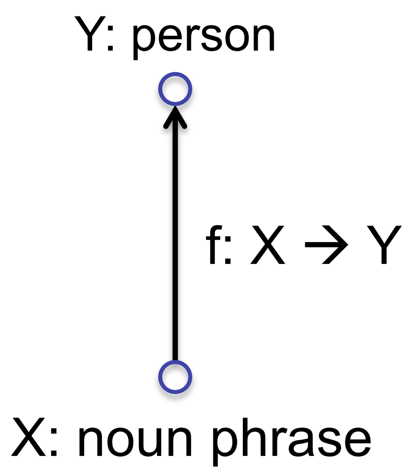
mayor of arg1
live in arg1

San Francisco
Berlin
denial

arg1 is home of
traits such as arg1

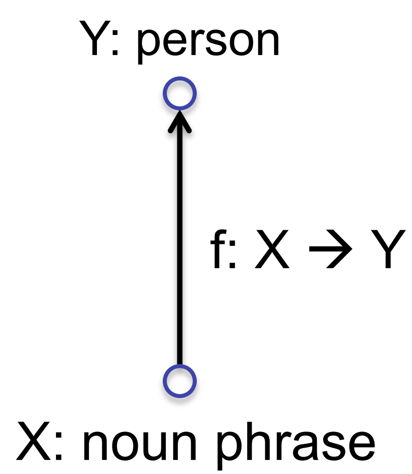
anxiety
selfishness
London

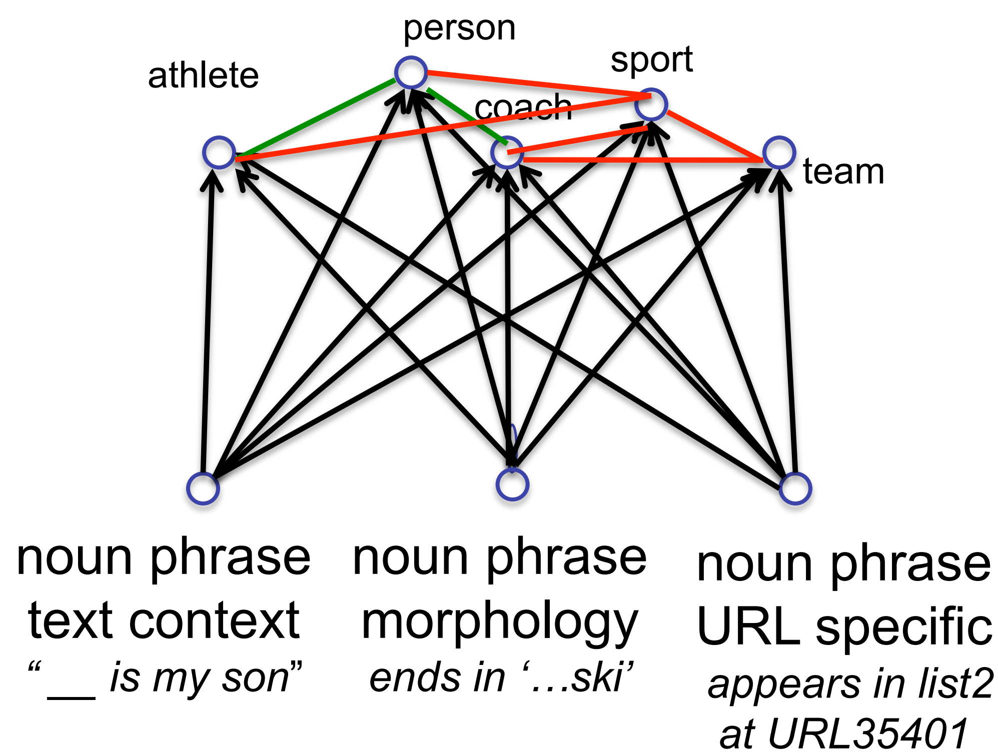
Key Idea 1: Coupled semi-supervised training: multi-view and multi-task



hard
(underconstrained)
semi-supervised
learning

Key Idea 1: Coupled semi-supervised training: multi-view and multi-task



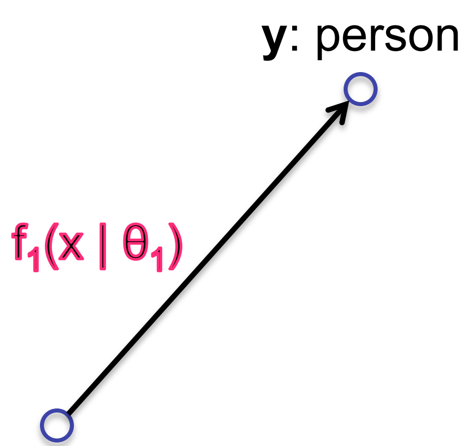


hard
(underconstrained)
semi-supervised
learning

much easier
(more constrained)
semi-supervised
learning

Supervised training of 1 function:

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

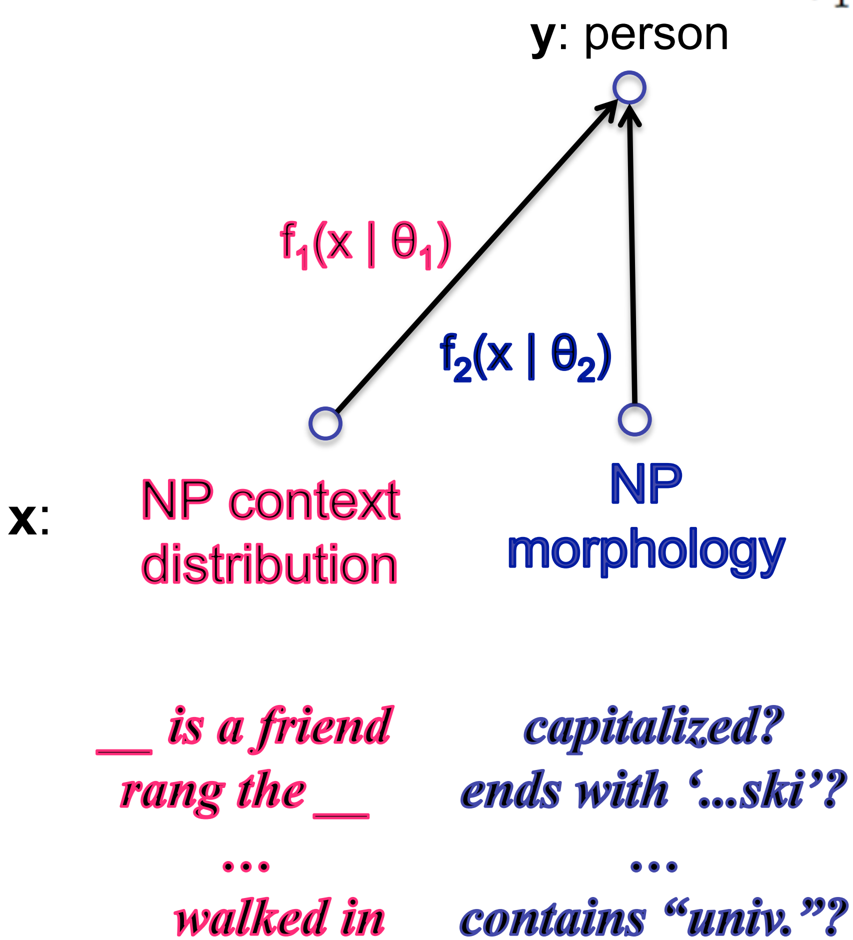


x: NP context distribution

__ is a friend
rang the __
…
__ walked in

Coupled training of 2 functions:

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



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

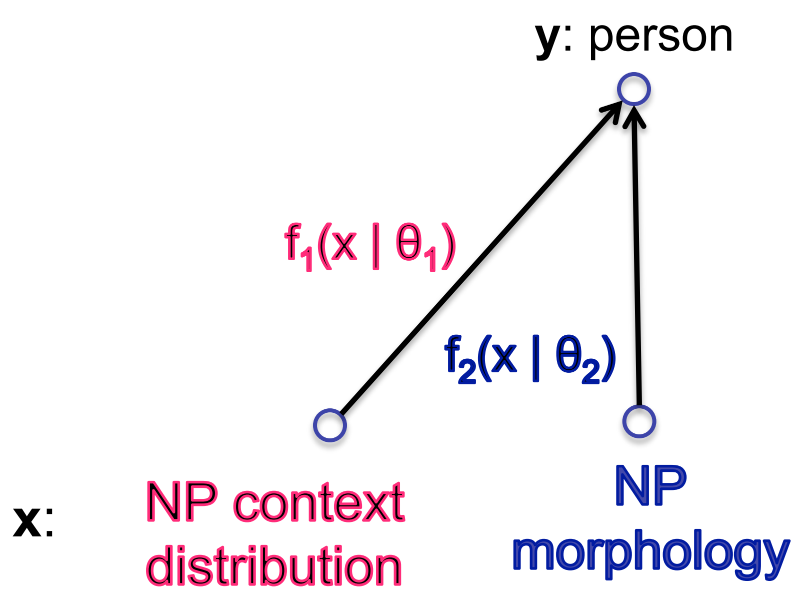
"_ is the only five-star hotel" "_ is the only hotel" "_ is the perfect accommodation" "_ is the perfect address" "_ is the perfect lodging" "_ is the sister hotel" "_ is the ultimate hotel" "_ is the value choice" "_ is uniquely situated in" "_ is Walking Distance" "_ is wonderfully situated in" "_ las vegas hotel" "_ los angeles hotels" "_ 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" "_ 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" "_ 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”

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

* logistic regression

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



__ is a friend
rang the __
…
__ walked in

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

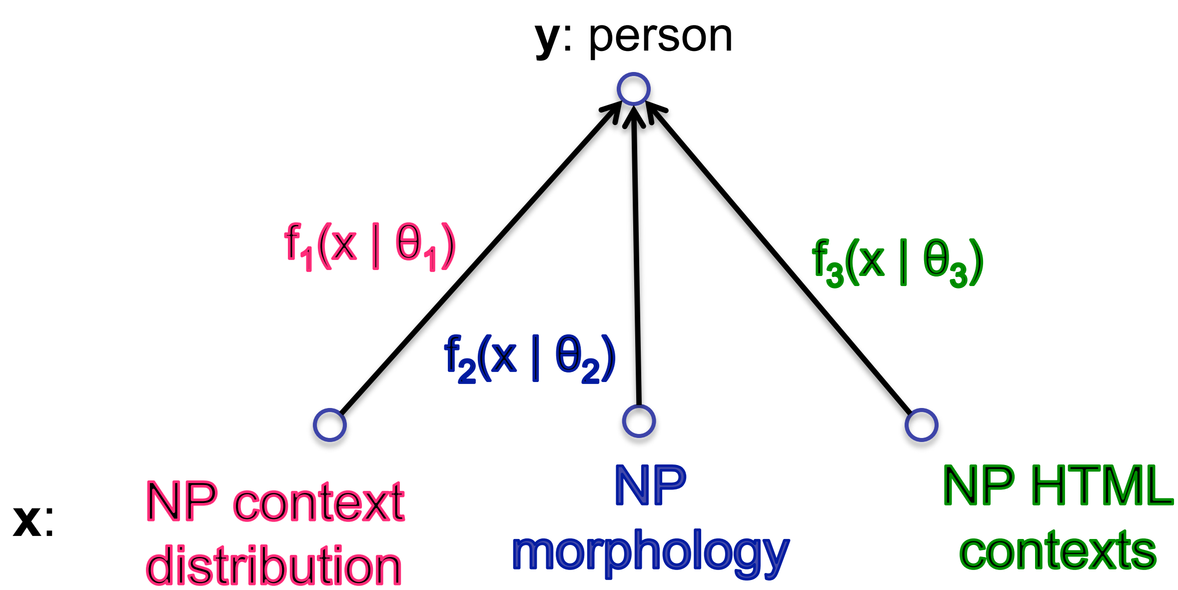
Theorem (Blum & Mitchell, 1998):

If f_1, and f_2 are PAC learnable from noisy *labeled* data, and X_1, X_2 are conditionally independent given Y,

Then f_1, f_2 are PAC learnable from polynomial *unlabeled* data plus a weak initial predictor

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

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

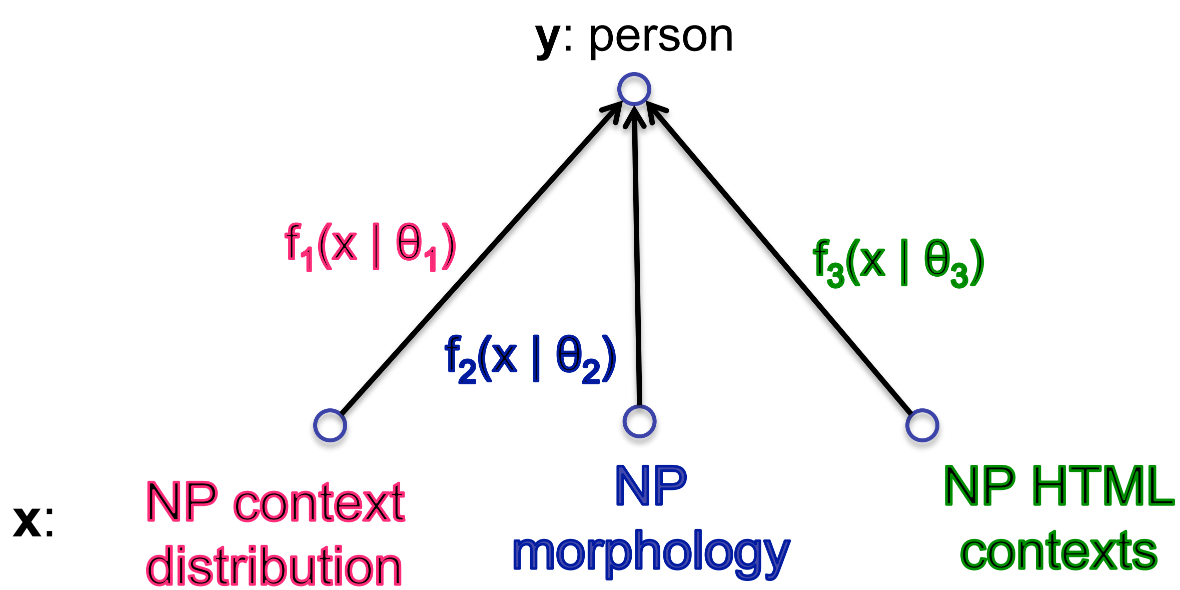


__ is a friend	*capitalized?*	*www.celebrities.com:*
rang the __	*ends with '...ski'?*	* __ *
...	...	
__ walked in	*contains "univ."?*	...

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

sample complexity drops exponentially in the number of views of X

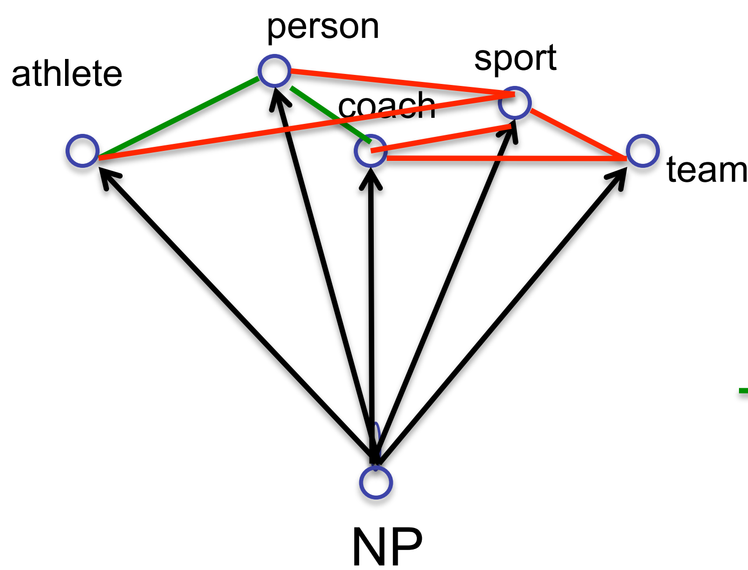
[Blum & Mitchell; 98]
[Dasgupta et al; 01]
[Balcan & Blum; 08]
[Ganchev et al., 08]
[Sridharan & Kakade, 08]
[Wang & Zhou, ICML10]



__ is a friend	*capitalized?*	*www.celebrities.com:*
rang the __	*ends with '...ski'?*	* __ *
...	...	
__ walked in	*contains "univ."?*	...

Type 2 Coupling: Multi-task, Structured Outputs

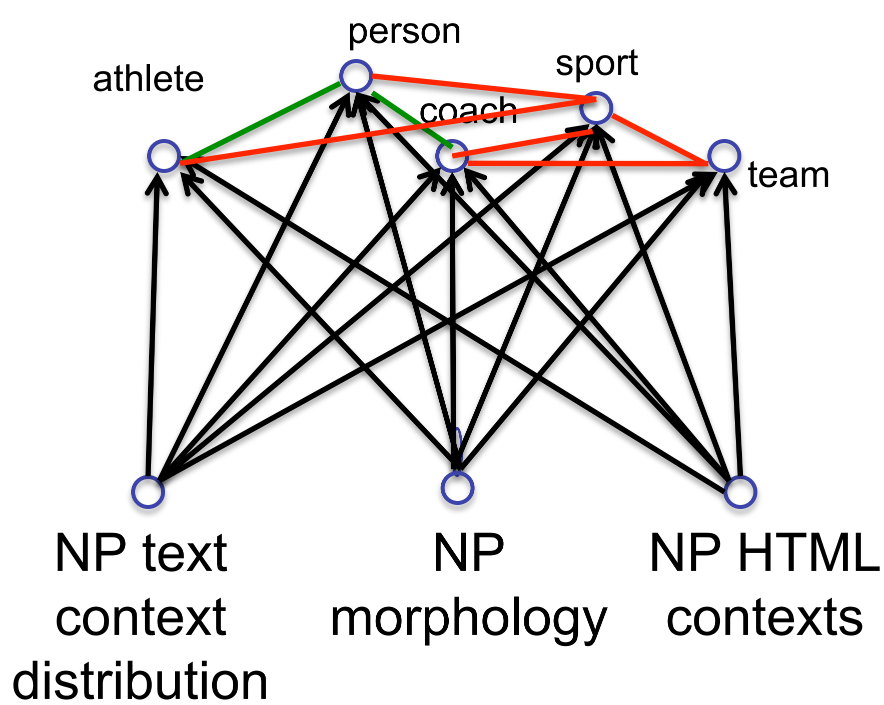
[Daume, 2008]
[Bakhir et al., eds. 2007]
[Roth et al., 2008]
[Taskar et al., 2009]
[Carlson et al., 2009]



subset/superset
athlete(NP) → person(NP)

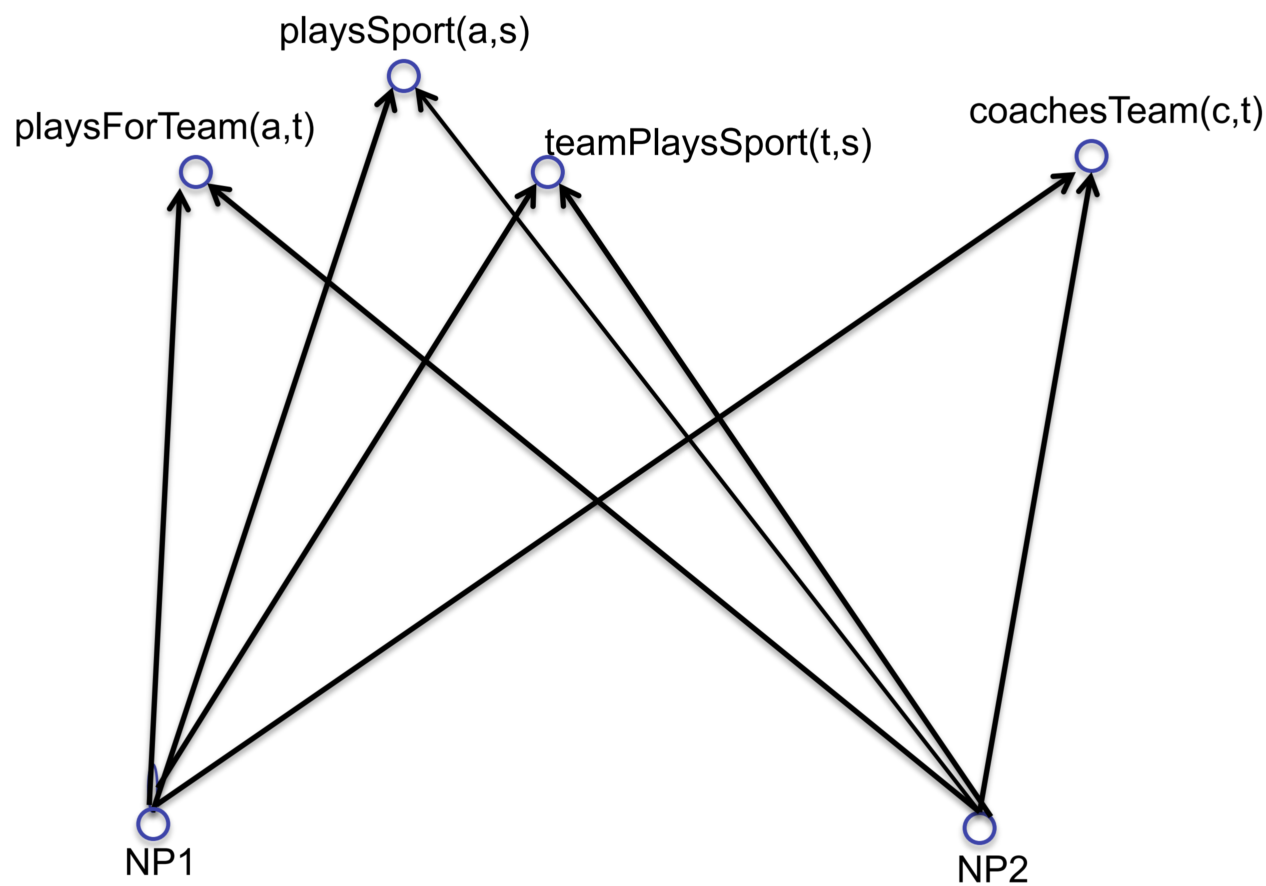
mutual exclusion
athlete(NP) → NOT sport(NP)
sport(NP) → NOT athlete(NP)

Multi-view, Multi-Task Coupling

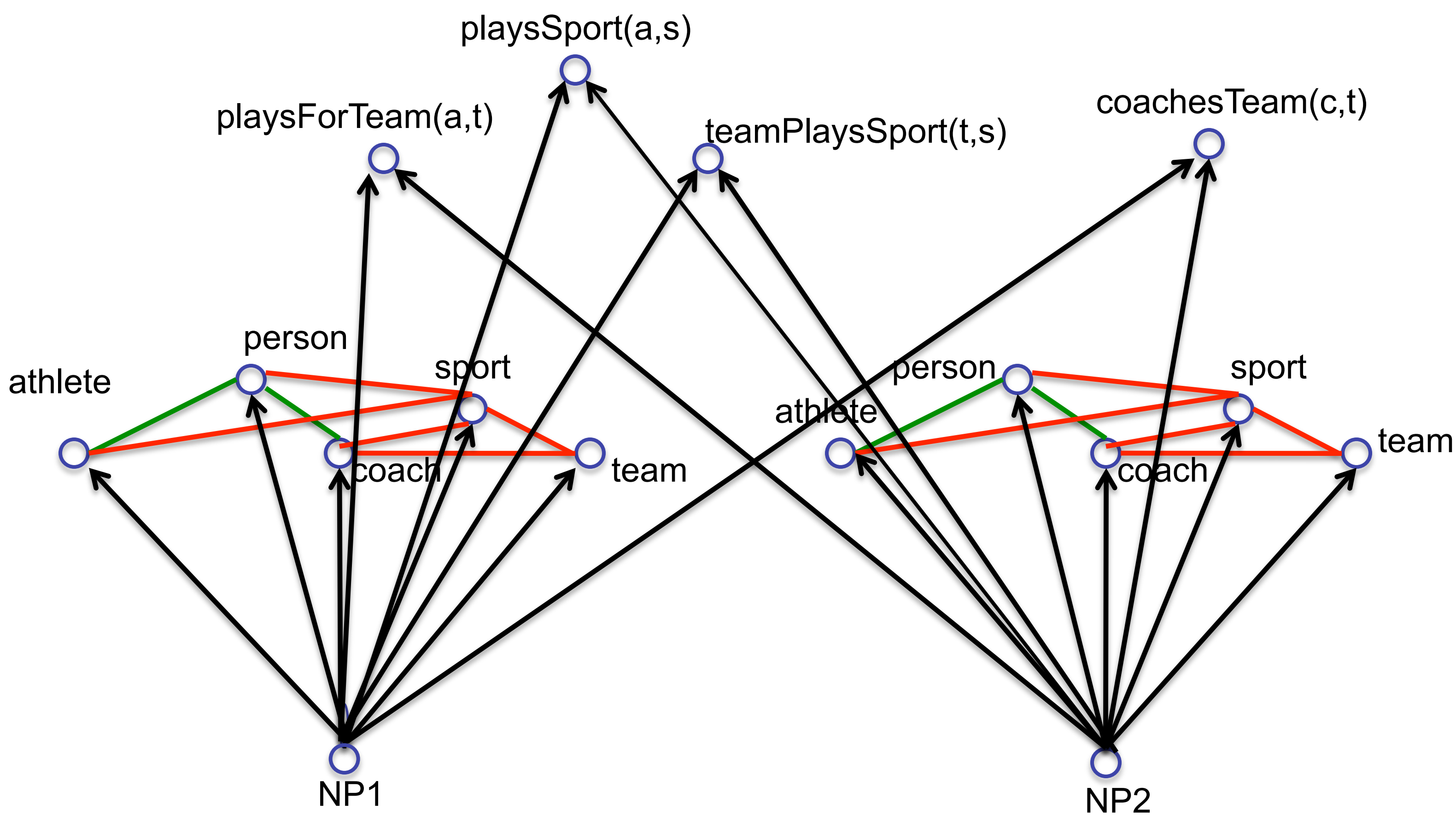


NP:

Type 3 Coupling: Relations and Argument Types

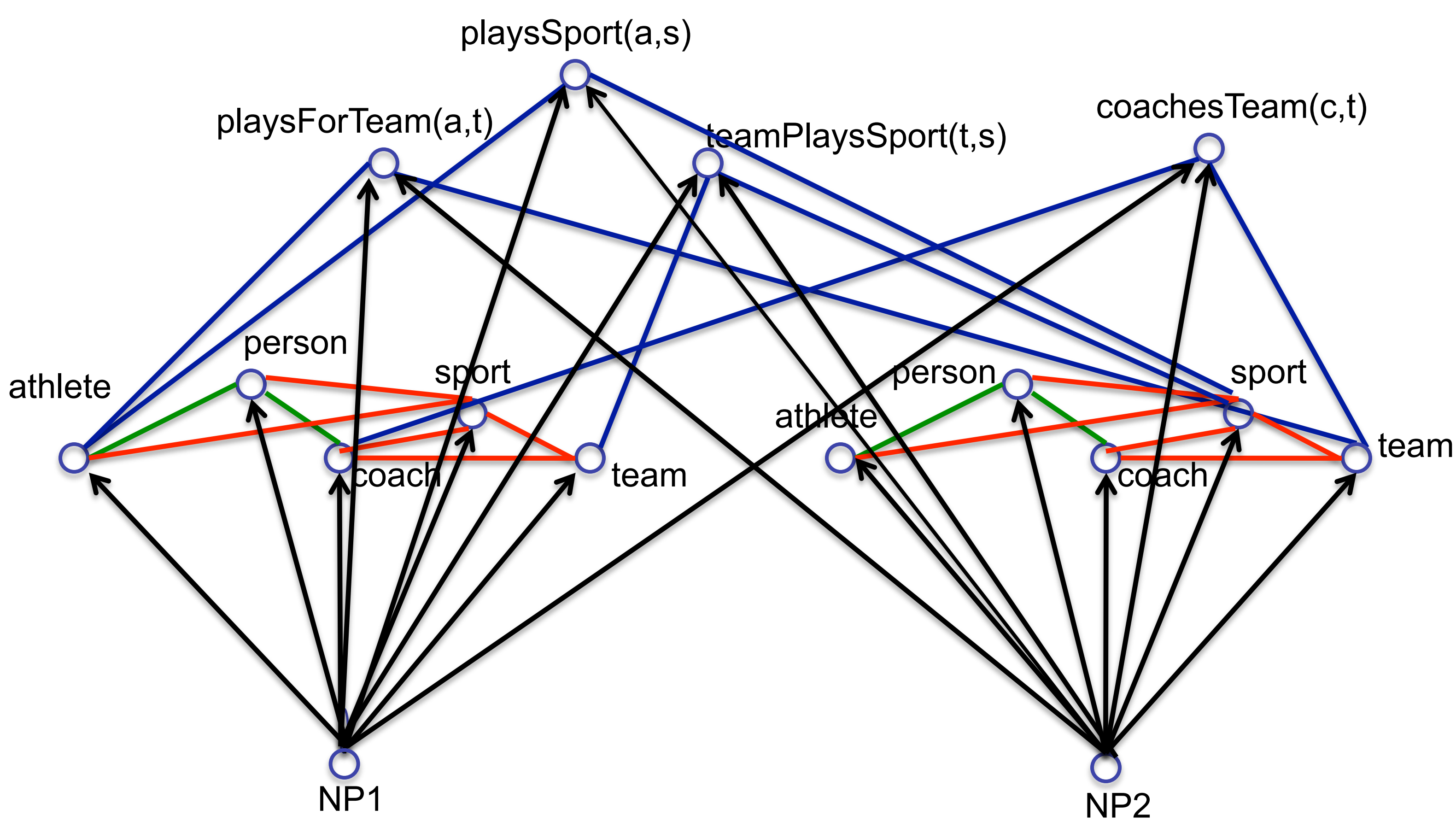


Type 3 Coupling: Relations and Argument Types



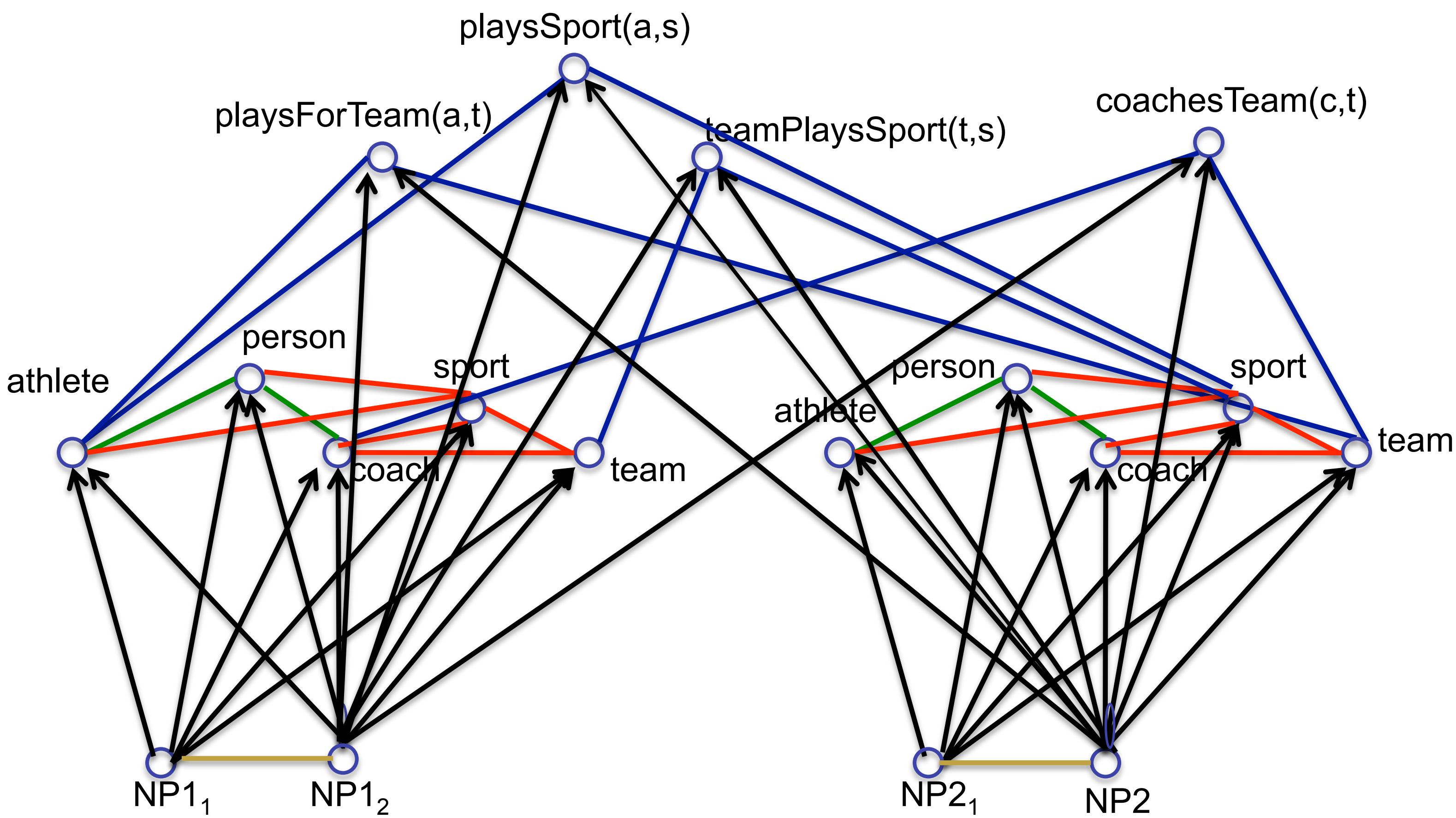
Type 3 Coupling: Relations and Argument Types

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



Type 3 Coupling: Relations and Argument Types

over 4000 coupled functions in NELL



multi-view consistency **subset/superset**

argument type consistency **mutual exclusion**

How to train

approximation to EM:

- E step: predict beliefs from unlabeled data (ie., the KB)
- M step: retrain

NELL approximation:

- bound number of new beliefs per iteration, per predicate
- rely on multiple iterations for information to propagate, partly through joint assignment, partly through training examples

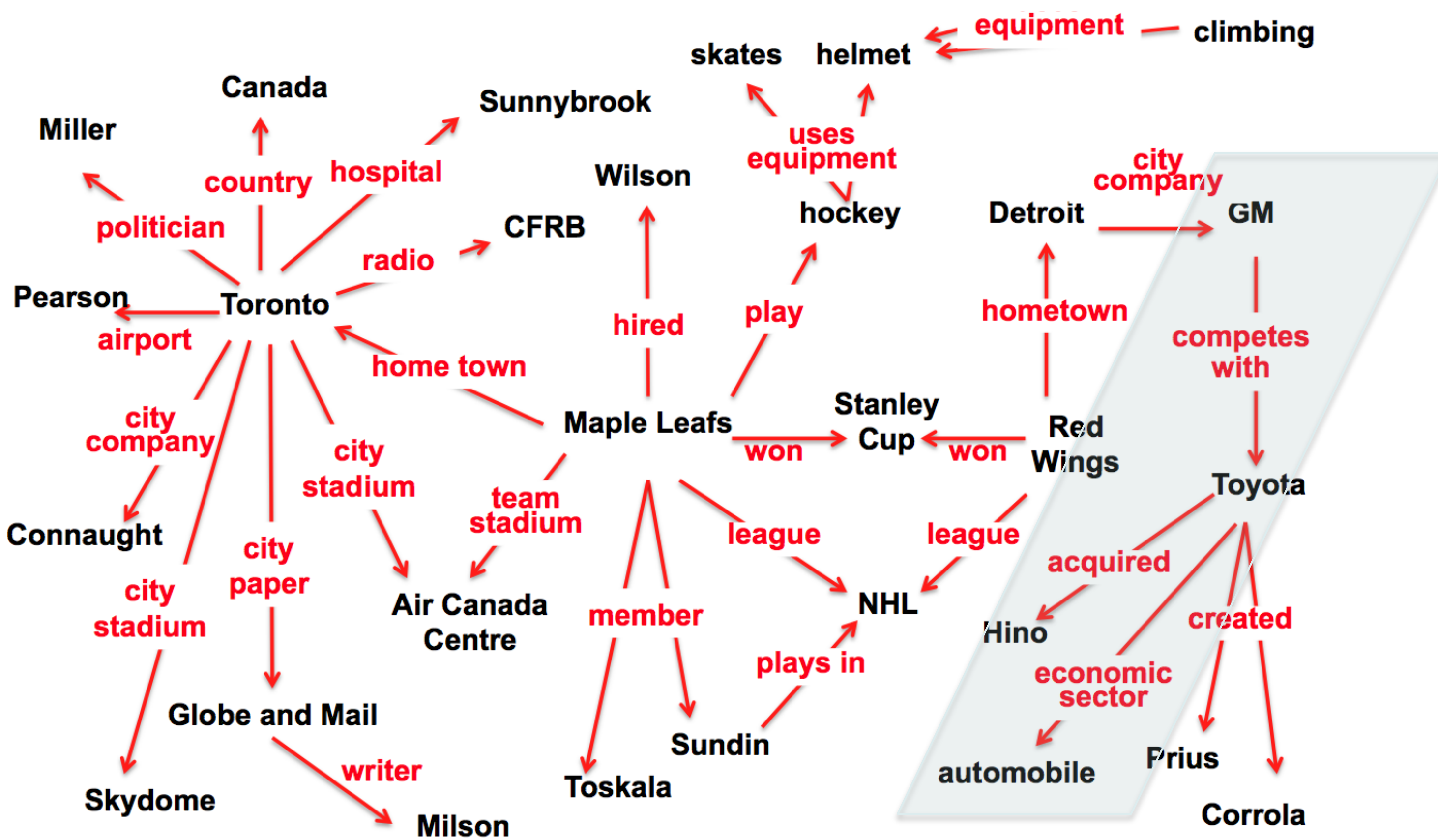
Better approximation:

- Joint assignments based on probabilistic soft logic [Pujara, et al., 2013] [Platanios et al., 2017]

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

Key Idea 2: Learn inference rules

PRA: [Lao, Mitchell, Cohen, *EMNLP* 2011]

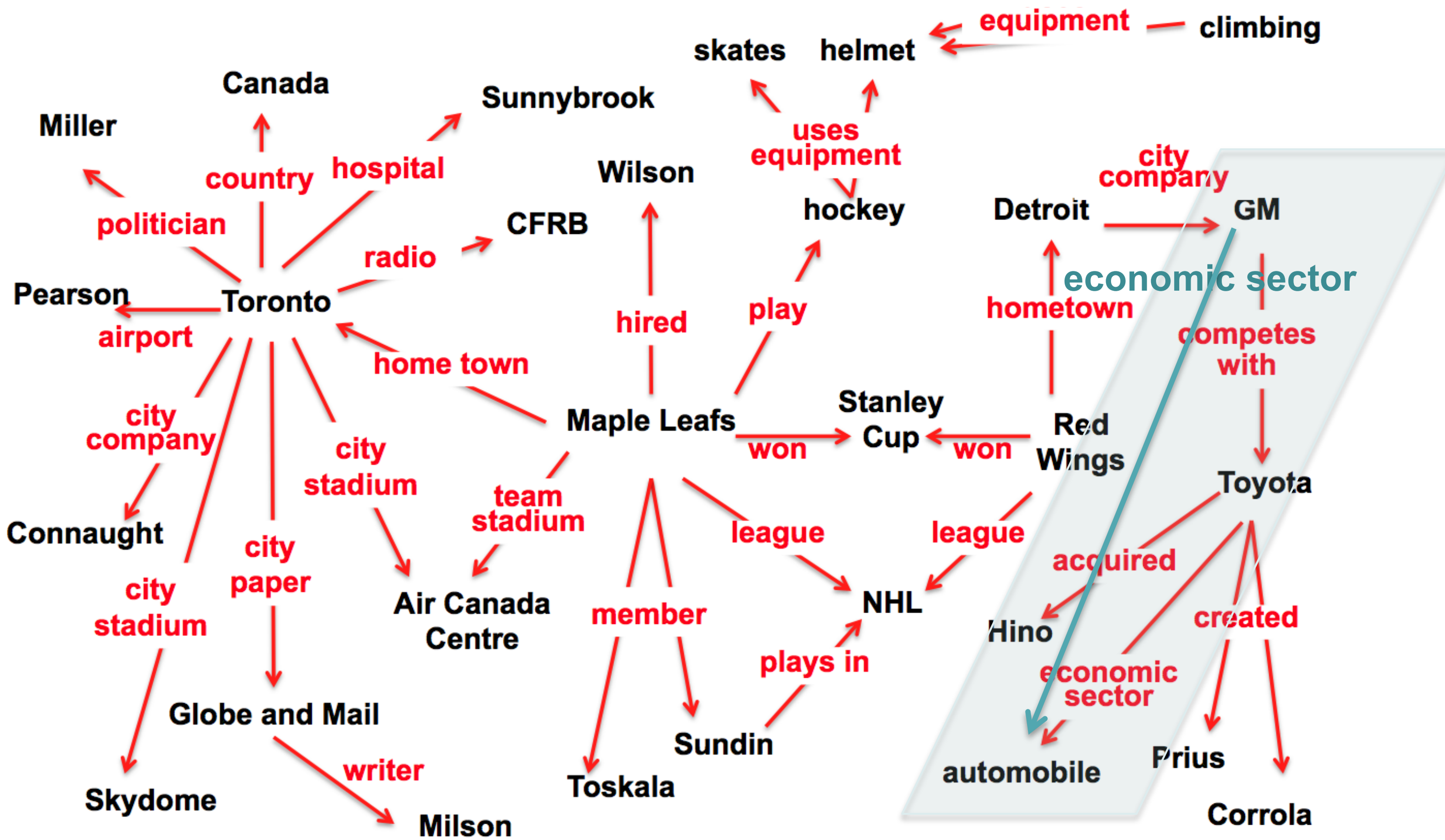


If: x1 — **competes with (x1,x2)** → x2 — **economic sector (x2, x3)** → x3

Then: **economic sector (x1, x3)** **with probability 0.9**

Key Idea 2: Learn inference rules

PRA: [Lao, Mitchell, Cohen, *EMNLP* 2011]

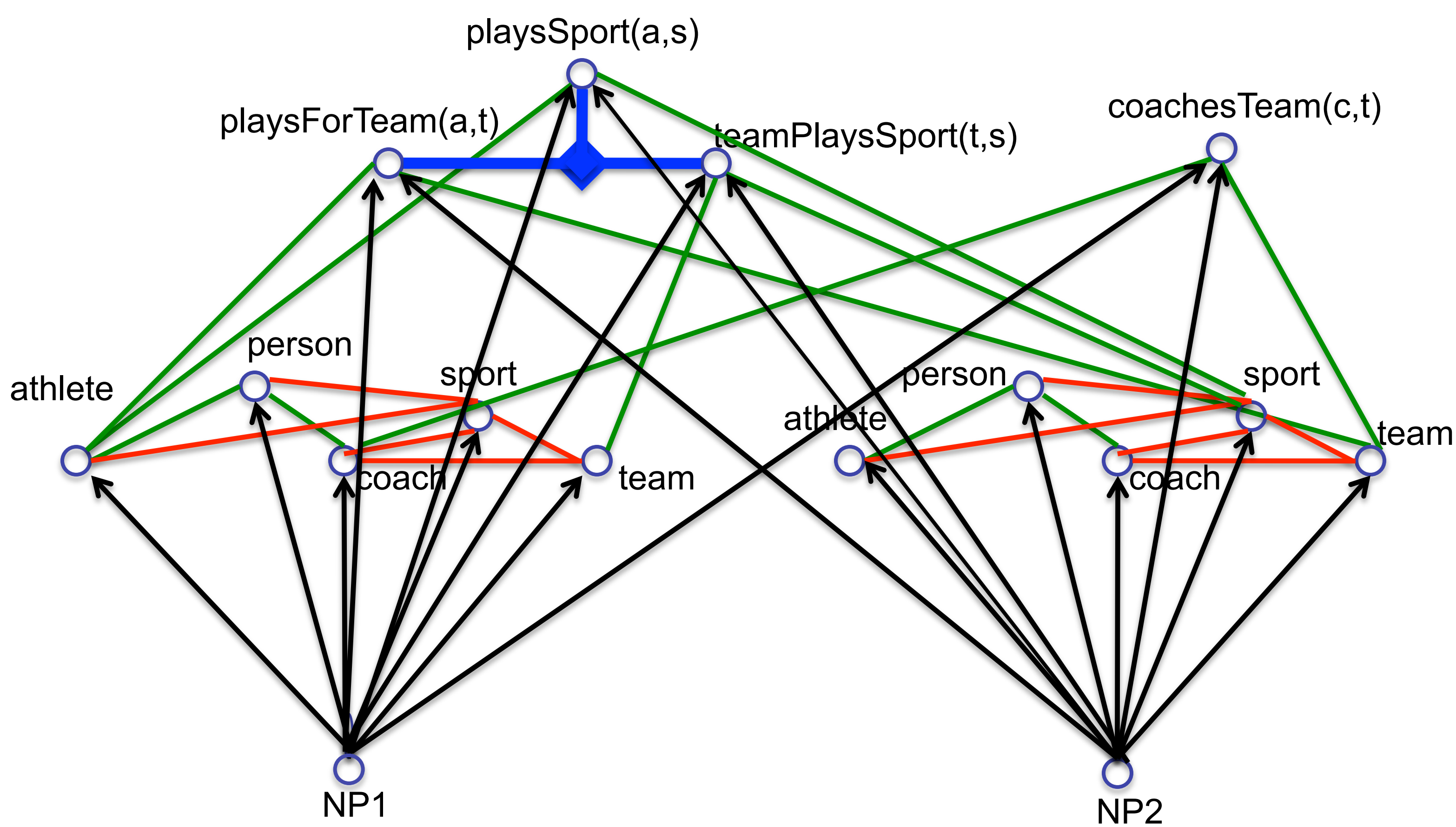


If: x1 — competes with (x1,x2) → x2 — economic sector (x2, x3) → x3

Then: economic sector (x1, x3) with probability 0.9

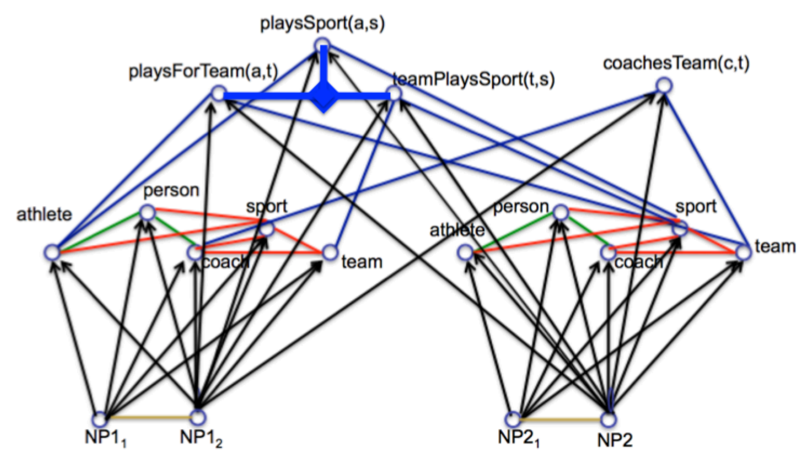
Learned Rules are New Coupling Constraints!

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



Learned Rules are New Coupling Constraints!

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



- Learning X makes one a better learner of Y
- Learning Y makes one a better learner of X

X = reading functions: `text → beliefs`

Y = Horn clause rules: `beliefs → beliefs`

Consistency and Correctness

what is the relationship?

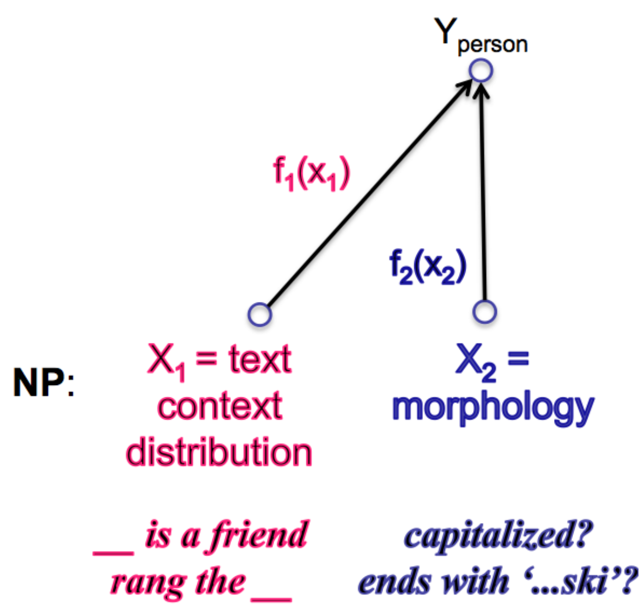
under what conditions?

The core problem:

- Unsupervised agents can measure their internal *consistency*, but not their *correctness*

Challenge:

- Under what conditions does *consistency* → *correctness*?

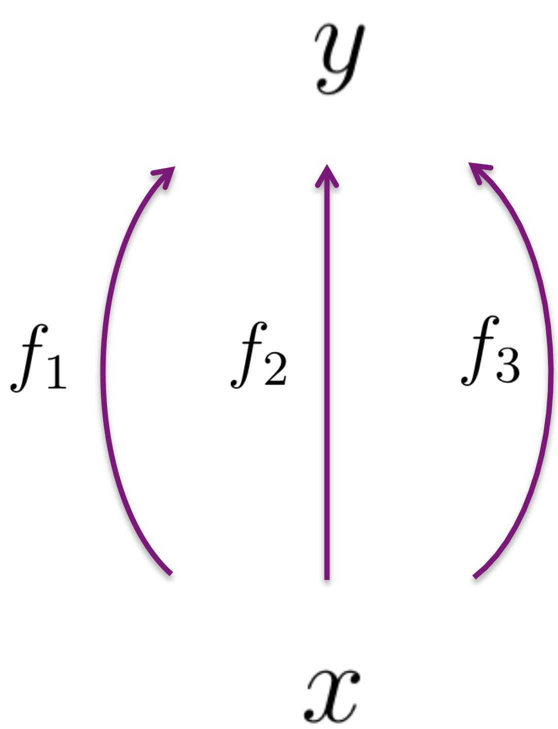


[Platanios, Blum, Mitchell]

Problem setting:

- have N different estimates $f_1, \ldots f_N$ of target function f^*

$$y = f^*(x); \quad y \in \{0, 1\}$$



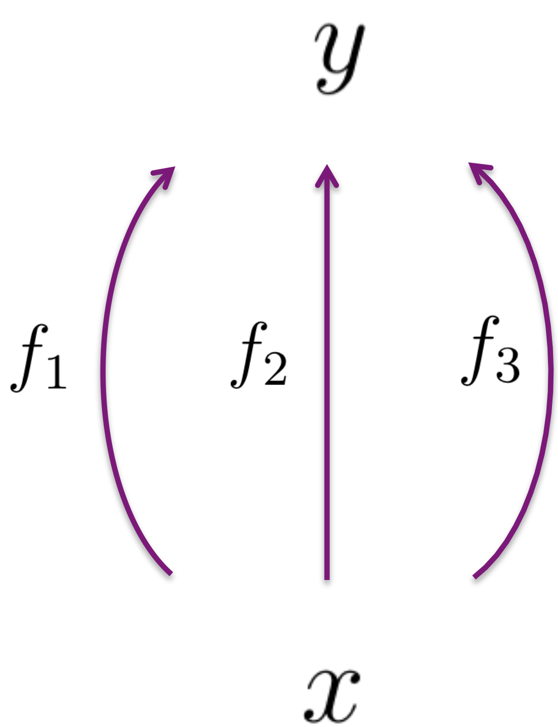
y = NELL category "city"

f_i = classifier based on ith view of x

x = noun phrase

Problem setting:

- have N different estimates $f_1, \dots f_N$ of target function f^*



y = disease

f_i = i[th] diagnostic test

x = medical patient

[Hui & Walter, 1980; Collins & Huynh, 2014]

[Platanios, Blum, Mitchell]

Problem setting:

- have N different estimates $f_1, \ldots f_N$ of target function f^*

$$f^* : X \to Y; \quad Y \in \{0, 1\}$$

Goal:

- estimate accuracy of each of $f_1, \ldots f_N$ from **unlabeled** data

[Platanios, Blum, Mitchell]

Problem setting:

- have N different estimates $f_1, \dots f_N$ of target function f^*

  $$f^* : X \rightarrow Y; \quad Y \in \{0, 1\}$$

- *agreement* between f_i, f_j : $a_{ij} \equiv P_x(f_i(x) = f_j(x))$

Problem setting:

- have N different estimates $f_1, \dots f_N$ of target function f^*

  $f^* : X \rightarrow Y; \quad Y \in \{0,1\}$

- *agreement* between f_i, f_j : $a_{ij} \equiv P_x(f_i(x) = f_j(x))$

Key insight: errors and agreement rates are related

agreement can be estimated from unlabeled data

$$a_{ij} = \text{Pr[neither makes error] + Pr[both make error]}$$

$$a_{ij} = 1 - e_i - e_j + 2e_{ij}$$

prob. f_i and f_i agree

prob. f_i error

prob. f_j error

prob. f_i and f_j simultaneous error

Estimating Error from Unlabeled Data

1. IF f_1, f_2, f_3 make <u>independent</u> errors, and accuracies > 0.5

   then $$a_{ij} = 1 - e_i - e_j + 2e_{ij}$$

   becomes $$a_{ij} = 1 - e_i - e_j + 2e_i e_j$$

Determine errors from unlabeled data!

- use unlabeled data to estimate a_{12}, a_{13}, a_{23}
- solve three equations for three unknowns e_1, e_2, e_3

Estimating Error from Unlabeled Data

1. IF f_1, f_2, f_3 make indep. errors, accuracies > 0.5

   then $a_{ij} = 1 - e_i - e_j + 2e_{ij}$

   becomes $a_{ij} = 1 - e_i - e_j + 2e_i e_j$

2. but if errors **not** independent

Estimating Error from Unlabeled Data

1. IF f_1, f_2, f_3 make indep. errors, accuracies > 0.5

   then $$a_{ij} = 1 - e_i - e_j + 2e_{ij}$$

   becomes $$a_{ij} = 1 - e_i - e_j + 2e_i e_j$$

2. but if errors **not** independent, add prior:

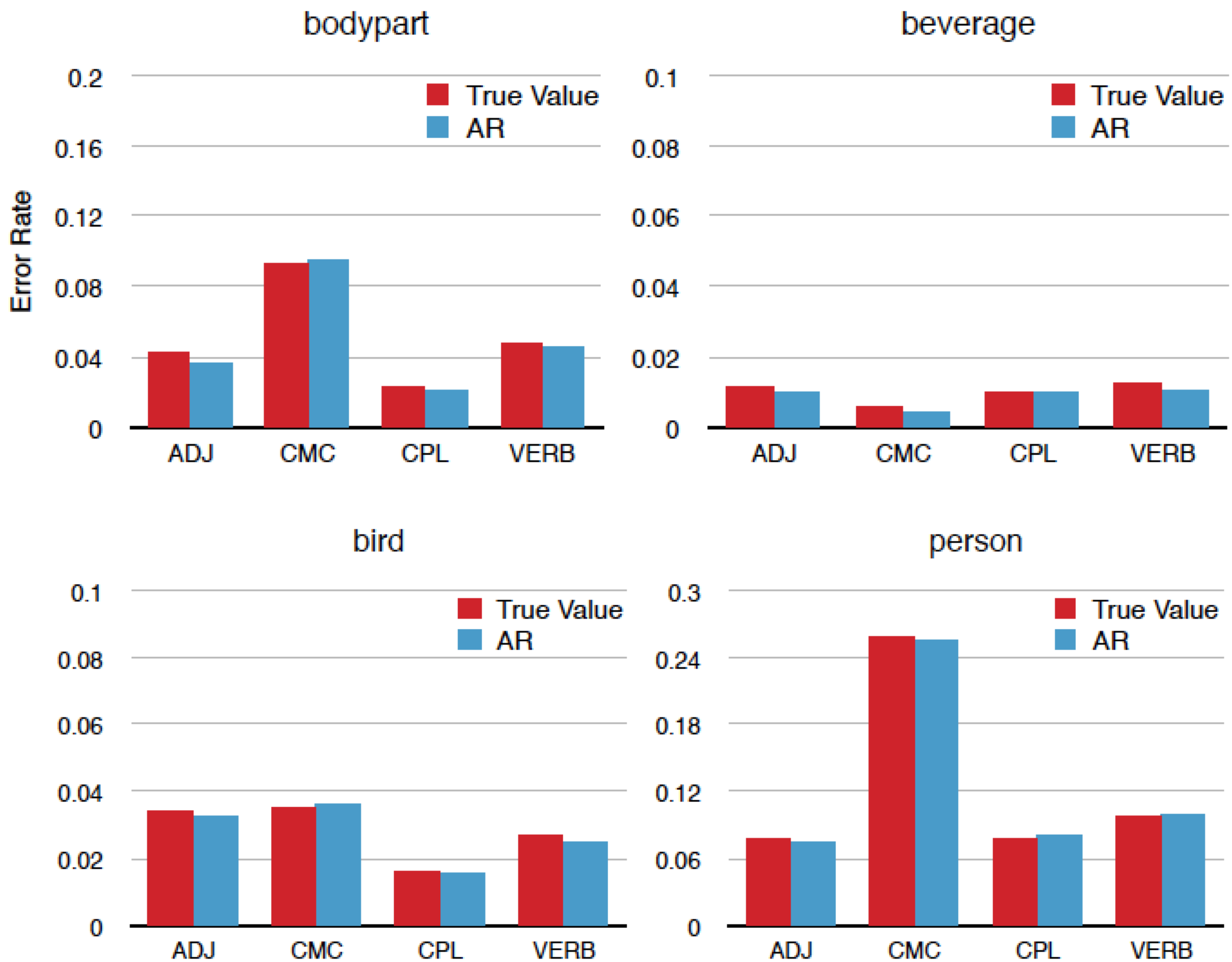
   the more independent, the more probable

$$\boxed{\begin{array}{l} \min \displaystyle\sum_{i,j} (e_{ij} - e_i e_j)^2 \\ \text{such that} \\ \qquad (\forall i, j)\ a_{ij} = 1 - e_i - e_j + 2e_{ij} \end{array}}$$

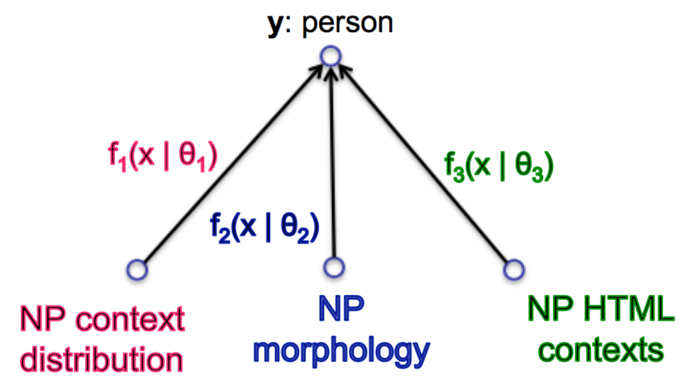
True error (red), estimated error (blue)

[Platanios et al., 2014]

NELL classifiers:



Multiview setting



Given functions f_i: $X_i \rightarrow \{0,1\}$ that

- make independent errors
- are better than chance

If you have at least **2** such functions

- they can be PAC learned by training them to agree over unlabeled data [Blum & Mitchell, 1998]

If you have at least **3** such functions

- their accuracy can be calculated from agreement rates over unlabeled data [Platanios et al., 2014]

Is accuracy estimation strictly harder than learning?

thank you!

follow NELL on Twitter: @CMUNELL
browse/download NELL's KB at http://rtw.ml.cmu.edu