Coupled Bayesian Sets Algorithm for Semi-supervised Learning and Information Extraction

Saurabh Verma
Baranas Hindu University, India

Estevam R. Hruschka Jr.
Federal University of São Carlos, Brazil

ECML/PKDD2012 Bristol, UK September, 26th, 2012

http://rtw.ml.cmu.edu

Read the Web

Research Project at Carnegie Mellon University

Home | Project Overview | Resources & Data | Publications | People

NELL: Never-Ending Language Learning

Can computers learn to read? We think so. "Read the Web" is a research project that attempts to create a computer system that learns over time to read the web. Since January 2010, our computer system called NELL (Never-Ending Language Learner) has been running continuously, attempting to perform two tasks each day:

Browse the Knowledge Base!

- First, it attempts to "read," or extract facts from text found in hundreds of millions of web pages (e.g., `playsInstrument(George_Harrison, guitar)`).
- Second, it attempts to improve its reading competence, so that tomorrow it can extract more facts from the web, more accurately.

So far, NELL has accumulated over 15 million candidate beliefs by reading the web, and it is considering these at different levels of confidence. NELL has high confidence in 1,471,011 of these beliefs — these are displayed on this website. It is not perfect, but NELL is learning. You can track NELL's progress below or @cmunell on Twitter, browse and download its knowledge base, read more about our technical approach, or join the discussion group.

NELL: Never-Ending Language Learner

Inputs:

- initial ontology
- handful of examples of each predicate in ontology
- the web
- occasional interaction with human trainers

The task:

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

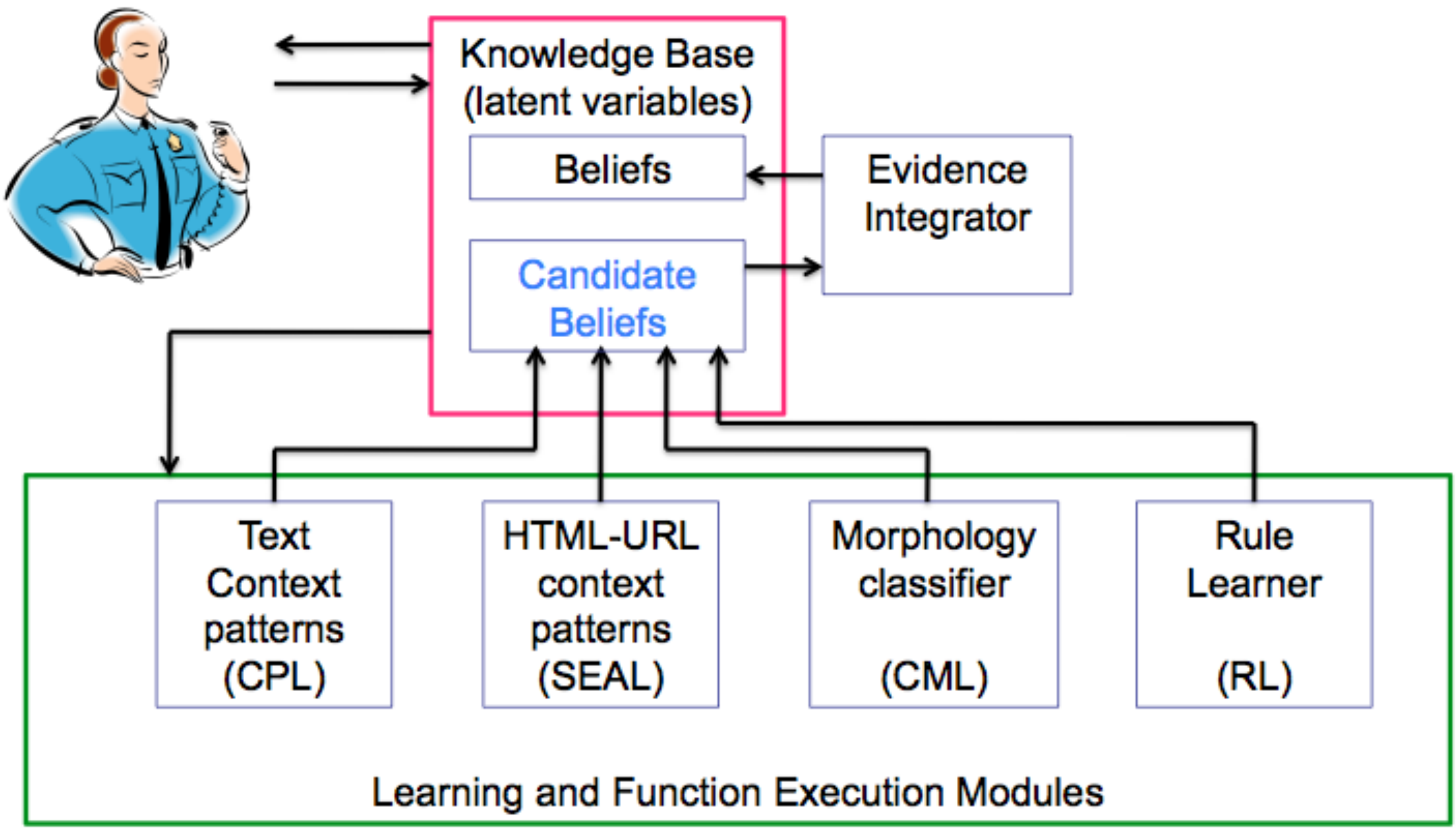
NELL: Never-Ending Language Learner

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

Today...
Running 24 x 7, since January, 2010
Input:
- ontology defining ~800 categories and relations
- 10-20 seed examples of each
- 1 billion web pages (ClueWeb – Jamie Callan)

Result:
- continuously growing KB with +1,300,000 extracted beliefs

NELL Architecture



Bayesian Sets (BS)

Ghahramani & Heller; NIPS 2005

Given $D = \{\mathbf{x}\}$ and $D_c \subset D$, rank the elements of D by how well they would "fit into" a set which includes D_c

Define a score for each $\mathbf{x} \in D$:

$$score(\mathbf{x}) = \frac{p(\mathbf{x} | D_c)}{p(\mathbf{x})}$$

From Bayes rule, the score can be re-written as:

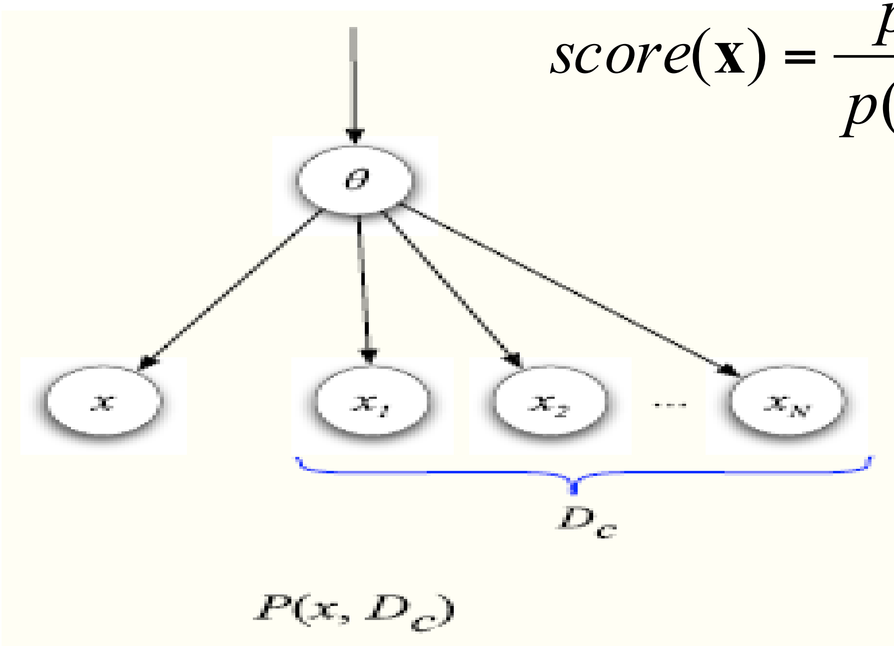
$$score(\mathbf{x}) = \frac{p(\mathbf{x}, D_c)}{p(\mathbf{x})\, p(D_c)}$$

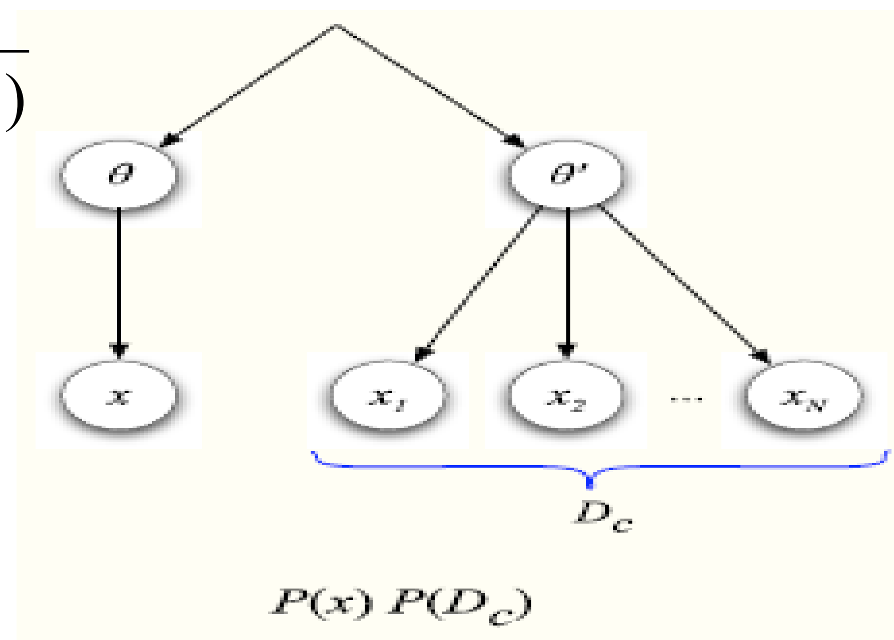
Bayesian Sets (BS)

Ghahramani & Heller; NIPS 2005

Intuitively, the score compares the probability that $\mathbf{x}$ and D_c were generated by the same model with the same unknown parameters θ, to the probability that $\mathbf{x}$ and D_c came from models with different parameters θ and θ'.

$$score(\mathbf{x}) = \frac{p(\mathbf{x}, D_c)}{p(\mathbf{x})\,p(D_c)}$$



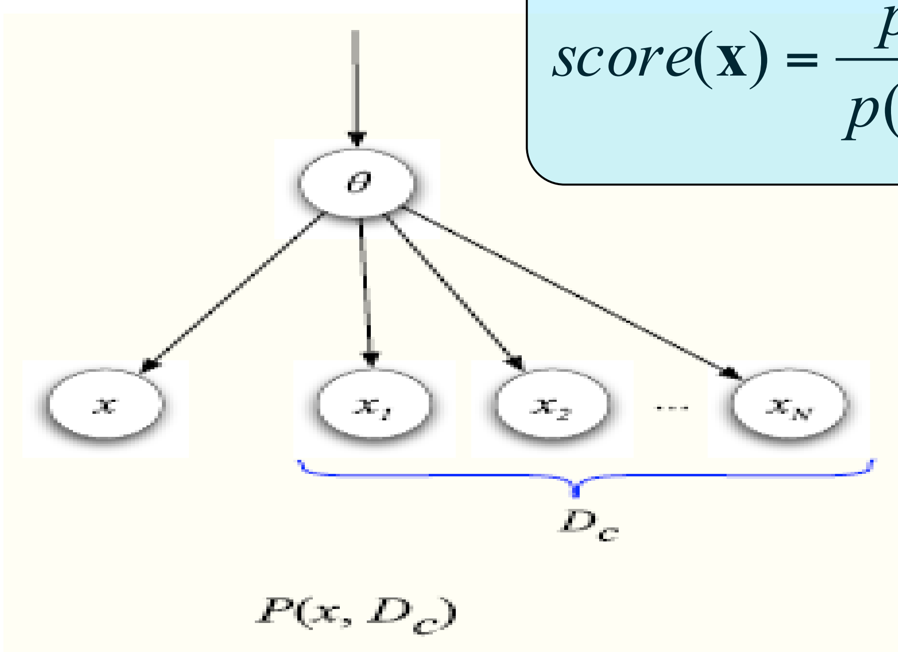


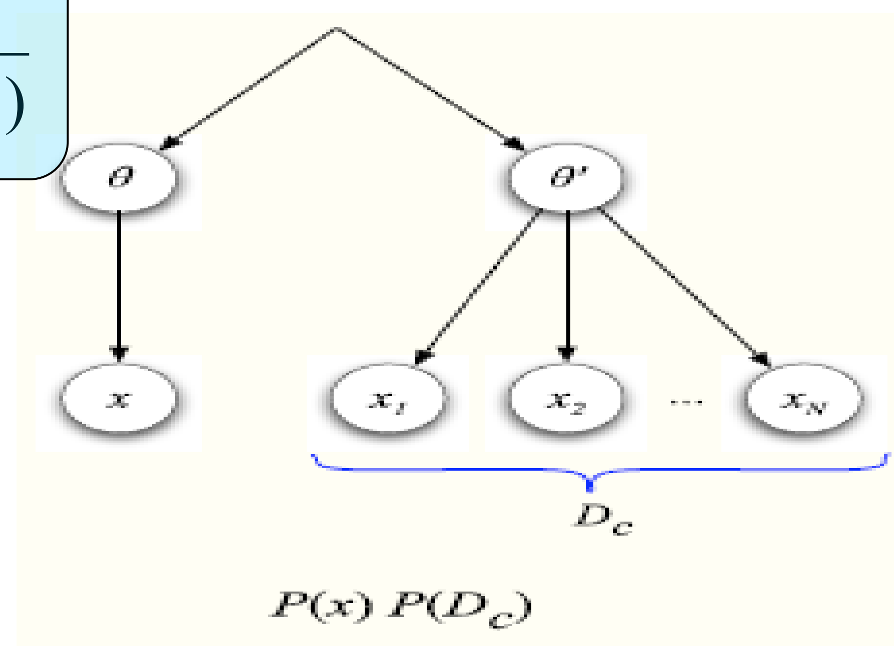
Bayesian Sets (BS)

Ghahramani & Heller; NIPS 2005

Intuitively, the score compares the probability that $\mathbf{x}$ and D_c were generated by the same model with the same unknown parameters θ, to the probability that $\mathbf{x}$ and D_c came from models with different parameters θ and θ’.

$$score(\mathbf{x}) = \frac{p(\mathbf{x}, D_c)}{p(\mathbf{x})\,p(D_c)}$$



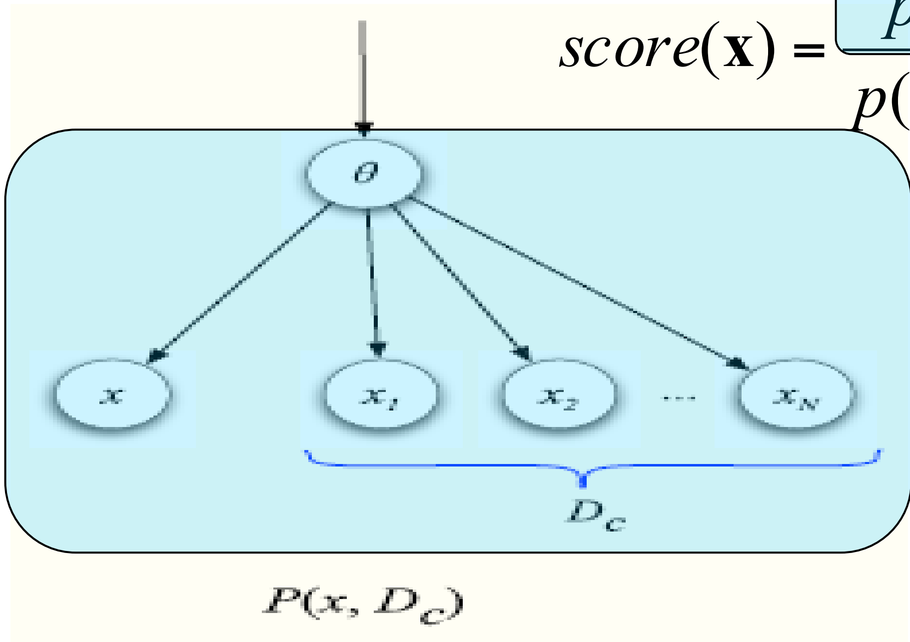


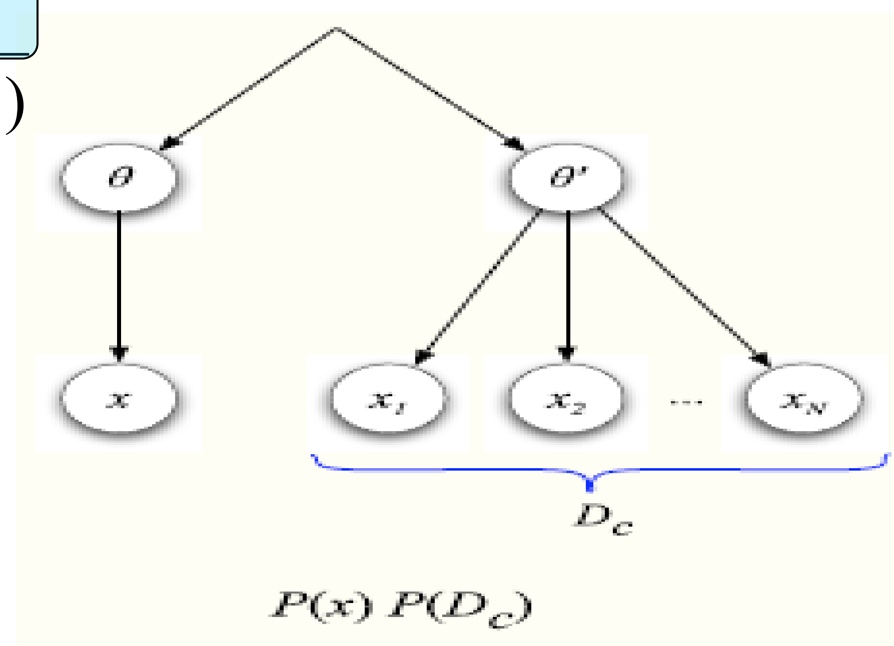
Bayesian Sets (BS)

Ghahramani & Heller; NIPS 2005

Intuitively, the score compares the probability that $\mathbf{x}$ and D_c were generated by the same model with the same unknown parameters θ, to the probability that $\mathbf{x}$ and D_c came from models with different parameters θ and θ’.

$$score(\mathbf{x}) = \frac{p(\mathbf{x}, D_c)}{p(\mathbf{x})\, p(D_c)}$$



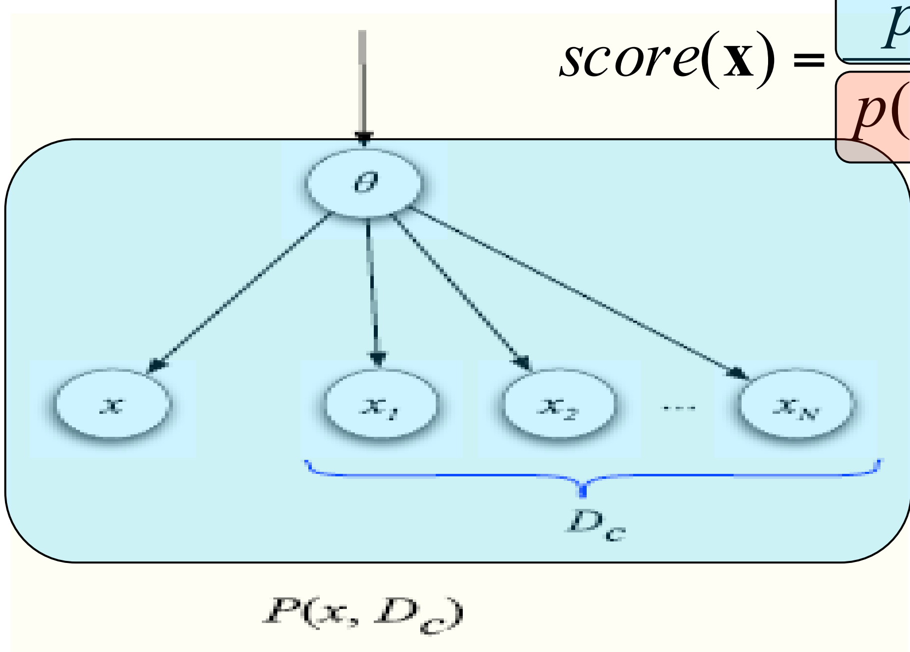


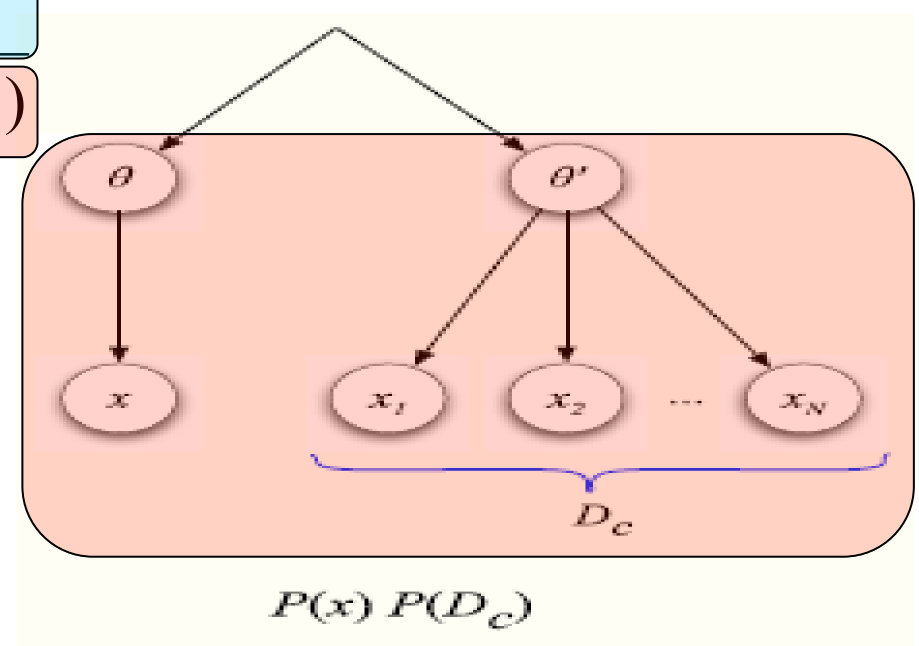
Bayesian Sets (BS)

Ghahramani & Heller; NIPS 2005

Intuitively, the score compares the probability that $\mathbf{x}$ and D_c were generated by the same model with the same unknown parameters θ, to the probability that $\mathbf{x}$ and D_c came from models with different parameters θ and θ'.

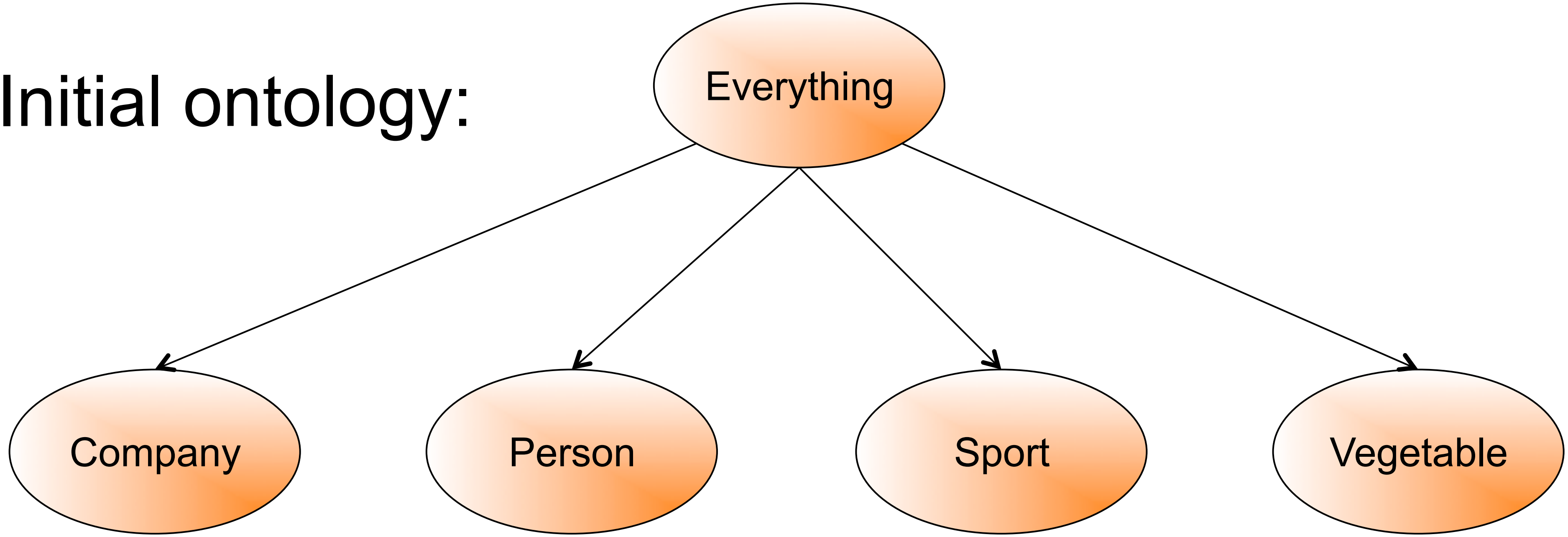
$$score(\mathbf{x}) = \frac{p(\mathbf{x}, D_c)}{p(\mathbf{x})\, p(D_c)}$$





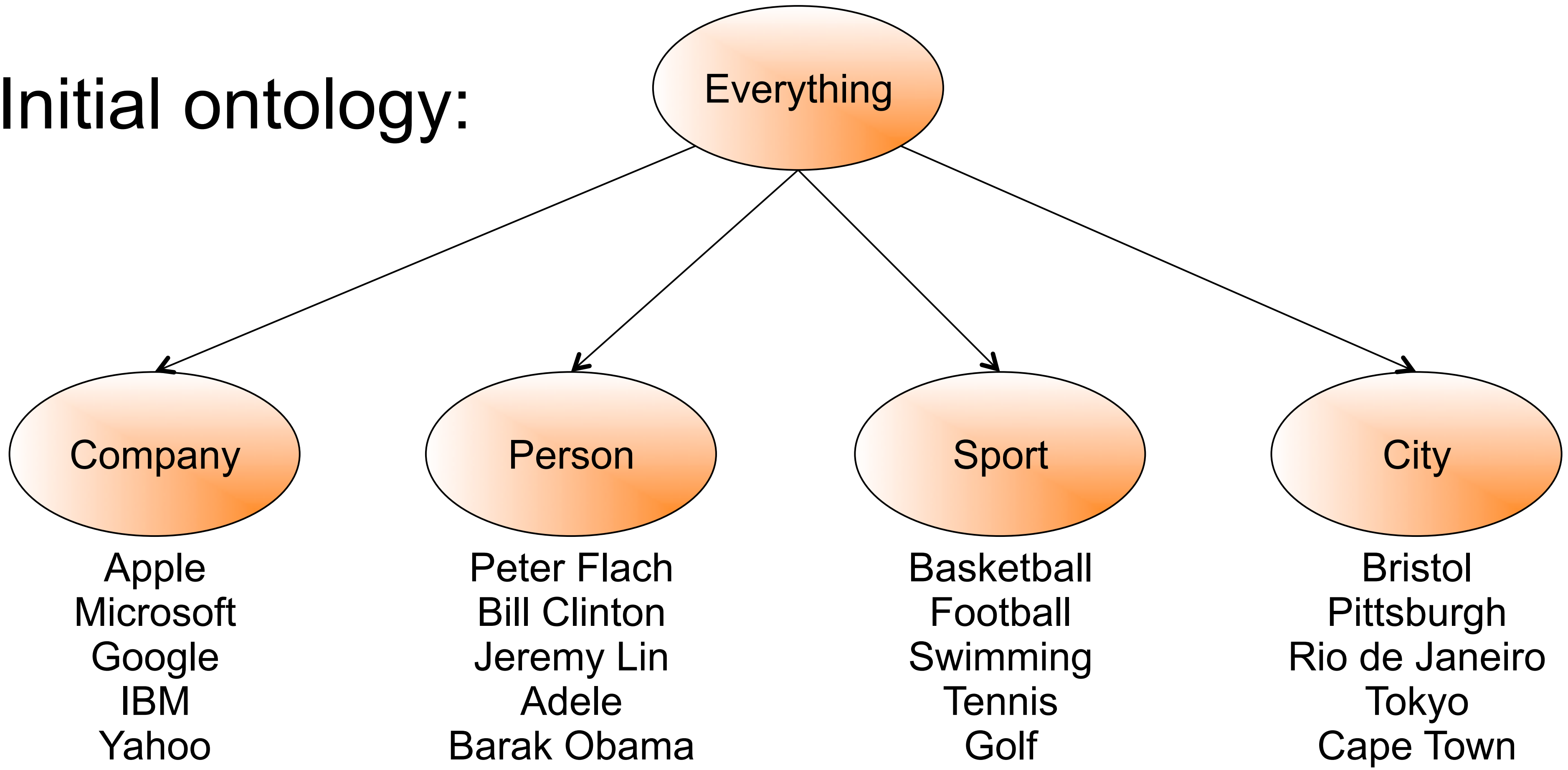
BS using NELL's Ontology

Initial ontology:



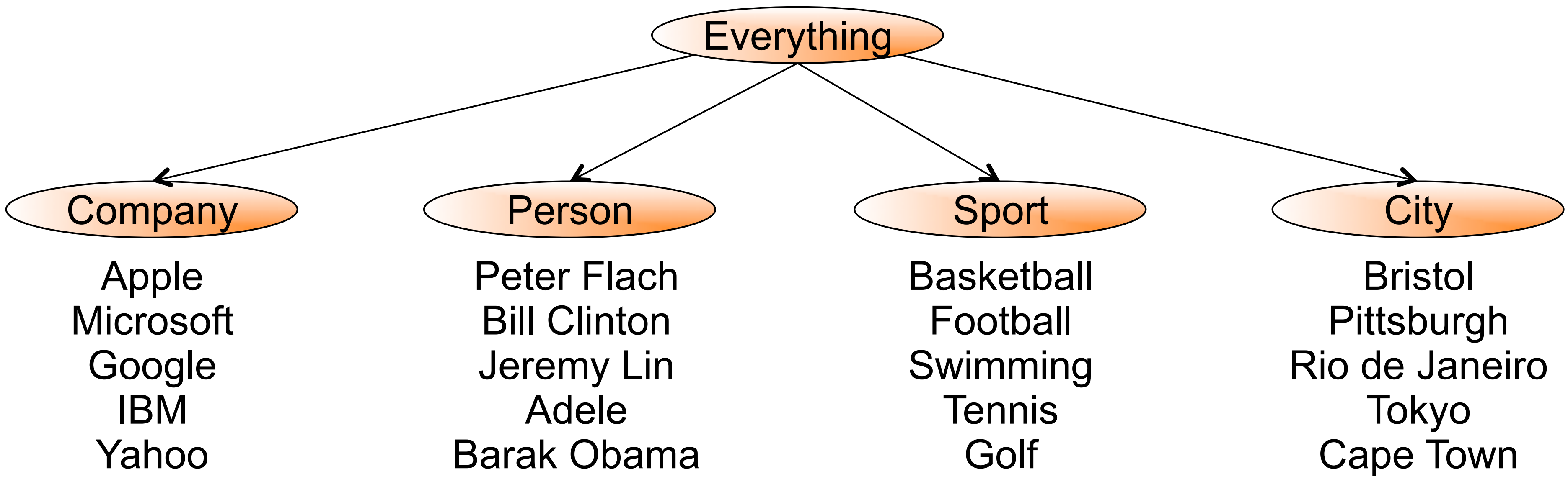
BS using NELL's Ontology

Initial ontology:



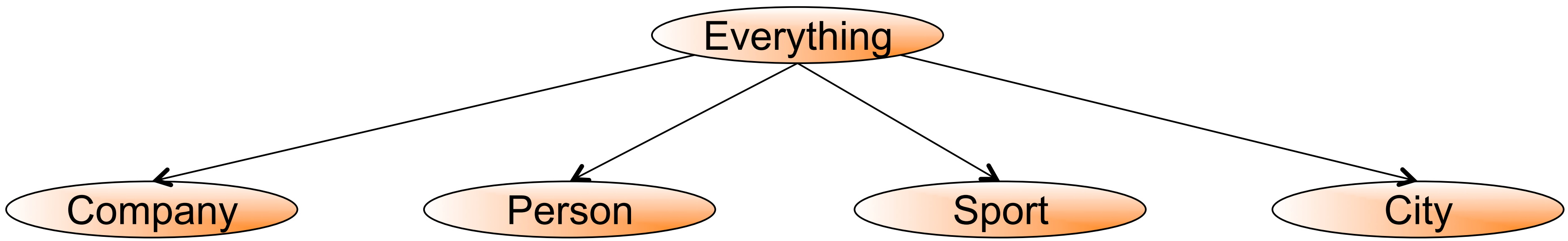
BS using NELL's Ontology

Given a huge web corpus, run BS once



BS using NELL's Ontology

Given a huge web corpus, run BS once



Company	Person	Sport	City
Apple	Peter Flach	Basketball	Bristol
Microsoft	Bill Clinton	Football	Pittsburgh
Google	Jeremy Lin	Swimming	Rio de Janeiro
IBM	Adele	Tennis	Tokyo
Yahoo	Barak Obama	Golf	Cape Town
AT&T	**Dalai Lama**	**Soccer**	**New York**
Boeing	**Freud**	**Volleyball**	**London**
Brazil Telecom	**Tom Mitchell**	**Jogging**	**Sao Paulo**
Texaco	**Aristotle**	**Marathon**	**Brisbane**
Facebook	**Alan Turing**	**Baseball**	**Beijing**
DELL	**Alexander Fleming**	**Badminton**	**Cairo**
…	**…**	**…**	**…**

BS using NELL's Ontology

Table 1. Top 20 instances for Category Sport in the first and second iterations of CBS, BS and Bas-all

Iteration 1			Iteration 2		
CBS	**BS**	**BaS-all**	**CBS**	**BS**	**BaS-all**
Football	Football	football	football	golf	sports
Baseball	Baseball	baseball	Baseball	football	boxing
Basketball	basketball	Basketball	Basketball	baseball	dance
Soccer	Soccer	Soccer	Soccer	soccer	**politics**
Skiing	Skiing	Skiing	Skiing	surfing	fishing
Tennis	Tennis	Tennis	Tennis	skiing	golf
Hockey	Hockey	Hockey	Hockey	cricket	football
Swimming	swimming	Swimming	Swimming	Tennis	baseball
Wrestling	Wrestling	Wrestling	Wrestling	hockey	basketball
Boxing	Boxing	Boxing	Boxing	swimming	soccer
Volleyball	Golf	sport	Volleyball	chess	skiing
Polo	Volleyball	golf	Softball	wrestling	tennis
Badminton	Chess	fishing	Polo	boxing	hockey
Curling	Cricket	chess	Badminton	dancing	chess
table tennis	**Yoga**	cricket	table tennis	**Meditation**	swimming
water polo	surfing	**guitar**	Curling	**cooking**	wrestling
Bocce	**guitar**	dancing	cycling	**piano**	**photography**
Softball	Dancing	hunting	scuba diving	**guitar**	**yoga**
cycling	sailing	sailing	water polo	sailing	**writing**

BS using NELL's Ontology

Table 1. Top 20 instances for Category Sport in the first and second iterations of CBS, BS and Bas-all

Iteration 1			Iteration 2		
CBS	**BS**	**BaS-all**	**CBS**	**BS**	**BaS-all**
Football	Football	football	football	golf	sports
Baseball	Baseball	baseball	Baseball	football	boxing
Basketball	basketball	Basketball	Basketball	baseball	dance
Soccer	Soccer	Soccer	Soccer	soccer	**politics**
Skiing	Skiing	Skiing	Skiing	surfing	fishing
Tennis	Tennis	Tennis	Tennis	skiing	golf
Hockey	Hockey	Hockey	Hockey	cricket	football
Swimming	swimming	Swimming	Swimming	Tennis	baseball
Wrestling	Wrestling	Wrestling	Wrestling	hockey	basketball
Boxing	Boxing	Boxing	Boxing	swimming	soccer
Volleyball	Golf	sport	Volleyball	chess	skiing
Polo	Volleyball	golf	Softball	wrestling	tennis
Badminton	Chess	fishing	Polo	boxing	hockey
Curling	Cricket	chess	Badminton	dancing	chess
table tennis	**Yoga**	cricket	table tennis	**Meditation**	swimming
water polo	surfing	**guitar**	Curling	**cooking**	wrestling
Bocce	**guitar**	dancing	cycling	**piano**	**photography**
Softball	Dancing	hunting	scuba diving	**guitar**	**yoga**
cycling	sailing	sailing	water polo	sailing	**writing**

BS using NELL's Ontology

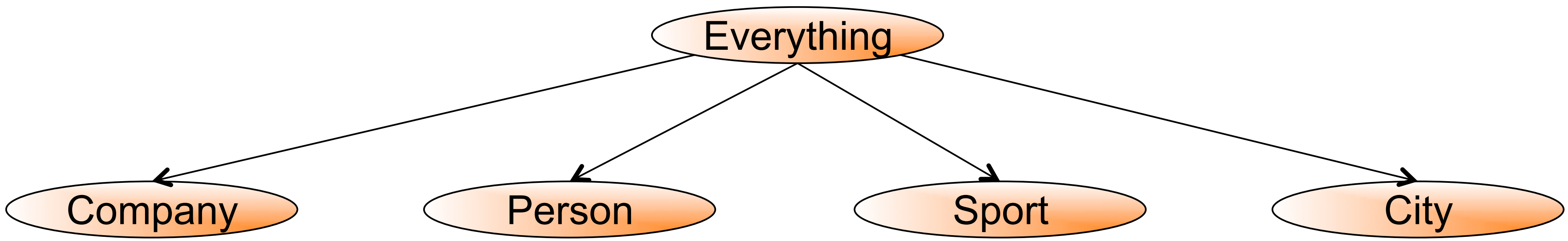
Table 1. Top 20 instances for Category Sport in the first and second iterations of CBS, BS and Bas-all

Iteration 1			Iteration 2		
CBS	**BS**	**BaS-all**	**CBS**	**BS**	**BaS-all**
Football	Football	football	football	golf	sports
Baseball	Baseball	baseball	Baseball	football	boxing
Basketball	basketball	Basketball	Basketball	baseball	dance
Soccer	Soccer	Soccer	Soccer	soccer	**politics**
Skiing	Skiing	Skiing	Skiing	surfing	fishing
Tennis	Tennis	Tennis	Tennis	skiing	golf
Hockey	Hockey	Hockey	Hockey	cricket	football
Swimming	swimming	Swimming	Swimming	Tennis	baseball
Wrestling	Wrestling	Wrestling	Wrestling	hockey	basketball
Boxing	Boxing	Boxing	Boxing	swimming	soccer
Volleyball	Golf	sport	Volleyball	chess	skiing
Polo	Volleyball	golf	Softball	wrestling	tennis
Badminton	Chess	fishing	Polo	boxing	hockey
Curling	Cricket	chess	Badminton	dancing	chess
table tennis	**Yoga**	cricket	table tennis	**Meditation**	swimming
water polo	surfing	**guitar**	Curling	**cooking**	wrestling
Bocce	**guitar**	dancing	cycling	**piano**	**photography**
Softball	Dancing	hunting	scuba diving	**guitar**	**yoga**
cycling	sailing	sailing	water polo	sailing	**writing**

Iterative BS using NELL's Ontology

Zhang & Liu, 2011

Given a huge web corpus, iteratively run BS

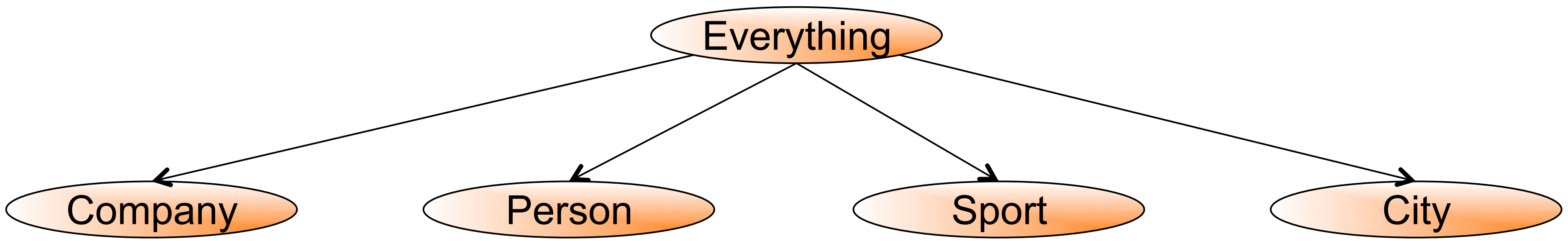


Apple	Peter Flach	Basketball	Bristol
Microsoft	Bill Clinton	Football	Pittsburgh
Google	Jeremy Lin	Swimming	Rio de Janeiro
IBM	Adele	Tennis	Tokyo
Yahoo	Barak Obama	Golf	Cape Town
AT&T	**Dalai Lama**	**Soccer**	**New York**
Boeing	**Freud**	**Volleyball**	**London**

Iterative BS using NELL's Ontology

Zhang & Liu, 2011

Given a huge web corpus, iteratively run BS

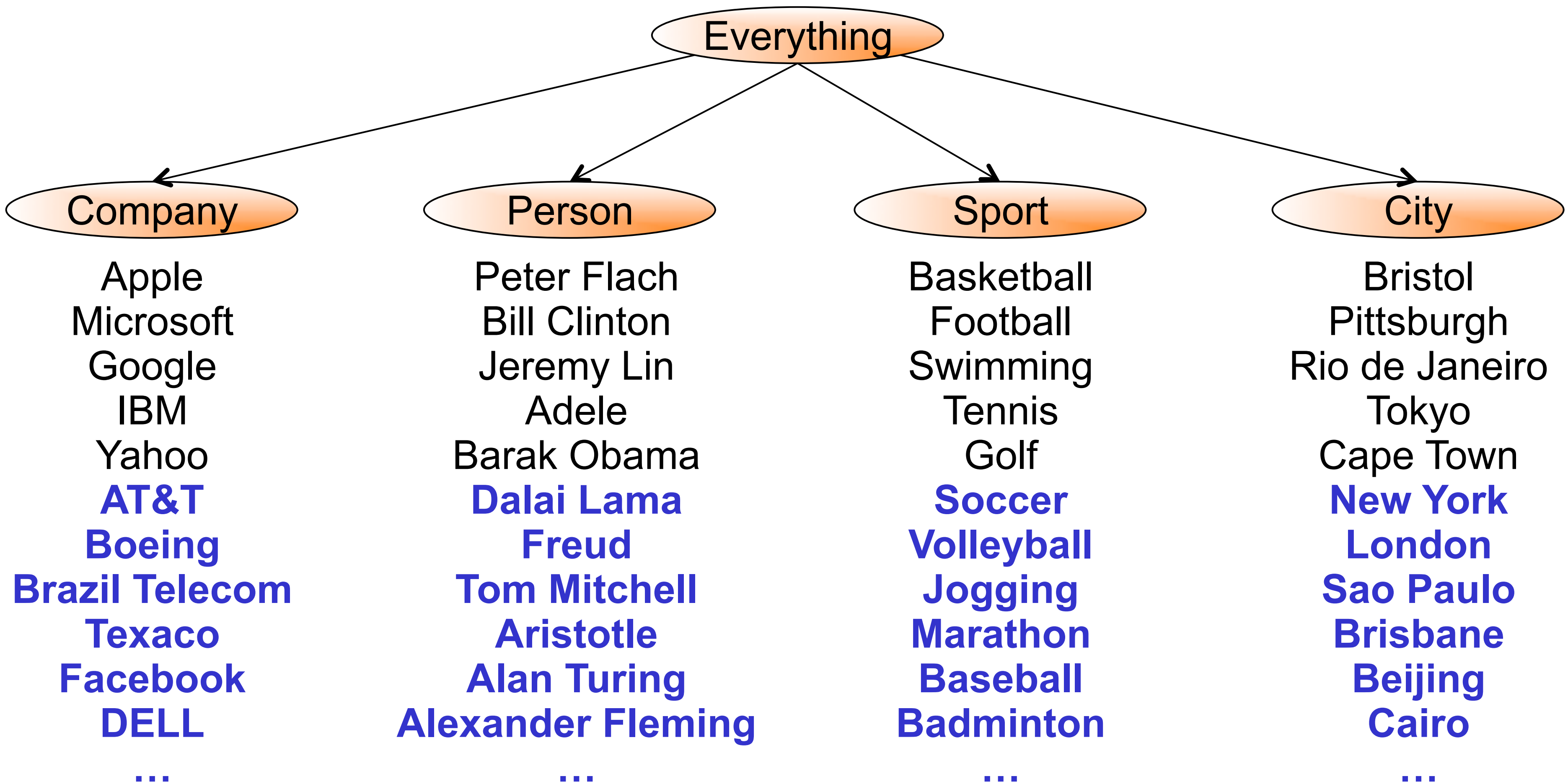


Company	Person	Sport	City
Apple	Peter Flach	Basketball	Bristol
Microsoft	Bill Clinton	Football	Pittsburgh
Google	Jeremy Lin	Swimming	Rio de Janeiro
IBM	Adele	Tennis	Tokyo
Yahoo	Barak Obama	Golf	Cape Town
AT&T	**Dalai Lama**	**Soccer**	**New York**
Boeing	**Freud**	**Volleyball**	**London**
Brazil Telecom	**Tom Mitchell**	**Jogging**	**Sao Paulo**
Texaco	**Aristotle**	**Marathon**	**Brisbane**

Iterative BS using NELL's Ontology

Zhang & Liu, 2011

Given a huge web corpus, iteratively run BS



Iterative BS using NELL's Ontology

Table 1. Top 20 instances for Category Sport in the first and second iterations of CBS, BS and Bas-all

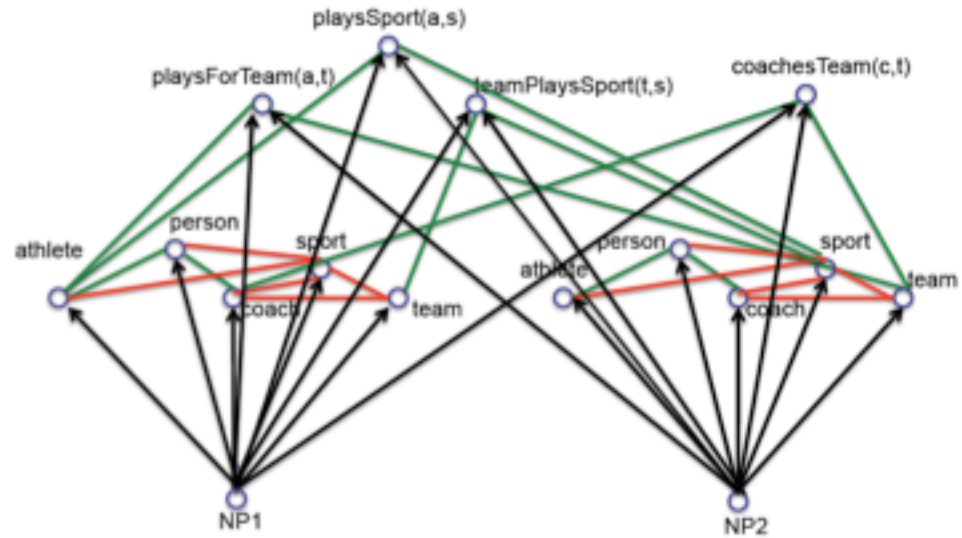
Iteration 1			Iteration 2		
CBS	**BS**	**BaS-all**	**CBS**	**BS**	**BaS-all**
Football	Football	football	football	golf	sports
Baseball	Baseball	baseball	Baseball	football	boxing
Basketball	basketball	Basketball	Basketball	baseball	dance
Soccer	Soccer	Soccer	Soccer	soccer	**politics**
Skiing	Skiing	Skiing	Skiing	surfing	fishing
Tennis	Tennis	Tennis	Tennis	skiing	golf
Hockey	Hockey	Hockey	Hockey	cricket	football
Swimming	swimming	Swimming	Swimming	Tennis	baseball
Wrestling	Wrestling	Wrestling	Wrestling	hockey	basketball
Boxing	Boxing	Boxing	Boxing	swimming	soccer
Volleyball	Golf	sport	Volleyball	chess	skiing
Polo	Volleyball	golf	Softball	wrestling	tennis
Badminton	Chess	fishing	Polo	boxing	hockey
Curling	Cricket	chess	Badminton	dancing	chess
table tennis	**Yoga**	cricket	table tennis	**Meditation**	swimming
water polo	surfing	**guitar**	Curling	**cooking**	wrestling
Bocce	**guitar**	dancing	cycling	**piano**	**photography**
Softball	Dancing	hunting	scuba diving	**guitar**	**yoga**
cycling	sailing	sailing	water polo	sailing	**writing**

Iterative BS using NELL's Ontology

Table 1. Top 20 instances for Category Sport in the first and second iterations of CBS, BS and Bas-all

Iteration 1			Iteration 2		
CBS	**BS**	**BaS-all**	**CBS**	**BS**	**BaS-all**
Football	Football	football	football	golf	sports
Baseball	Baseball	baseball	Baseball	football	boxing
Basketball	basketball	Basketball	Basketball	baseball	dance
Soccer	Soccer	Soccer	Soccer	soccer	politics
Skiing	Skiing	Skiing	Skiing	surfing	fishing
Tennis	Tennis	Tennis	Tennis	skiing	golf
Hockey	Hockey	Hockey	Hockey	cricket	football
Swimming	swimming	Swimming	Swimming	Tennis	baseball
Wrestling	Wrestling	Wrestling	Wrestling	hockey	basketball
Boxing	Boxing	Boxing	Boxing	swimming	soccer
Volleyball	Golf	sport	Volleyball	chess	skiing
Polo	Volleyball	golf	Softball	wrestling	tennis
Badminton	Chess	fishing	Polo	boxing	hockey
Curling	Cricket	chess	Badminton	dancing	chess
table tennis	**Yoga**	cricket	table tennis	**Meditation**	swimming
water polo	surfing	guitar	Curling	**cooking**	wrestling
Bocce	**guitar**	dancing	cycling	**piano**	photography
Softball	Dancing	hunting	scuba diving	**guitar**	**yoga**
cycling	sailing	sailing	water polo	sailing	writing

NELL: Coupled semi-supervised training of many functions



hard (underconstrained) semi-supervised learning problem

much easier (more constrained) semi-supervised learning problem

Coupled Training Type 2:
Structured Outputs, Multitask, Posterior Regularization, Multilabel

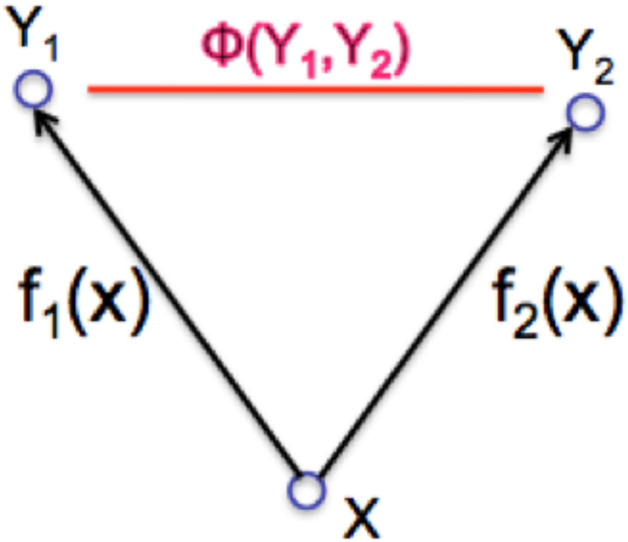
Learn functions with the same input, different outputs, where we know some constraint

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

Coupled Training Type 2:
Structured Outputs, Multitask, Posterior Regularization, Multilabel

Learn functions with the same input, different outputs, where we know some constraint

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

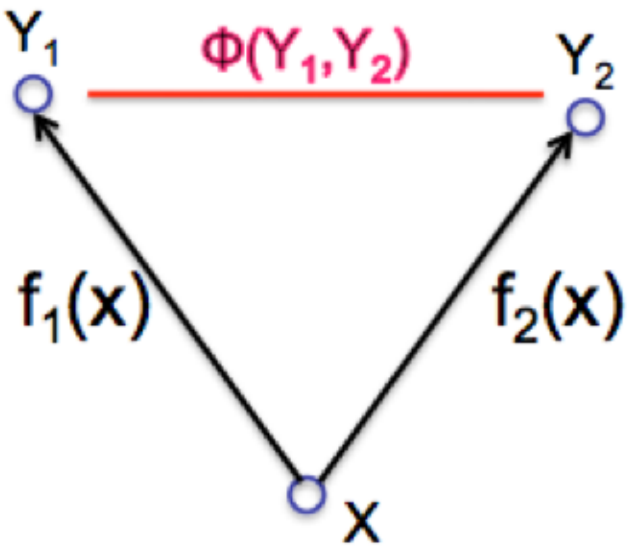


Constraint: $\Phi(f_1(x), f_2(x))$

Coupled Training Type 2:
Structured Outputs, Multitask, Posterior Regularization, Multilabel

Learn functions with the same input, different outputs, where we know some constraint

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



Effectiveness ~ probability that $\Phi(Y_1,Y_2)$ will be violated by incorrect f_j and f_k

Constraint: $\Phi(f_1(x), f_2(x))$

Coupled Bayesian Sets (CBS)

Algorithm 1. Coupled Bayesian Sets algorithm

1: **Input:** An initial ontology O (defining categories, mutually exclusiveness relations and a small set of labeled examples to each category) and a corpus C
2: **Output:** Trusted instances for each given category
3: **for** $i = 0$ to ∞ **do**
4: **for** *each category* **do**
5: extract new instances using available labeled examples
6: filter instances which are violating coupling;
7: rank instances using score
8: label top ranked instances;
9: **end for**
10: **end for**

$$\log score(x) = c + \sum_j q_j^c x_{.j} - \sum_i \sum_j q_j^i x_{.j}$$

Coupled Bayesian Sets (CBS)

Algorithm 1. Coupled Bayesian Sets algorithm

```
Input: An initial ontology O (defining categories, mutually exclusiveness relations
    and a small set of labeled examples to each category) and a corpus C
Output: Trusted instances for each given category
for i = 0 to ∞ do
    for each category do
        extract new instances using available labeled examples
        filter instances which are violating coupling;
        rank instances using score
        label top ranked instances;
    end for
end for
```

$$\log score(x) = c + \sum_j q_j^c x_{.j} - \sum_i \sum_j q_j^i x_{.j}$$

Coupled Bayesian Sets (CBS)

Algorithm 1. Coupled Bayesian Sets algorithm

Input: An initial ontology O (defining categories, mutually exclusiveness relations and a small set of labeled examples to each category) and a corpus C
Output: Trusted instances for each given category
for $i = 0$ to ∞ **do**
  for *each category* **do**
    extract new instances using available labeled examples
    filter instances which are violating coupling;
    rank instances using score
    label top ranked instances;
  end for
end for

$$\log score(x) = c + \sum_j q_j^c x_{.j} - \sum_i \sum_j q_j^i x_{.j}$$

Coupled Bayesian Sets (CBS)

Algorithm 1. Coupled Bayesian Sets algorithm

Input: An initial ontology O (defining categories, mutually exclusiveness relations and a small set of labeled examples to each category) and a corpus C
Output: Trusted instances for each given category
for $i = 0$ to ∞ **do**
    for *each category* **do**
        extract new instances using available labeled examples
        filter instances which are violating coupling;
        rank instances using score
        label top ranked instances;
    end for
end for

$$\log score(x) = c + \sum_j q_j^c x_{.j} - \sum_i \sum_j q_j^i x_{.j}$$

Coupled Bayesian Sets (CBS)

Algorithm 1. Coupled Bayesian Sets algorithm

Input: An initial ontology O (defining categories, mutually exclusiveness relations and a small set of labeled examples to each category) and a corpus C
Output: Trusted instances for each given category
for $i = 0$ to ∞ **do**
  for *each category* **do**
    extract new instances using available labeled examples
    filter instances which are violating coupling;
    rank instances using score
    label top ranked instances;
  end for
end for

$$\boxed{\log score(x) = c + \sum_j q_j^c x_{.j}} - \sum_i \sum_j q_j^i x_{.j}$$

Coupled Bayesian Sets (CBS)

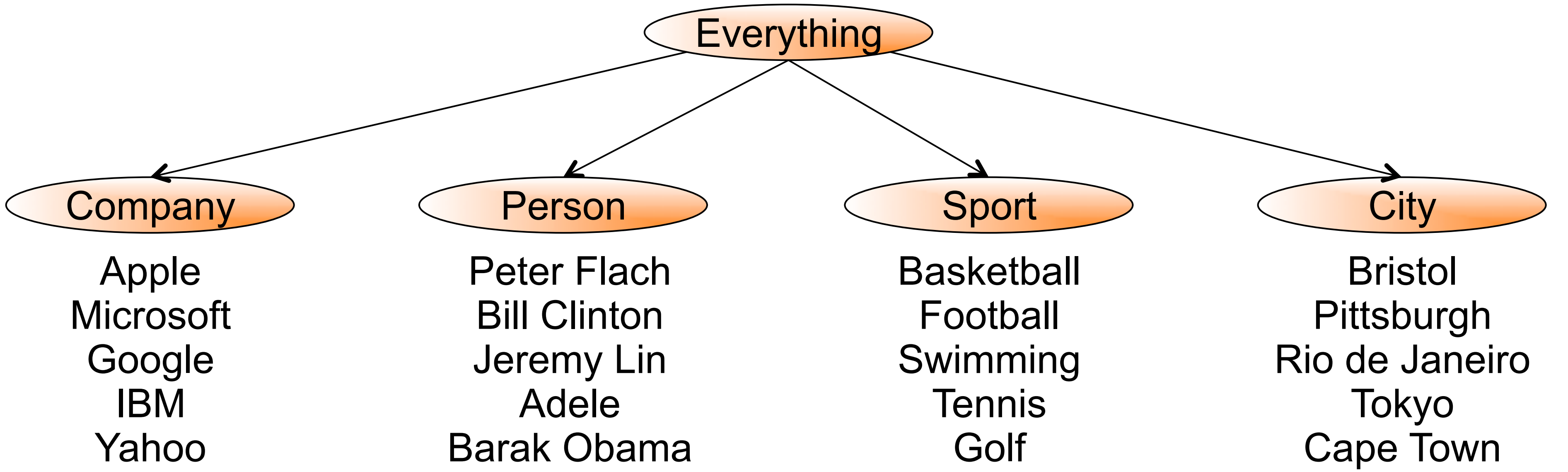
Algorithm 1. Coupled Bayesian Sets algorithm

```
Input: An initial ontology O (defining categories, mutually exclusiveness relations
  and a small set of labeled examples to each category) and a corpus C
Output: Trusted instances for each given category
for i = 0 to ∞ do
    for each category do
        extract new instances using available labeled examples
        filter instances which are violating coupling;
        rank instances using score
        label top ranked instances;
    end for
end for
```

$$\boxed{\log score(x) = c + \sum_j q_j^c x_{\cdot j}} - \boxed{\sum_i \sum_j q_j^i x_{\cdot j}}$$

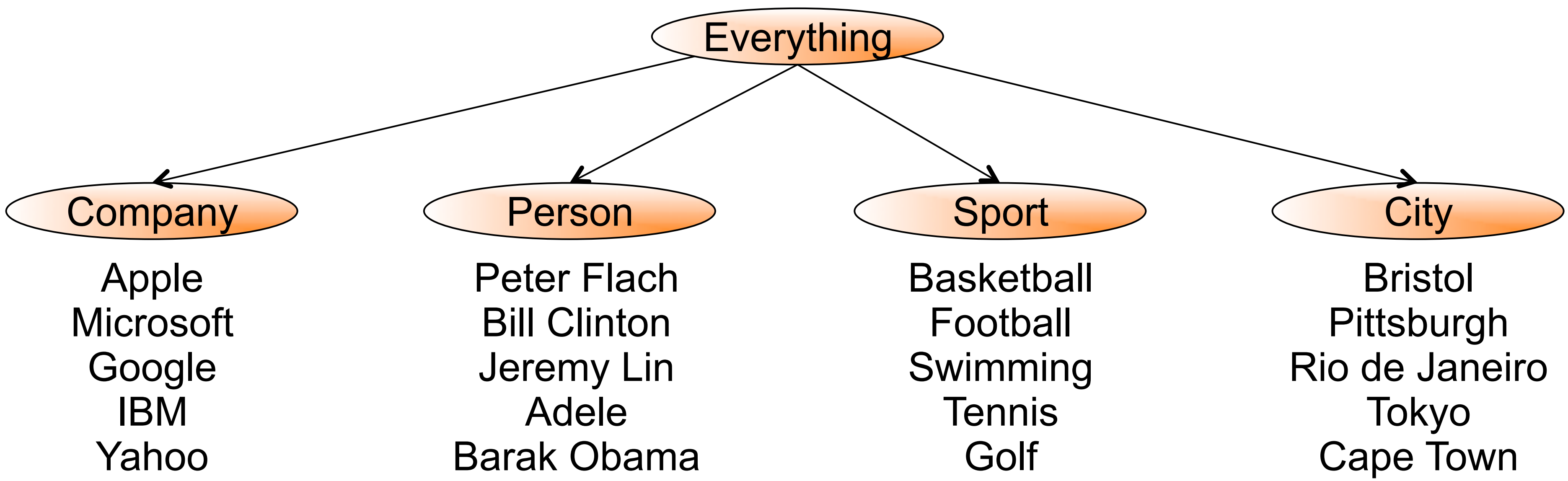
CBS using NELL's Ontology

Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS



CBS using NELL's Ontology

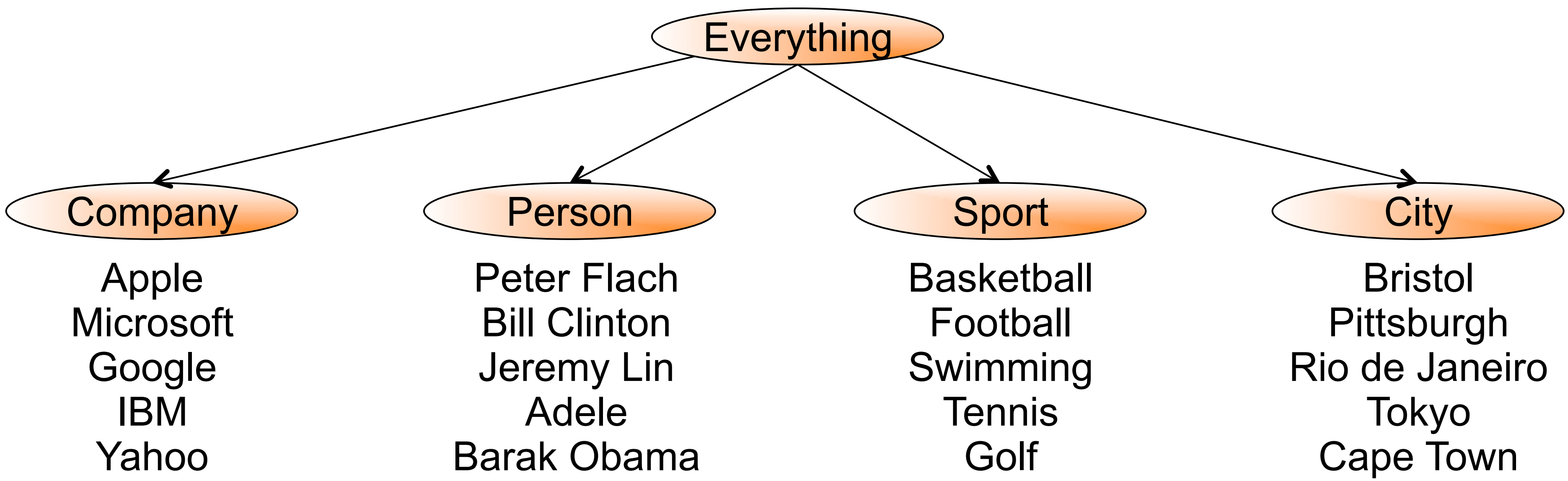
Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS



MutuallyExclusive(Company,Person);
MutuallyExclusive(Company,Sport);
MutuallyExclusive(Company,City);
MutuallyExclusive(Pearson,Sport);
…

CBS using NELL's Ontology

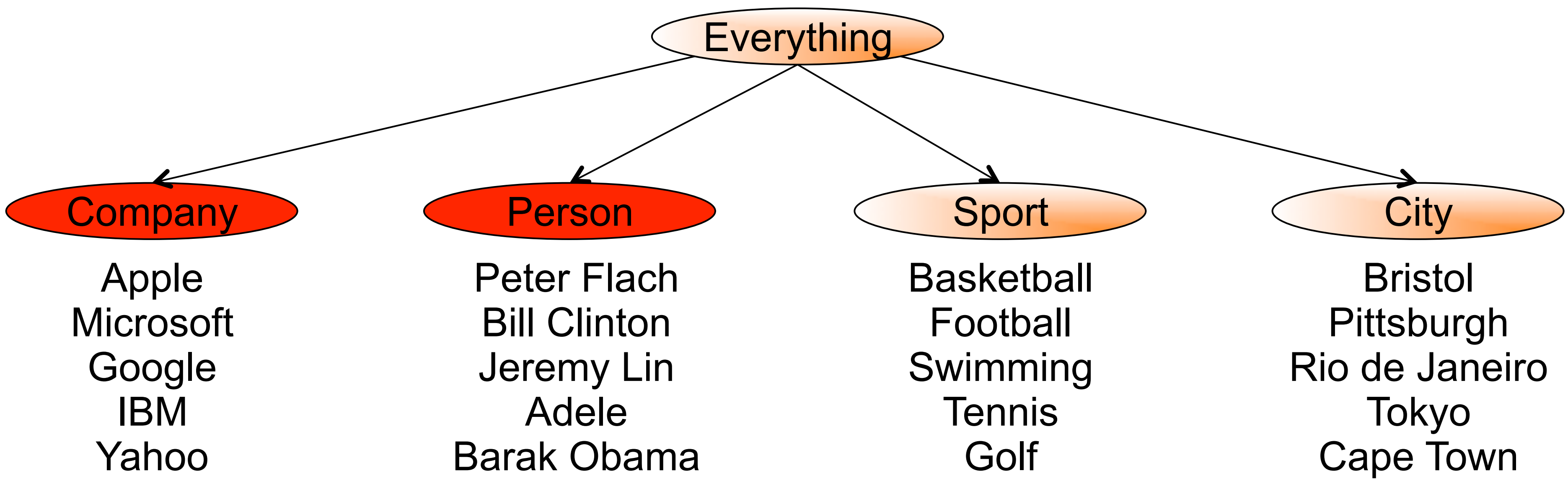
Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS



MutuallyExclusive(Company,Person);
MutuallyExclusive(Company,Sport);
MutuallyExclusive(Company,City);
MutuallyExclusive(Pearson,Sport);
…

CBS using NELL's Ontology

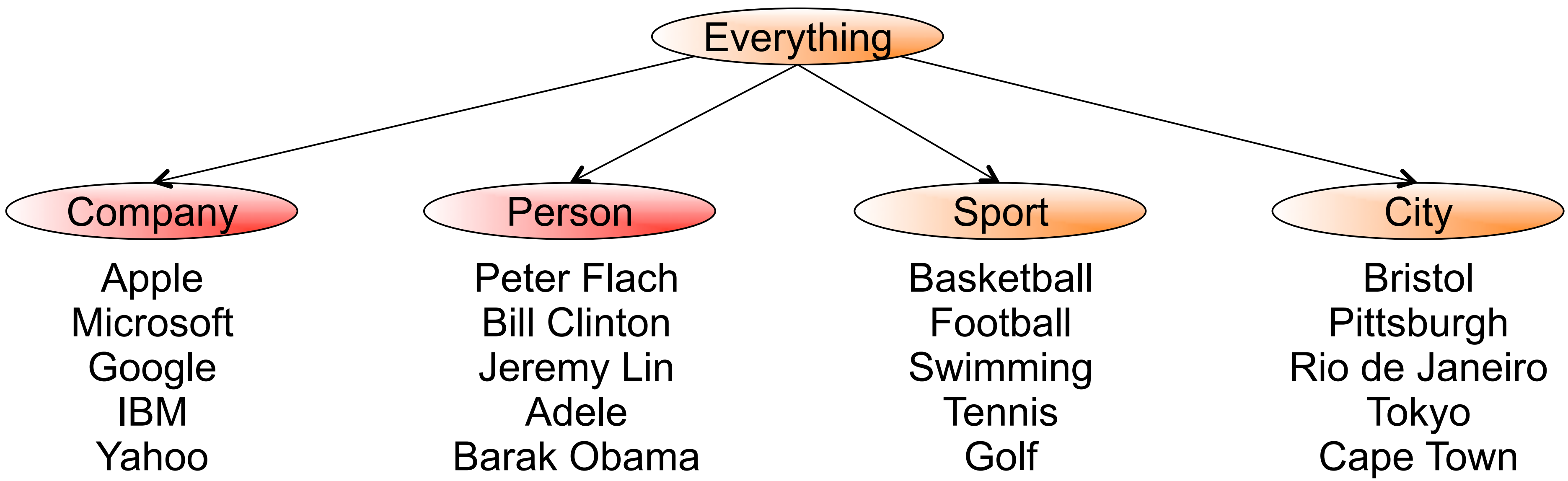
Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS



MutuallyExclusive(Company,Person);
MutuallyExclusive(Company,Sport);
MutuallyExclusive(Company,City);
MutuallyExclusive(Pearson,Sport);
…

CBS using NELL's Ontology

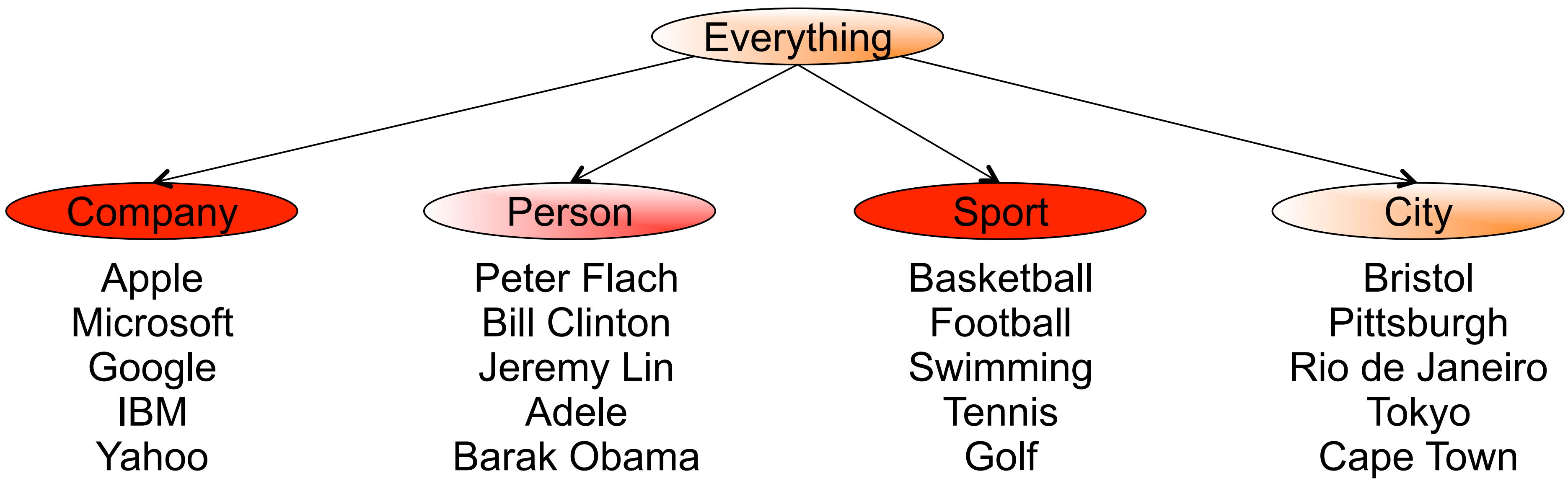
Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS



MutuallyExclusive(Company,Person);
MutuallyExclusive(Company,Sport);
MutuallyExclusive(Company,City);
MutuallyExclusive(Pearson,Sport);
…

CBS using NELL's Ontology

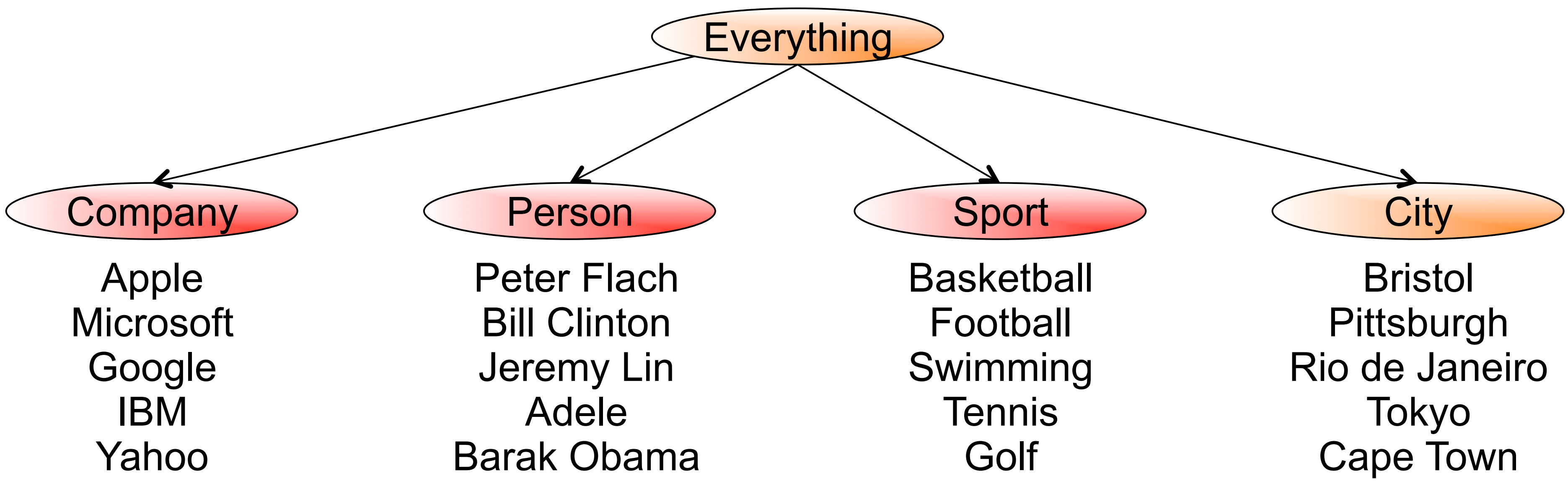
Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS



MutuallyExclusive(Company,Person);
MutuallyExclusive(Company,Sport);
MutuallyExclusive(Company,City);
MutuallyExclusive(Pearson,Sport);
…

CBS using NELL's Ontology

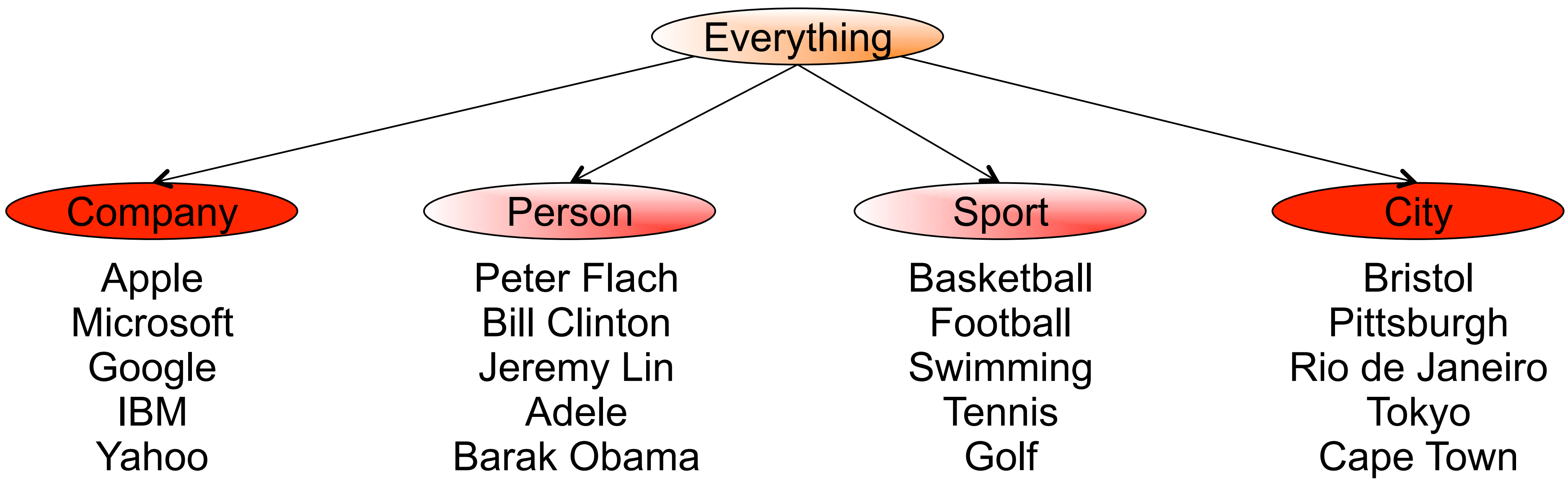
Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS



MutuallyExclusive(Company,Person);
MutuallyExclusive(Company,Sport);
MutuallyExclusive(Company,City);
MutuallyExclusive(Pearson,Sport);
…

CBS using NELL's Ontology

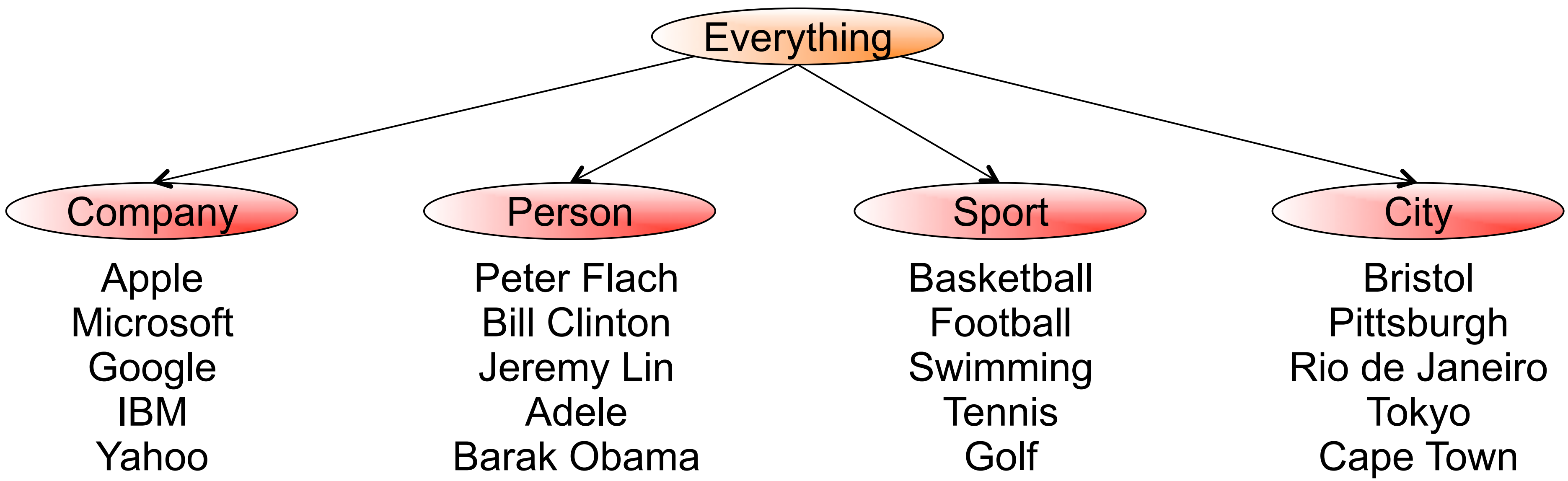
Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS



MutuallyExclusive(Company,Person);
MutuallyExclusive(Company,Sport);
MutuallyExclusive(Company,City);
MutuallyExclusive(Pearson,Sport);
…

CBS using NELL's Ontology

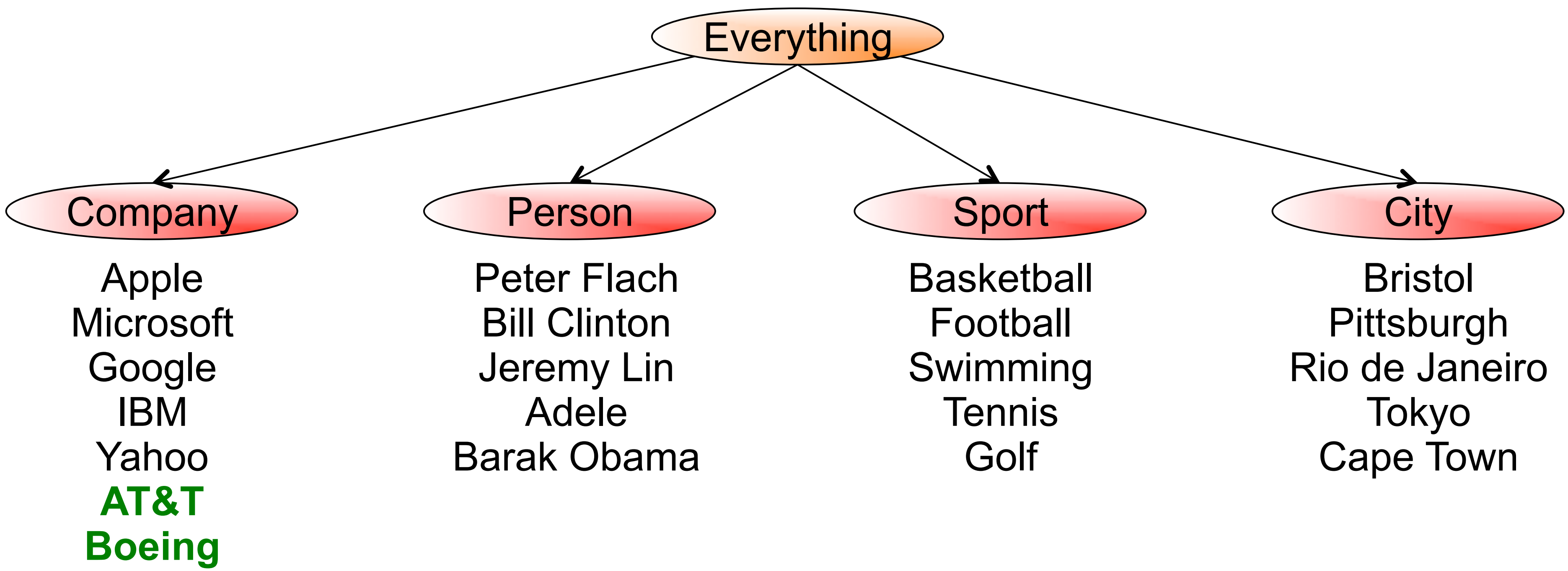
Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS



MutuallyExclusive(Company,Person);
MutuallyExclusive(Company,Sport);
MutuallyExclusive(Company,City);
MutuallyExclusive(Pearson,Sport);
…

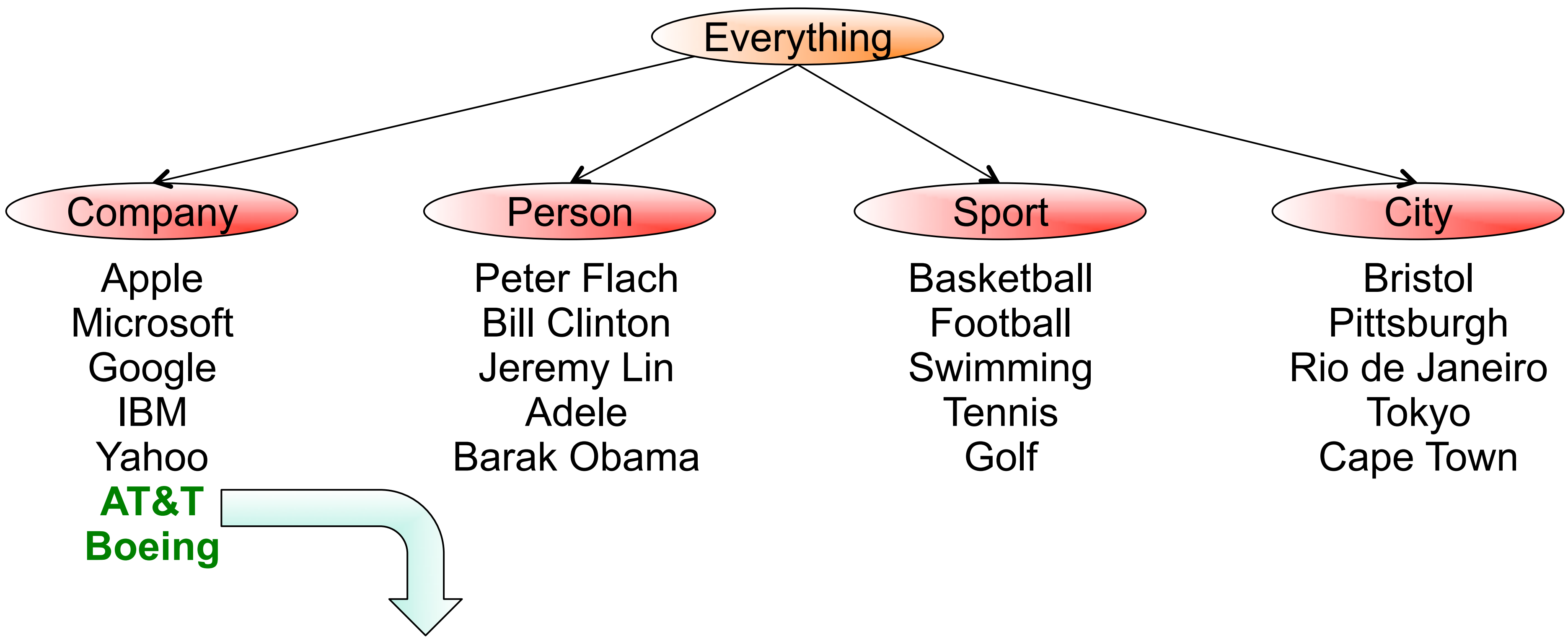
CBS using NELL's Ontology

Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS



CBS using NELL's Ontology

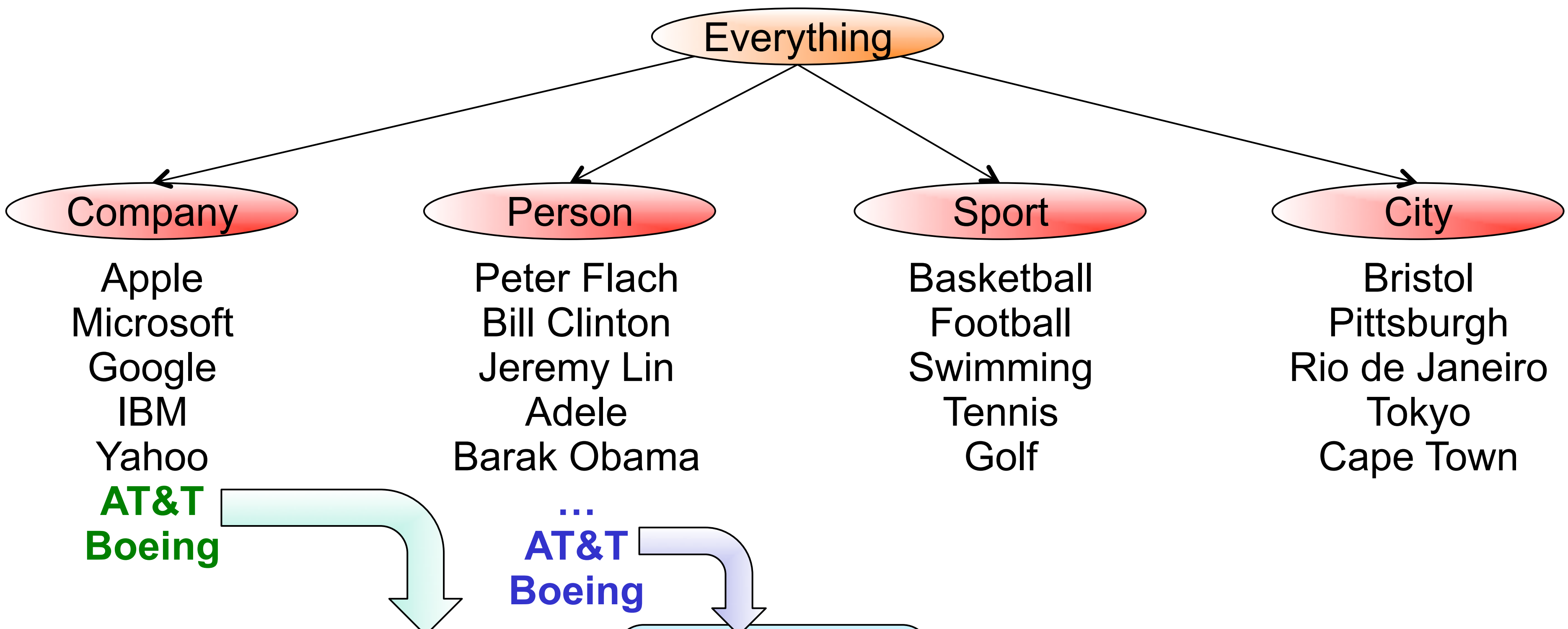
Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS



$$\log score(x) = c + \boxed{\sum_j q_j^c x_{.j}} - \sum_i \sum_j q_j^i x_{.j}$$

CBS using NELL's Ontology

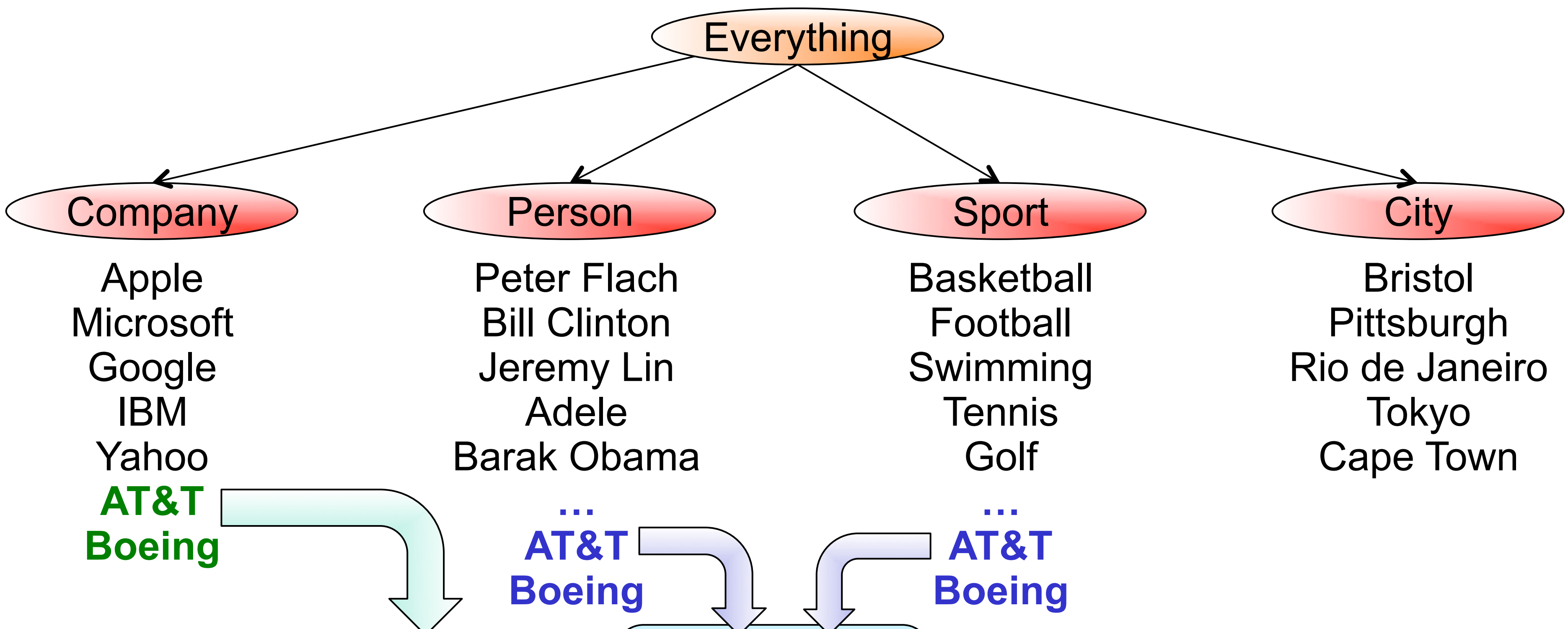
Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS



$$\log score(x) = c + \boxed{\sum_j q_j^c x_{\cdot j}} - \boxed{\sum_i \sum_j q_j^i x_{\cdot j}}$$

CBS using NELL's Ontology

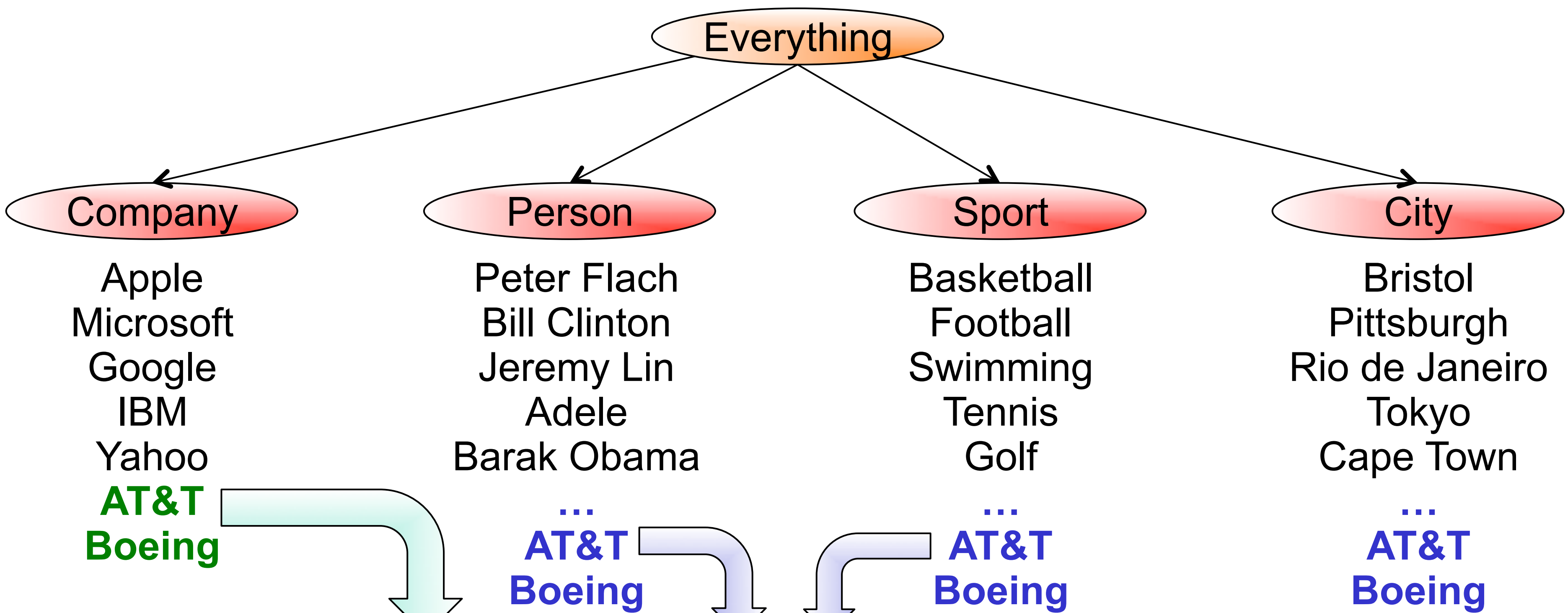
Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS



$$\log score(x) = c + \boxed{\sum_j q_j^c x_{.j}} - \boxed{\sum_i \sum_j q_j^i x_{.j}}$$

CBS using NELL's Ontology

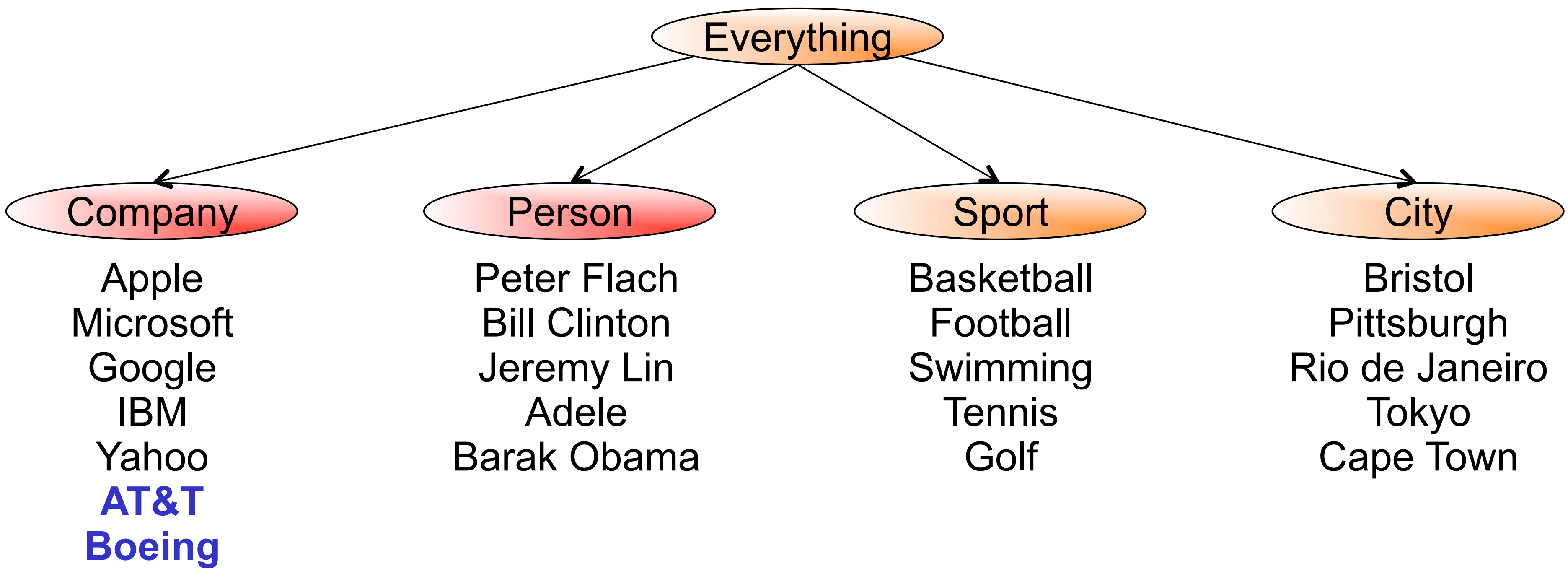
Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS



$$\log score(x) = c + \boxed{\sum_j q_j^c x_{.j}} - \boxed{\sum_i \sum_j q_j^i x_{.j}}$$

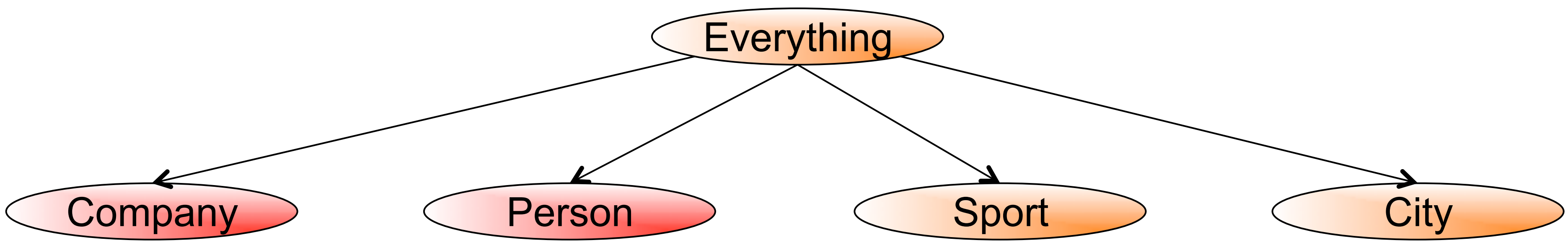
CBS using NELL’s Ontology

Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS



CBS using NELL's Ontology

Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS



Company:
Apple
Microsoft
Google
IBM
Yahoo
AT&T
Boeing
Brazil Telecom
Texaco

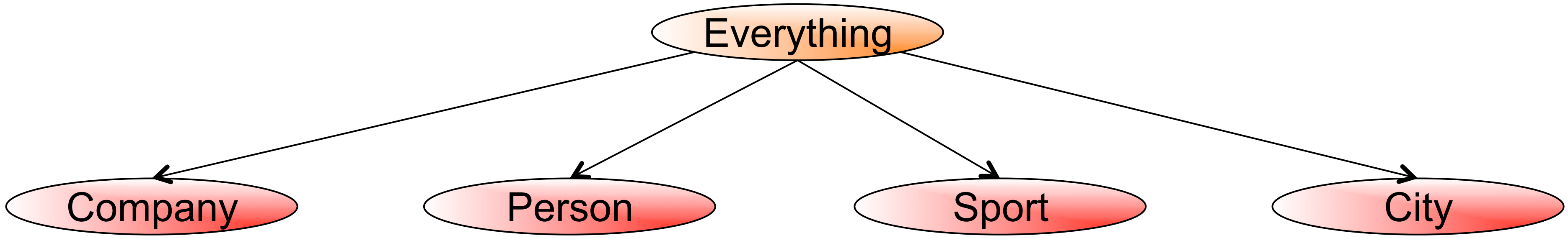
Person:
Peter Flach
Bill Clinton
Jeremy Lin
Adele
Barak Obama

Sport:
Basketball
Football
Swimming
Tennis
Golf

City:
Bristol
Pittsburgh
Rio de Janeiro
Tokyo
Cape Town

CBS using NELL's Ontology

Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS

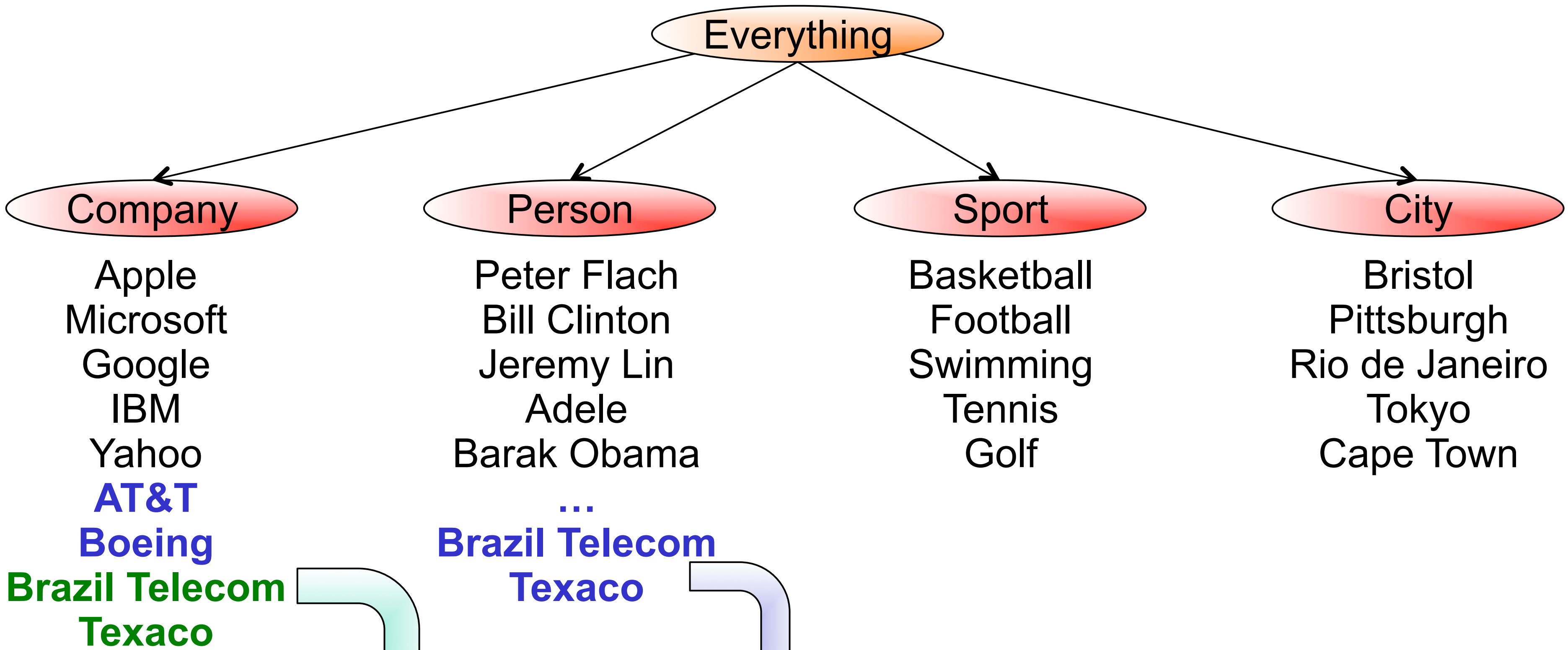


Company	Person	Sport	City
Apple	Peter Flach	Basketball	Bristol
Microsoft	Bill Clinton	Football	Pittsburgh
Google	Jeremy Lin	Swimming	Rio de Janeiro
IBM	Adele	Tennis	Tokyo
Yahoo	Barak Obama	Golf	Cape Town
AT&T			
Boeing			
Brazil Telecom			
Texaco			

$$\log score(x) = c + \boxed{\sum_j q_j^c x_{.j}} - \sum_i \sum_j q_j^i x_{.j}$$

CBS using NELL's Ontology

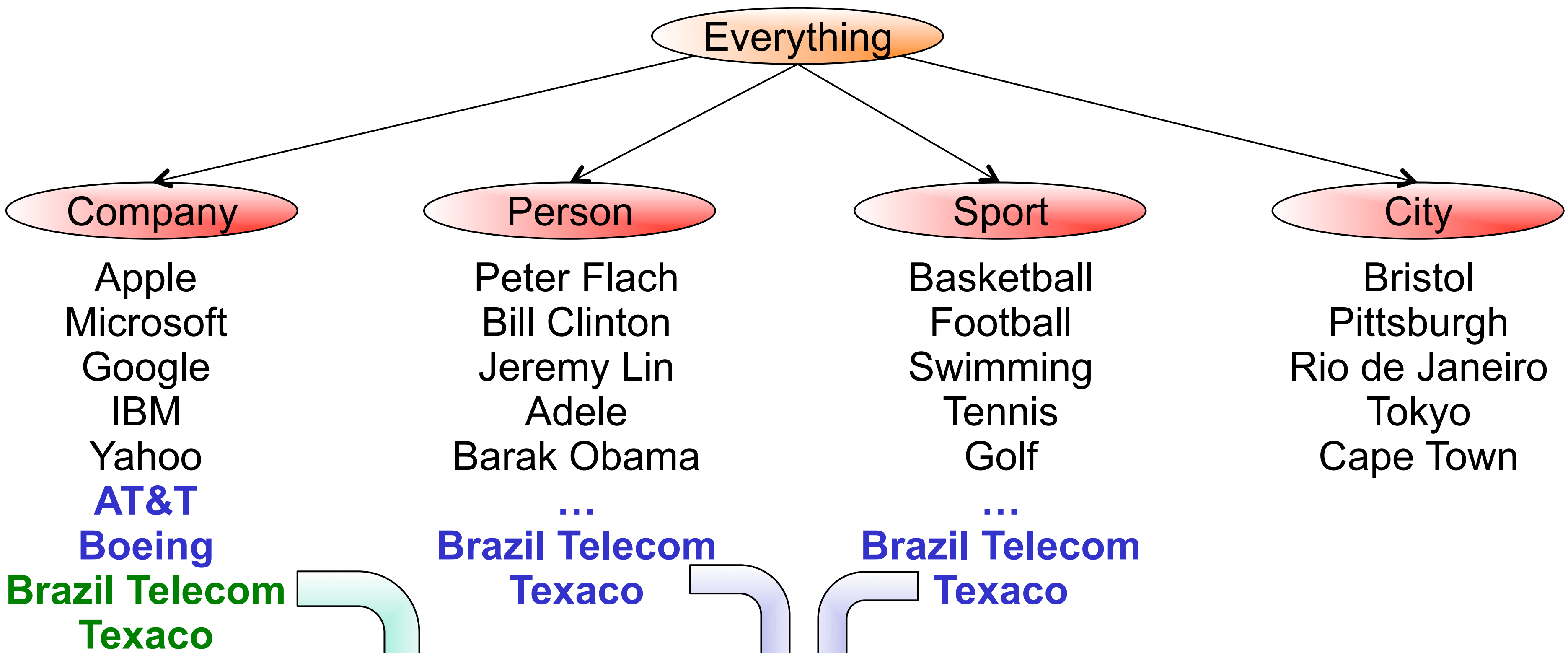
Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS



$$\log score(x) = c + \boxed{\sum_j q_j^c x_{.j}} - \boxed{\sum_i \sum_j q_j^i x_{.j}}$$

CBS using NELL's Ontology

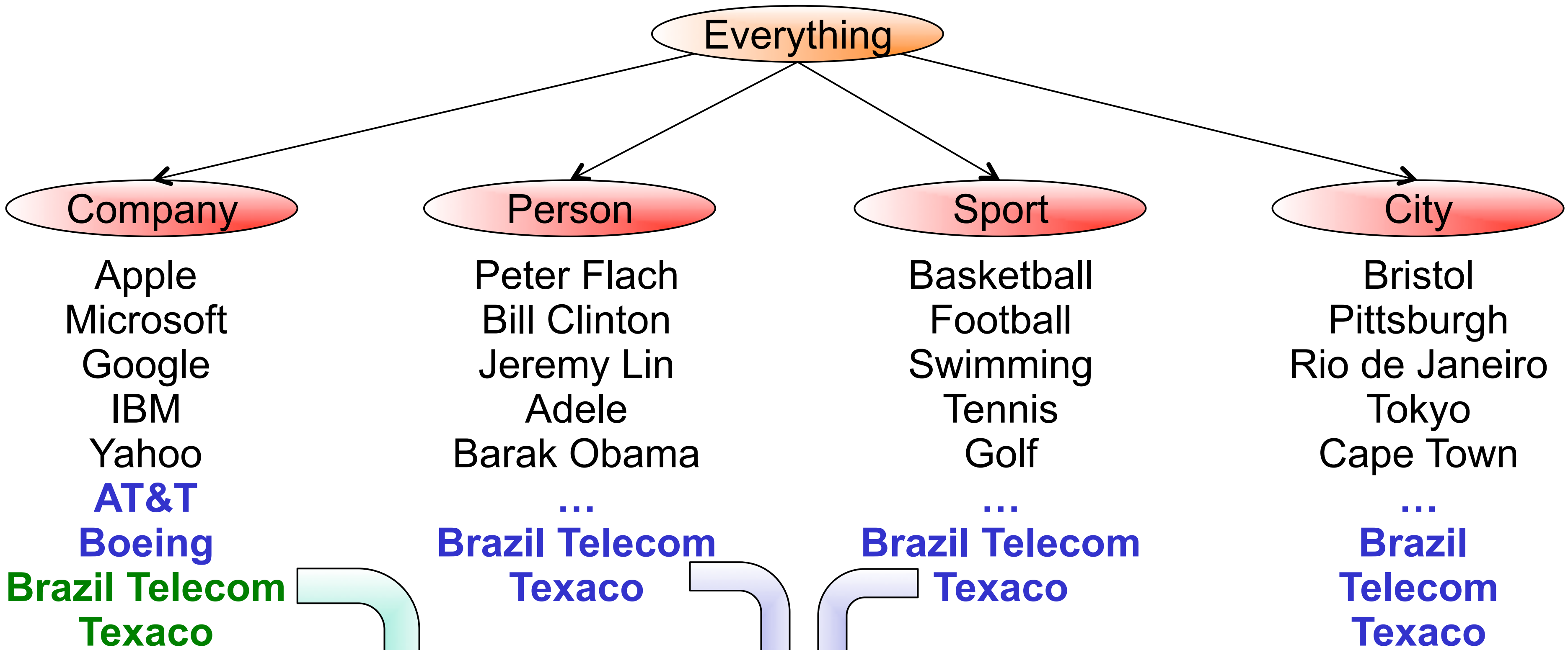
Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS



$$\log score(x) = c + \boxed{\sum_j q_j^c x_{.j}} - \boxed{\sum_i \sum_j q_j^i x_{.j}}$$

CBS using NELL's Ontology

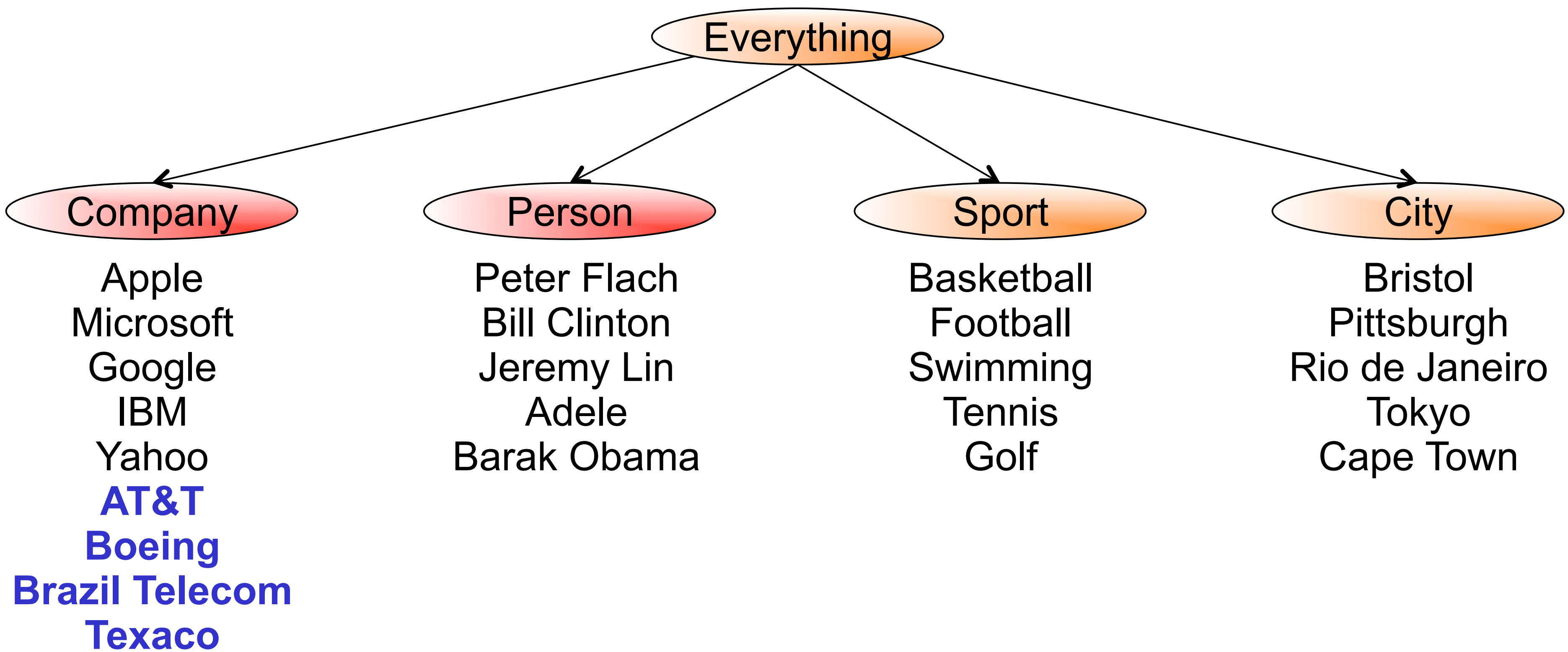
Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS



$$\log score(x) = c + \boxed{\sum_j q_j^c x_{.j}} - \boxed{\sum_i \sum_j q_j^i x_{.j}}$$

CBS using NELL's Ontology

Given a huge web corpus and mutually exclusiveness constraints, iteratively run BS



CBS using NELL's Ontology

Table 1. Top 20 instances for Category Sport in the first and second iterations of CBS, BS and Bas-all

Iteration 1			Iteration 2		
CBS	**BS**	**BaS-all**	**CBS**	**BS**	**BaS-all**
Football	Football	football	football	golf	sports
Baseball	Baseball	baseball	Baseball	football	boxing
Basketball	basketball	Basketball	Basketball	baseball	dance
Soccer	Soccer	Soccer	Soccer	soccer	**politics**
Skiing	Skiing	Skiing	Skiing	surfing	fishing
Tennis	Tennis	Tennis	Tennis	skiing	golf
Hockey	Hockey	Hockey	Hockey	cricket	football
Swimming	swimming	Swimming	Swimming	Tennis	baseball
Wrestling	Wrestling	Wrestling	Wrestling	hockey	basketball
Boxing	Boxing	Boxing	Boxing	swimming	soccer
Volleyball	Golf	sport	Volleyball	chess	skiing
Polo	Volleyball	golf	Softball	wrestling	tennis
Badminton	Chess	fishing	Polo	boxing	hockey
Curling	Cricket	chess	Badminton	dancing	chess
table tennis	**Yoga**	cricket	table tennis	**Meditation**	swimming
water polo	surfing	**guitar**	Curling	**cooking**	wrestling
Bocce	**guitar**	dancing	cycling	**piano**	**photography**
Softball	Dancing	hunting	scuba diving	**guitar**	**yoga**
cycling	sailing	sailing	water polo	sailing	**writing**

CBS using NELL's Ontology

Table 1. Top 20 instances for Category Sport in the first and second iterations of CBS, BS and Bas-all

Iteration 1			Iteration 2		
CBS	**BS**	**BaS-all**	**CBS**	**BS**	**BaS-all**
Football	Football	football	football	golf	sports
Baseball	Baseball	baseball	Baseball	football	boxing
Basketball	basketball	Basketball	Basketball	baseball	dance
Soccer	Soccer	Soccer	Soccer	soccer	**politics**
Skiing	Skiing	Skiing	Skiing	surfing	fishing
Tennis	Tennis	Tennis	Tennis	skiing	golf
Hockey	Hockey	Hockey	Hockey	cricket	football
Swimming	swimming	Swimming	Swimming	Tennis	baseball
Wrestling	Wrestling	Wrestling	Wrestling	hockey	basketball
Boxing	Boxing	Boxing	Boxing	swimming	soccer
Volleyball	Golf	sport	Volleyball	chess	skiing
Polo	Volleyball	golf	Softball	wrestling	tennis
Badminton	Chess	fishing	Polo	boxing	hockey
Curling	Cricket	chess	Badminton	dancing	chess
table tennis	**Yoga**	cricket	table tennis	**Meditation**	swimming
water polo	surfing	**guitar**	Curling	**cooking**	wrestling
Bocce	**guitar**	dancing	cycling	**piano**	**photography**
Softball	Dancing	hunting	scuba diving	**guitar**	**yoga**
cycling	sailing	sailing	water polo	sailing	**writing**

CBS using NELL's Ontology

Table 2. Precision@30 of CBS, BS, CPL and Bas-all after one, three, five and ten iterations

	Precision@30 after Iteration				
Algorithms	1^{st}	3^{rd}	5^{th}	7^{th}	10^{th}
CBS	79%	84%	92%	90%	87%
BS	68	70%	72%	54%	36%
CPL	74%	78%	79%	82%	70%
Bas-all	70%	72%	74%	64%	39%

CBS using NELL's Ontology

Table 2. Precision@30 of CBS, BS, CPL and Bas-all after one, three, five and ten iterations

Algorithms	**Precision@30 after Iteration**				
	1^{st}	3^{rd}	5^{th}	7^{th}	10^{th}
CBS	79%	84%	92%	90%	87%
BS	68	70%	72%	54%	36%
CPL	74%	78%	79%	82%	70%
Bas-all	70%	72%	74%	64%	39%

CBS using NELL's Ontology

Table 2. Precision@30 of CBS, BS, CPL and Bas-all after one, three, five and ten iterations

Algorithms	**Precision@30 after Iteration**				
	1^{st}	3^{rd}	5^{th}	7^{th}	10^{th}
CBS	79%	84%	92%	90%	87%
BS	68	70%	72%	54%	36%
CPL	74%	78%	79%	82%	70%
Bas-all	70%	72%	74%	64%	39%

CBS using NELL's Ontology

Table 2. Precision@30 of CBS, BS, CPL and Bas-all after one, three, five and ten iterations

Algorithms	Precision@30 after Iteration				
	1^{st}	3^{rd}	5^{th}	7^{th}	10^{th}
CBS	79%	84%	92%	90%	87%
BS	68	70%	72%	54%	36%
CPL	74%	78%	79%	82%	70%
Bas-all	70%	72%	74%	64%	39%

CBS using NELL's Ontology

Table 3. Precision@30 for CBS, BS, CPL and Bas-all in all 11 categories (after 5 and 10 iterations)

	Iteration 5				Iteration 10			
Categories	**CBS**	**BS**	**CPL**	**Bas-all**	**CBS**	**BS**	**CPL**	**Bas-all**
Companies	100%	78%	64%	78%	100%	44%	54%	44%
Diseases	100%	84%	100%	84%	100%	48%	74%	54%
KitchenItems	94%	92%	97%	92%	94%	40%	94%	40%
Persons	100%	64%	82%	64%	100%	32%	68%	32%
PhysicsTerms	100%	78%	82%	84%	100%	36%	78%	48%
Plants	100%	68%	94%	74%	100%	38%	84%	32%
Professions	100%	84%	84%	84%	87%	54%	87%	54%
SocioPolitics	48%	30%	38%	30%	34%	18%	28%	14%
Sports	97%	84%	90%	84%	100%	43%	87%	54%
Websites	94%	64%	67%	74%	90%	36%	58%	36%
Vegetables	83%	72%	78%	64%	48%	14%	54%	14%
Average Precision@30	**92%**	**72%**	**79%**	**74%**	**87%**	**36%**	**70%**	**39%**

CBS using NELL's Ontology

Table 3. Precision@30 for CBS, BS, CPL and Bas-all in all 11 categories (after 5 and 10 iterations)

	Iteration 5				Iteration 10			
Categories	**CBS**	**BS**	**CPL**	**Bas-all**	**CBS**	**BS**	**CPL**	**Bas-all**
Companies	100%	78%	64%	78%	100%	44%	54%	44%
Diseases	100%	84%	100%	84%	100%	48%	74%	54%
KitchenItems	94%	92%	97%	92%	94%	40%	94%	40%
Persons	100%	64%	82%	64%	100%	32%	68%	32%
PhysicsTerms	100%	78%	82%	84%	100%	36%	78%	48%
Plants	100%	68%	94%	74%	100%	38%	84%	32%
Professions	100%	84%	84%	84%	87%	54%	87%	54%
SocioPolitics	48%	30%	38%	30%	34%	18%	28%	14%
Sports	97%	84%	90%	84%	100%	43%	87%	54%
Websites	94%	64%	67%	74%	90%	36%	58%	36%
Vegetables	83%	72%	78%	64%	48%	14%	54%	14%
Average Precision@30	**92%**	**72%**	**79%**	**74%**	**87%**	**36%**	**70%**	**39%**

CBS using NELL's Ontology

Table 3. Precision@30 for CBS, BS, CPL and Bas-all in all 11 categories (after 5 and 10 iterations)

	Iteration 5				Iteration 10			
Categories	**CBS**	**BS**	**CPL**	**Bas-all**	**CBS**	**BS**	**CPL**	**Bas-all**
Companies	100%	78%	64%	78%	100%	44%	54%	44%
Diseases	100%	84%	100%	84%	100%	48%	74%	54%
KitchenItems	94%	92%	97%	92%	94%	40%	94%	40%
Persons	100%	64%	82%	64%	100%	32%	68%	32%
PhysicsTerms	100%	78%	82%	84%	100%	36%	78%	48%
Plants	100%	68%	94%	74%	100%	38%	84%	32%
Professions	100%	84%	84%	84%	87%	54%	87%	54%
SocioPolitics	48%	30%	38%	30%	34%	18%	28%	14%
Sports	97%	84%	90%	84%	100%	43%	87%	54%
Websites	94%	64%	67%	74%	90%	36%	58%	36%
Vegetables	83%	72%	78%	64%	48%	14%	54%	14%
Average Precision@30	**92%**	**72%**	**79%**	**74%**	**87%**	**36%**	**70%**	**39%**

CBS using NELL's Ontology

Table 3. Precision@30 for CBS, BS, CPL and Bas-all in all 11 categories (after 5 and 10 iterations)

	Iteration 5				Iteration 10			
Categories	**CBS**	**BS**	**CPL**	**Bas-all**	**CBS**	**BS**	**CPL**	**Bas-all**
Companies	100%	78%	64%	78%	100%	44%	54%	44%
Diseases	100%	84%	100%	84%	100%	48%	74%	54%
KitchenItems	94%	92%	97%	92%	94%	40%	94%	40%
Persons	100%	64%	82%	64%	100%	32%	68%	32%
PhysicsTerms	100%	78%	82%	84%	100%	36%	78%	48%
Plants	100%	68%	94%	74%	100%	38%	84%	32%
Professions	100%	84%	84%	84%	87%	54%	87%	54%
SocioPolitics	48%	30%	38%	30%	34%	18%	28%	14%
Sports	97%	84%	90%	84%	100%	43%	87%	54%
Websites	94%	64%	67%	74%	90%	36%	58%	36%
Vegetables	83%	72%	78%	64%	48%	14%	54%	14%
Average Precision@30	**92%**	**72%**	**79%**	**74%**	**87%**	**36%**	**70%**	**39%**

CBS using NELL's Ontology

Table 6. CPL probability and CBS score for extracted instances (after 5 iterations) for category Sport

CPL	probability	CBS	score
Game	0.998047	Baseball	1782.201
Show	0.998047	Basketball	1630.333
Football	0.998047	Soccer	1223.195
Day	0.998047	Skiing	1162.535
Drama	0.996094	Tennis	1022.093
Music	0.996094	Hockey	1012.905
Basketball	0.996094	Sailing	984.733
chess	0.992188	Wrestling	802.307
Baseball	0.992188	Boxing	724.129
Golf	0.992188	Swimming	677.489

CBS using NELL's Ontology

Table 6. CPL probability and CBS score for extracted instances (after 5 iterations) for category Sport

CPL	probability	CBS	score
Game	0.998047	Baseball	1782.201
Show	0.998047	Basketball	1630.333
Football	0.998047	Soccer	1223.195
Day	0.998047	Skiing	1162.535
Drama	0.996094	Tennis	1022.093
Music	0.996094	Hockey	1012.905
Basketball	0.996094	Sailing	984.733
chess	0.992188	Wrestling	802.307
Baseball	0.992188	Boxing	724.129
Golf	0.992188	Swimming	677.489

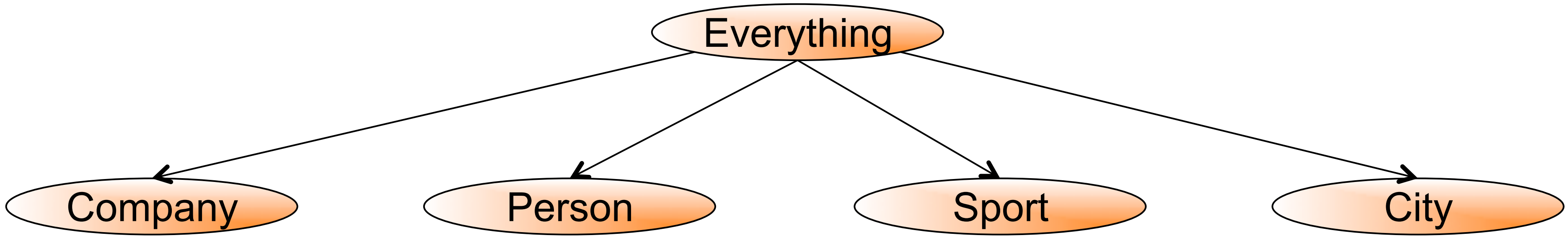
CBS using NELL's Ontology

Table 6. CPL probability and CBS score for extracted instances (after 5 iterations) for category Sport

CPL	probability	CBS	score
Game	0.998047	Baseball	1782.201
Show	0.998047	Basketball	1630.333
Football	0.998047	Soccer	1223.195
Day	0.998047	Skiing	1162.535
Drama	0.996094	Tennis	1022.093
Music	0.996094	Hockey	1012.905
Basketball	0.996094	Sailing	984.733
chess	0.992188	Wrestling	802.307
Baseball	0.992188	Boxing	724.129
Golf	0.992188	Swimming	677.489

CBS using NELL's Ontology

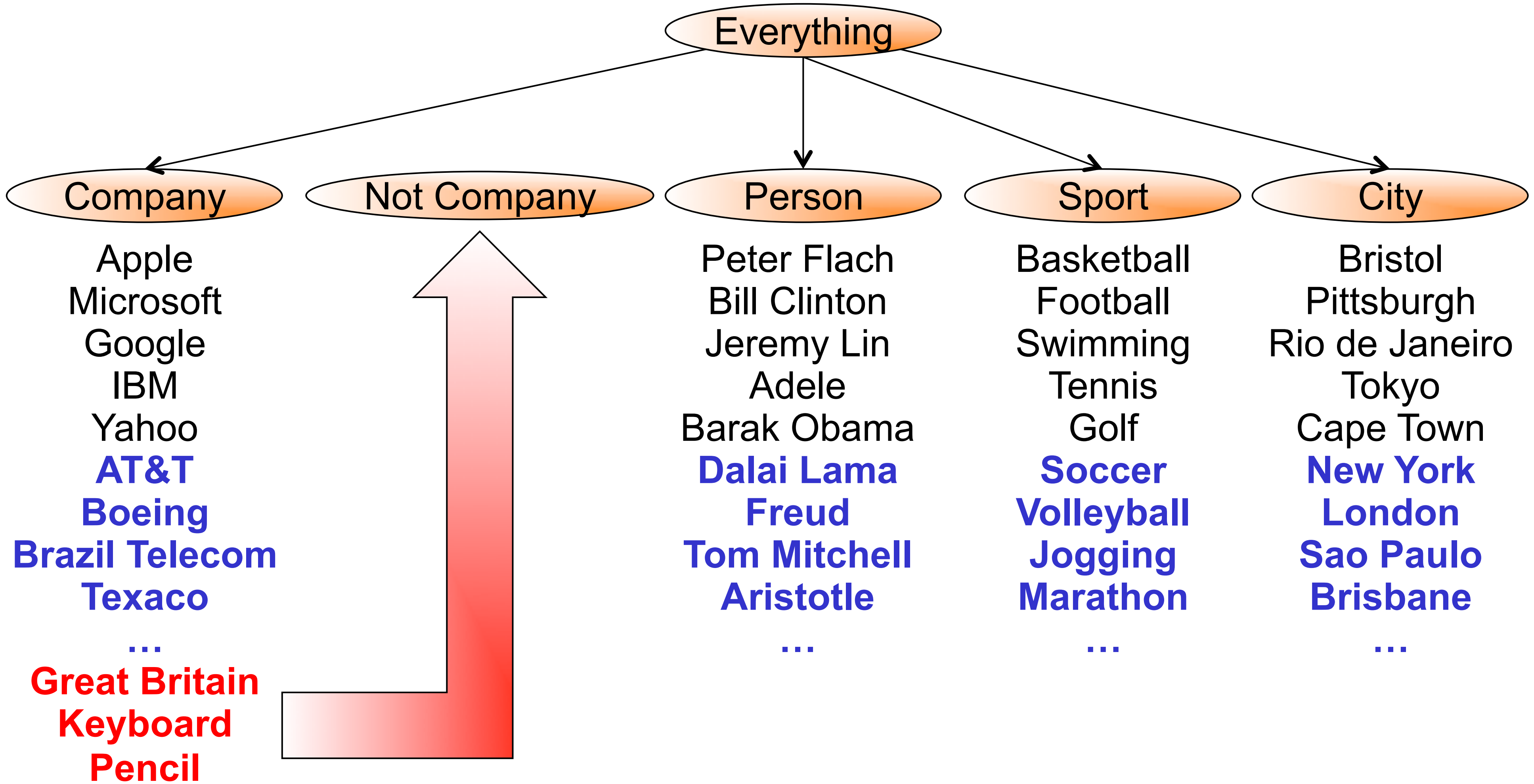
What if we do not have the mutual exclusiveness constraints?



Company	Person	Sport	City
Apple	Peter Flach	Basketball	Bristol
Microsoft	Bill Clinton	Football	Pittsburgh
Google	Jeremy Lin	Swimming	Rio de Janeiro
IBM	Adele	Tennis	Tokyo
Yahoo	Barak Obama	Golf	Cape Town
AT&T	**Dalai Lama**	**Soccer**	**New York**
Boeing	**Freud**	**Volleyball**	**London**
Brazil Telecom	**Tom Mitchell**	**Jogging**	**Sao Paulo**
Texaco	**Aristotle**	**Marathon**	**Brisbane**
…	**…**	**…**	**…**
Great Britain			
Keyboard			
Pencil			

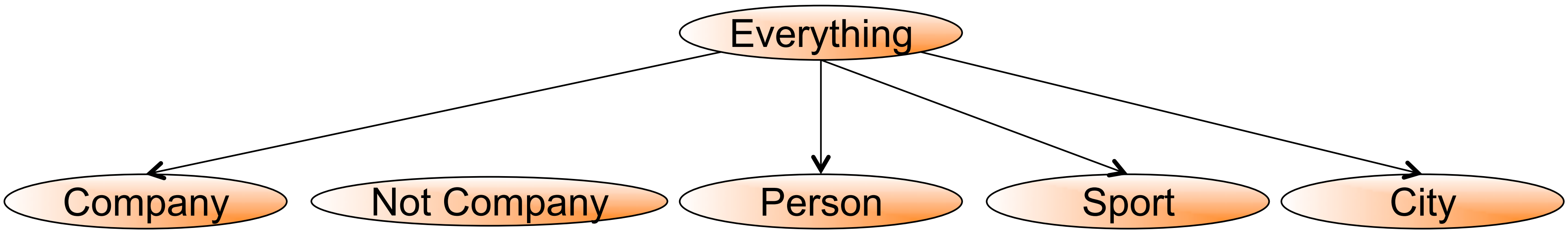
CBS using NELL's Ontology

What if we do not have the mutual exclusiveness constraints?



CBS using NELL’s Ontology

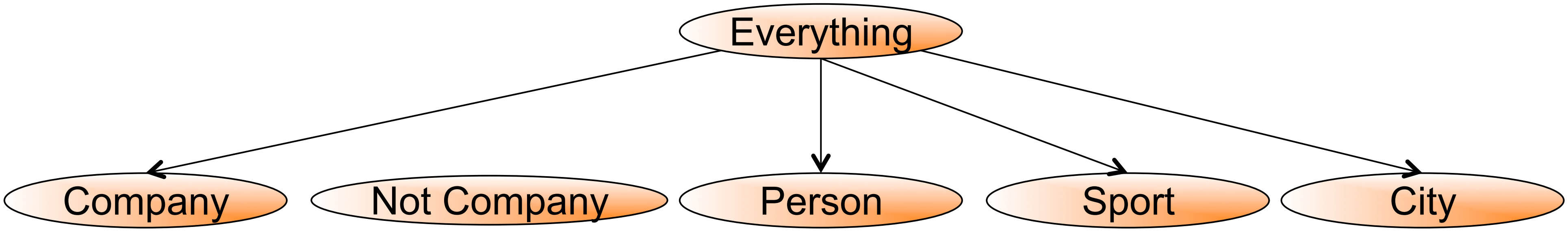
What if we do not have the mutual exclusiveness constraints?



Apple	**Great Britain**	Peter Flach	Basketball	Bristol
Microsoft	**Keyboard**	Bill Clinton	Football	Pittsburgh
Google	**Pencil**	Jeremy Lin	Swimming	Rio de Janeiro
IBM		Adele	Tennis	Tokyo
Yahoo		Barak Obama	Golf	Cape Town
AT&T		**Dalai Lama**	**Soccer**	**New York**
Boeing		**Freud**	**Volleyball**	**London**
Brazil Telecom		**Tom Mitchell**	**Jogging**	**Sao Paulo**
Texaco		**Aristotle**	**Marathon**	**Brisbane**
…		…	…	…
Great Britain				
Keyboard				
Pencil				

CBS using NELL's Ontology

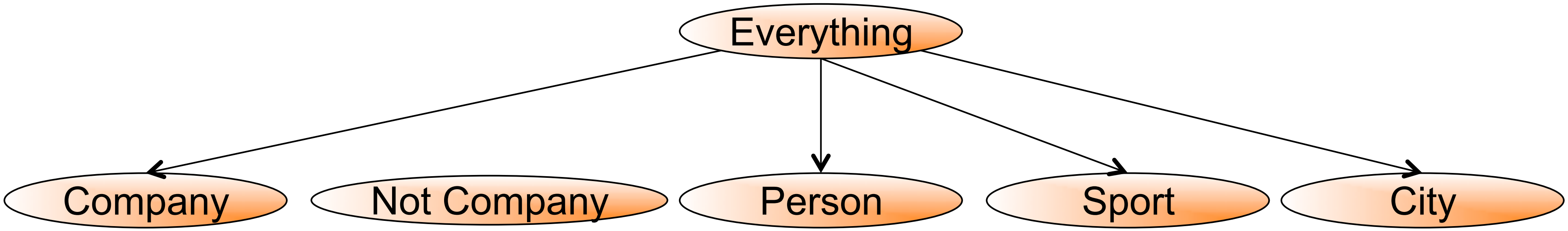
What if we do not have the mutual exclusiveness constraints?



Company	Not Company	Person	Sport	City
Apple	Great Britain	Peter Flach	Basketball	Bristol
Microsoft	Keyboard	Bill Clinton	Football	Pittsburgh
Google	Pencil	Jeremy Lin	Swimming	Rio de Janeiro
IBM		Adele	Tennis	Tokyo
Yahoo		Barak Obama	Golf	Cape Town
AT&T		**Dalai Lama**	**Soccer**	**New York**
Boeing		**Freud**	**Volleyball**	**London**
Brazil Telecom		**Tom Mitchell**	**Jogging**	**Sao Paulo**
Texaco		**Aristotle**	**Marathon**	**Brisbane**
…		**…**	**…**	**…**
Great Britain				
Keyboard				
Pencil				

CBS using NELL's Ontology

What if we do not have the mutual exclusiveness constraints?

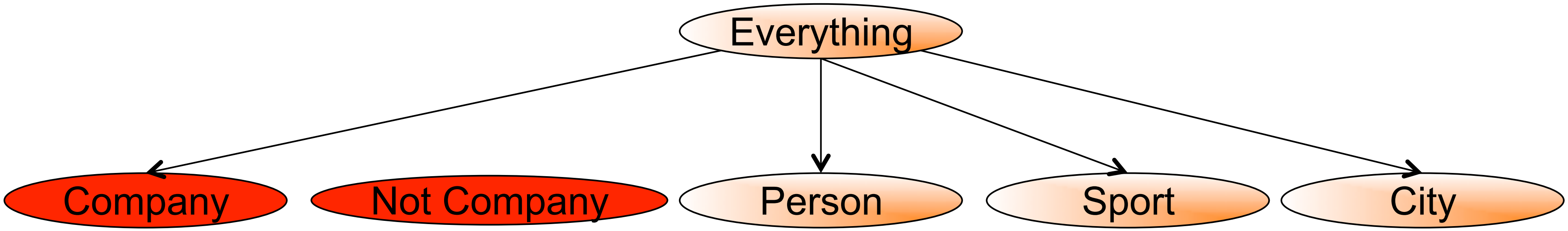


Apple	Great Britain	Peter Flach	Basketball	Bristol
Microsoft	Keyboard	Bill Clinton	Football	Pittsburgh
Google	Pencil	Jeremy Lin	Swimming	Rio de Janeiro
IBM		Adele	Tennis	Tokyo
Yahoo		Barak Obama	Golf	Cape Town
AT&T		**Dalai Lama**	**Soccer**	**New York**
Boeing		**Freud**	**Volleyball**	**London**
Brazil Telecom		**Tom Mitchell**	**Jogging**	**Sao Paulo**
Texaco		**Aristotle**	**Marathon**	**Brisbane**
…		**…**	**…**	**…**
Great Britain				
Keyboard				
Pencil				

MutuallyExclusive(Company,NotCompany);

CBS using NELL's Ontology

What if we do not have the mutual exclusiveness constraints?

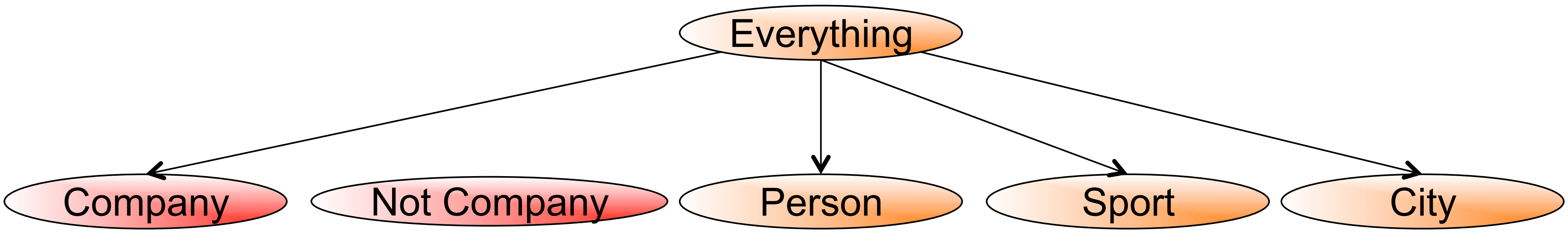


Company	Not Company	Person	Sport	City
Apple	Great Britain	Peter Flach	Basketball	Bristol
Microsoft	Keyboard	Bill Clinton	Football	Pittsburgh
Google	Pencil	Jeremy Lin	Swimming	Rio de Janeiro
IBM		Adele	Tennis	Tokyo
Yahoo		Barak Obama	Golf	Cape Town
AT&T		**Dalai Lama**	**Soccer**	**New York**
Boeing		**Freud**	**Volleyball**	**London**
Brazil Telecom		**Tom Mitchell**	**Jogging**	**Sao Paulo**
Texaco		**Aristotle**	**Marathon**	**Brisbane**
…		…	…	…
Great Britain				
Keyboard				
Pencil				

MutuallyExclusive(Company,NotCompany);

CBS using NELL's Ontology

What if we do not have the mutual exclusiveness constraints?

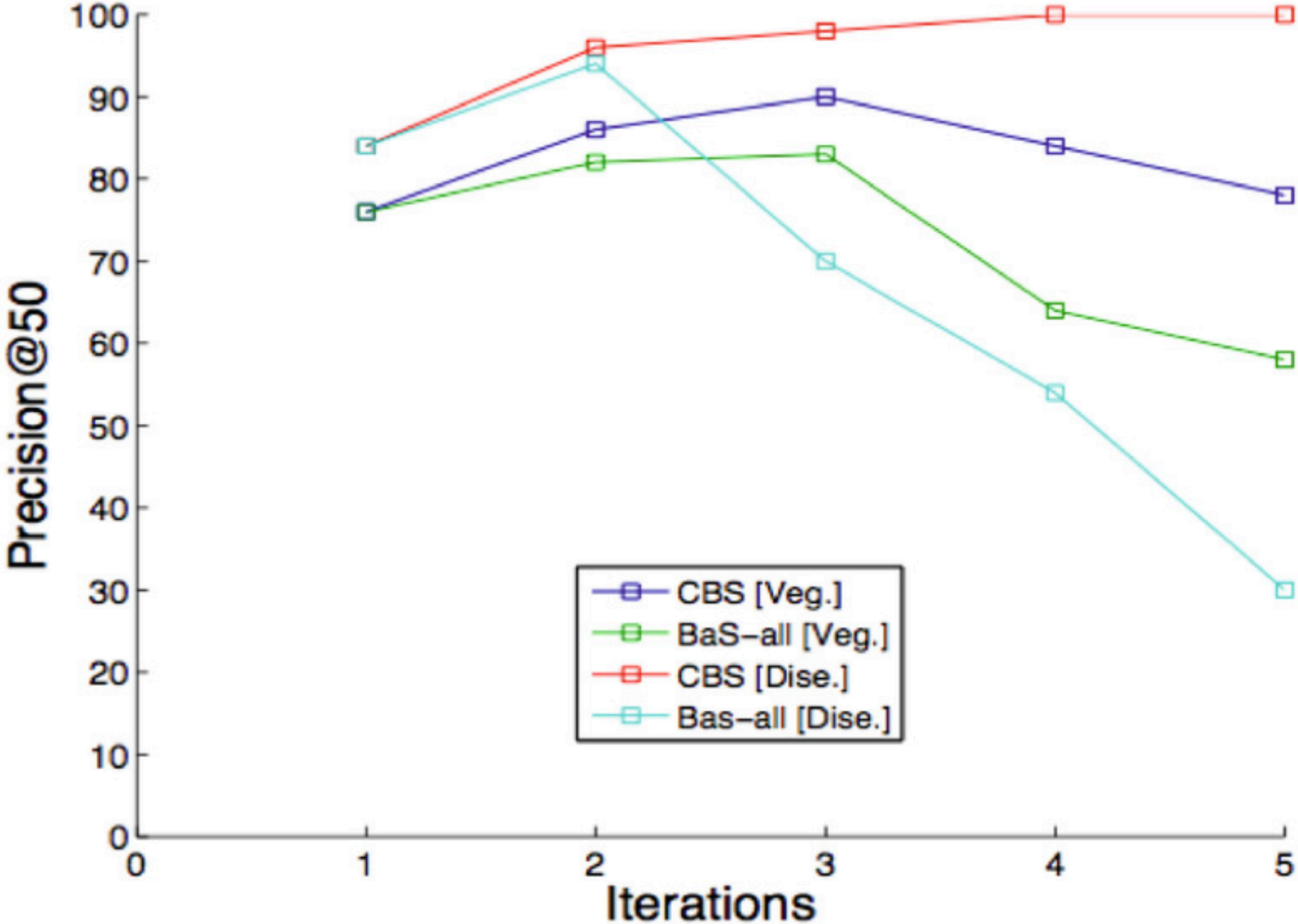


Apple	Great Britain	Peter Flach	Basketball	Bristol
Microsoft	Keyboard	Bill Clinton	Football	Pittsburgh
Google	Pencil	Jeremy Lin	Swimming	Rio de Janeiro
IBM		Adele	Tennis	Tokyo
Yahoo		Barak Obama	Golf	Cape Town
AT&T		**Dalai Lama**	**Soccer**	**New York**
Boeing		**Freud**	**Volleyball**	**London**
Brazil Telecom		**Tom Mitchell**	**Jogging**	**Sao Paulo**
Texaco		**Aristotle**	**Marathon**	**Brisbane**
…		**…**	**…**	**…**
Great Britain				
Keyboard				
Pencil				

MutuallyExclusive(Company,NotCompany);

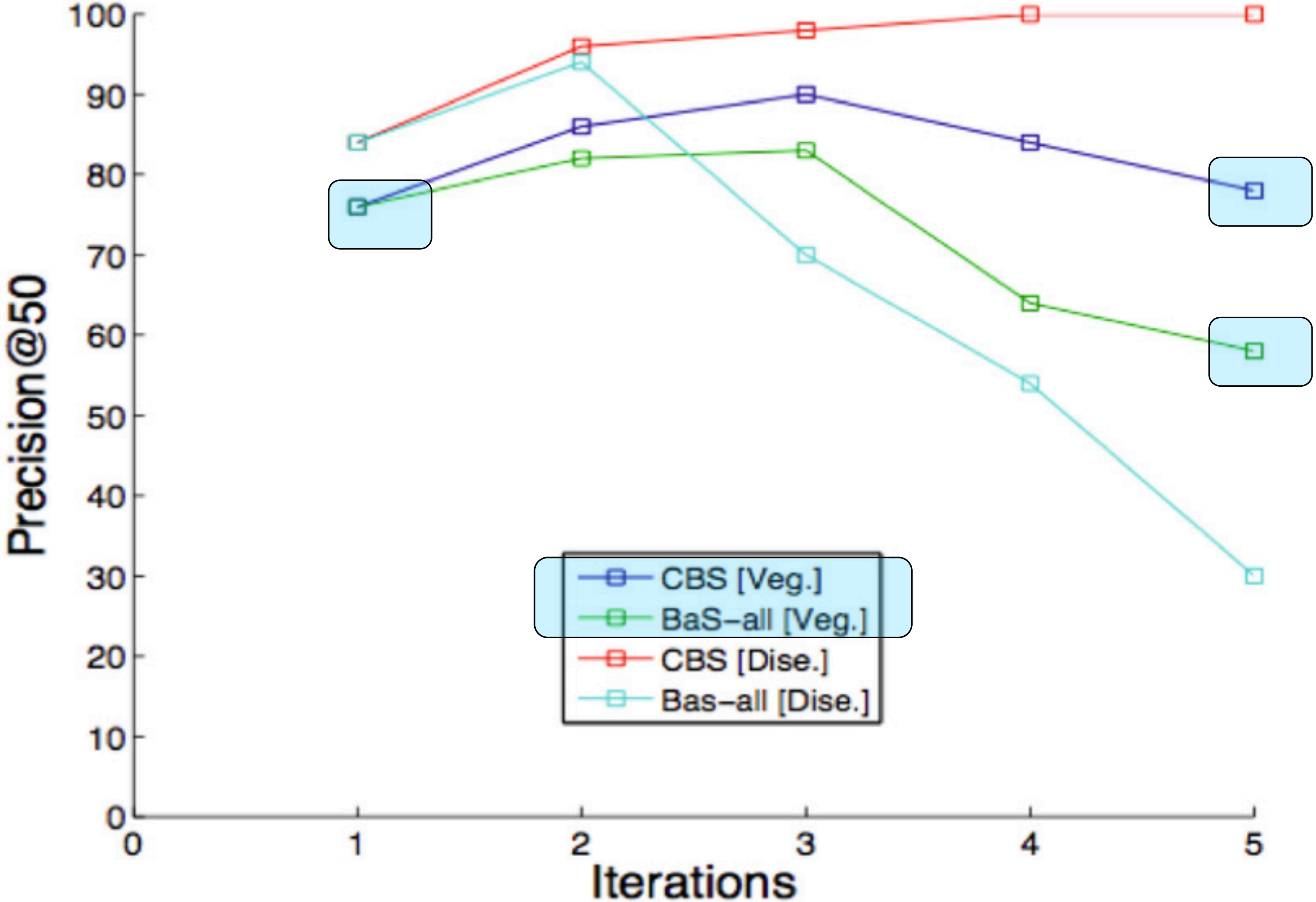
CBS using NELL's Ontology

What if we do not have the mutual exclusiveness constraints?



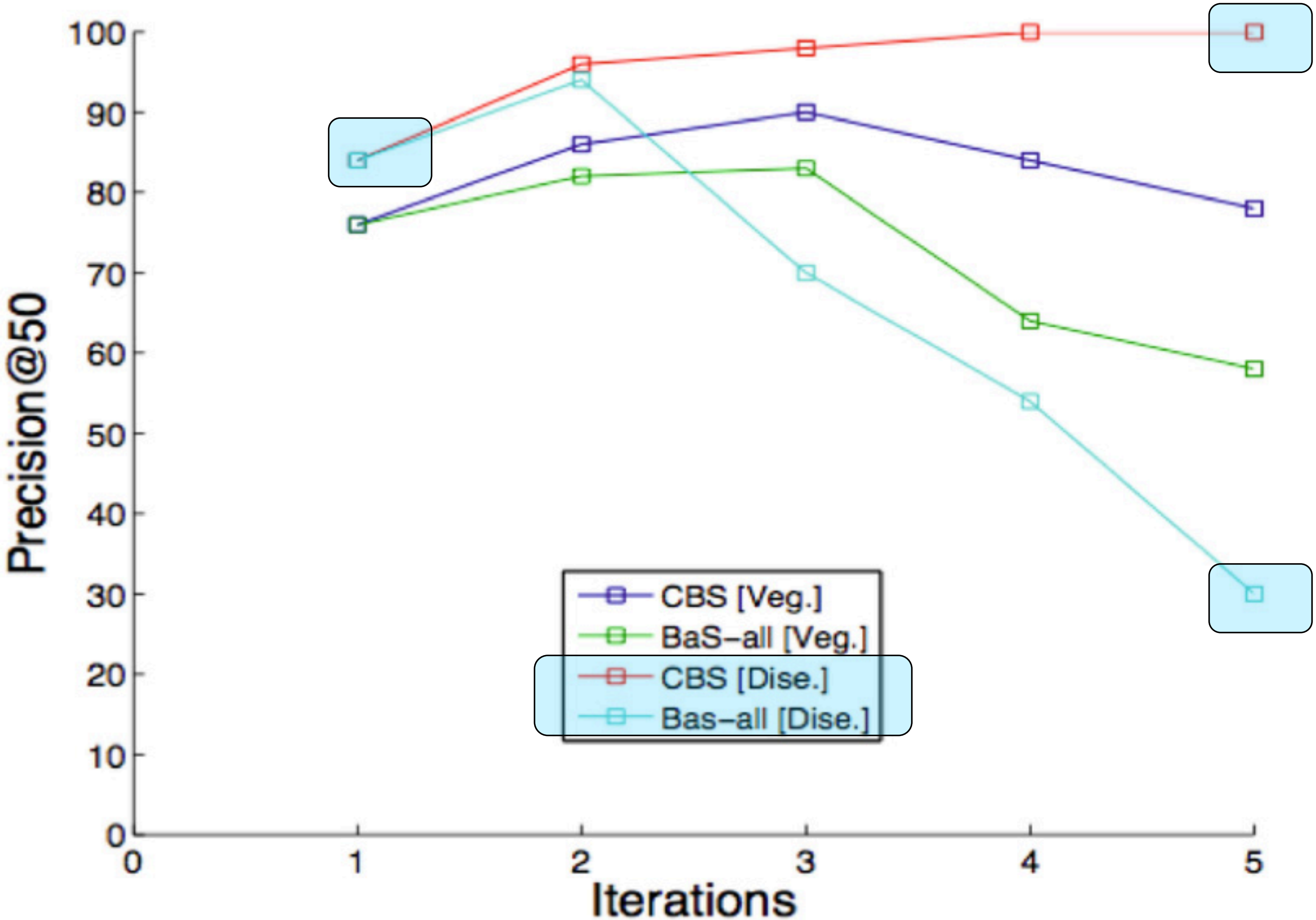
CBS using NELL's Ontology

What if we do not have the mutual exclusiveness constraints?



CBS using NELL's Ontology

What if we do not have the mutual exclusiveness constraints?



CBS using NELL's Ontology

What if we do not have the mutual exclusiveness constraints?

Precision@30

Categories	Automatic Neagtive (CBS)				Bas-all(set expansionalgorithm)		
	Iteration1	Iteration3	Iteration5		Iteration1	Iteration3	Iteration5
Companies	86%	92%	92%		86%	78%	78%
Diseases	82%	94%	100%		78%	92%	84%
KitchenItems	84%	92%	92%		92%	92%	92%
Persons	92%	100%	88%		82%	74%	64%
PhysicsTerms	76%	84%	88%		74%	84%	84%
Plants	80%	86%	90%		84%	74%	74%
Professions	78%	84%	94%		76%	82%	84%
SocioPolitics	64%	76%	82%		66%	46%	30%
Sports	98%	100%	100%		98%	100%	84%
Websites	86%	92%	92%		84%	88%	74%
Vegetables	74%	86%	78%		72%	78%	64%
Average	82%	90%	91%		81%	81%	74%

CBS using NELL's Ontology

What if we do not have the mutual exclusiveness constraints?

Precision@30

Categories	Automatic Neagtive (CBS)			Bas-all(set expansionalgorithm)		
	Iteration1	Iteration3	Iteration5	Iteration1	Iteration3	Iteration5
Companies	86%	92%	92%	86%	78%	78%
Diseases	82%	94%	100%	78%	92%	84%
KitchenItems	84%	92%	92%	92%	92%	92%
Persons	92%	100%	88%	82%	74%	64%
PhysicsTerms	76%	84%	88%	74%	84%	84%
Plants	80%	86%	90%	84%	74%	74%
Professions	78%	84%	94%	76%	82%	84%
SocioPolitics	64%	76%	82%	66%	46%	30%
Sports	98%	100%	100%	98%	100%	84%
Websites	86%	92%	92%	84%	88%	74%
Vegetables	74%	86%	78%	72%	78%	64%
Average	82%	90%	91%	81%	81%	74%

CBS using NELL's Ontology

What if we do not have the mutual exclusiveness constraints?

Precision@30

Categories	Automatic Neagtive (CBS)			Bas-all(set expansionalgorithm)		
	Iteration1	Iteration3	Iteration5	Iteration1	Iteration3	Iteration5
Companies	86%	92%	92%	86%	78%	78%
Diseases	82%	94%	100%	78%	92%	84%
KitchenItems	84%	92%	92%	92%	92%	92%
Persons	92%	100%	88%	82%	74%	64%
PhysicsTerms	76%	84%	88%	74%	84%	84%
Plants	80%	86%	90%	84%	74%	74%
Professions	78%	84%	94%	76%	82%	84%
SocioPolitics	64%	76%	82%	66%	46%	30%
Sports	98%	100%	100%	98%	100%	84%
Websites	86%	92%	92%	84%	88%	74%
Vegetables	74%	86%	78%	72%	78%	64%
Average	82%	90%	91%	81%	81%	74%

CBS using NELL's Ontology

What if we do not have the mutual exclusiveness constraints?

Precision@30

Categories	Automatic Neagtive (CBS)			Bas-all(set expansionalgorithm)		
	Iteration1	Iteration3	Iteration5	Iteration1	Iteration3	Iteration5
Companies	86%	92%	92%	86%	78%	78%
Diseases	82%	94%	100%	78%	92%	84%
KitchenItems	84%	92%	92%	92%	92%	92%
Persons	92%	100%	88%	82%	74%	64%
PhysicsTerms	76%	84%	88%	74%	84%	84%
Plants	80%	86%	90%	84%	74%	74%
Professions	78%	84%	94%	76%	82%	84%
SocioPolitics	64%	76%	82%	66%	46%	30%
Sports	98%	100%	100%	98%	100%	84%
Websites	86%	92%	92%	84%	88%	74%
Vegetables	74%	86%	78%	72%	78%	64%
Average	82%	90%	91%	81%	81%	74%

CBS using NELL's Ontology

What about Semantic Relations?

	Precision@20					
	CBS			BS		
Relations	Iteration1	Iteration3	Iteration5	Iteration1	Iteration3	Iteration5
Cities&countries	80%	88%	82%	76%	48%	38%
Countries&languages	82%	76%	76%	78%	74%	64%
Sports&Persons	92%	100%	100%	88%	84%	84%
University&state	84%	76%	74%	84%	74%	68%
Company&website	94%	100%	86%	88%	84%	72%
Average	86%	88%	84%	83%	73%	65%

CBS using NELL's Ontology

What about Semantic Relations?

	Precision@20					
	CBS			BS		
Relations	Iteration1	Iteration3	Iteration5	Iteration1	Iteration3	Iteration5
Cities&countries	80%	88%	82%	76%	48%	38%
Countries&languages	82%	76%	76%	78%	74%	64%
Sports&Persons	92%	100%	100%	88%	84%	84%
University&state	84%	76%	74%	84%	74%	68%
Company&website	94%	100%	86%	88%	84%	72%
Average	86%	88%	84%	83%	73%	65%

CBS using NELL's Ontology

What about Semantic Relations?

	Precision@20					
	CBS			BS		
Relations	Iteration1	Iteration3	Iteration5	Iteration1	Iteration3	Iteration5
Cities&countries	80%	88%	82%	76%	48%	38%
Countries&languages	82%	76%	76%	78%	74%	64%
Sports&Persons	92%	100%	100%	88%	84%	84%
University&state	84%	76%	74%	84%	74%	68%
Company&website	94%	100%	86%	88%	84%	72%
Average	86%	88%	84%	83%	73%	65%

Conclusions

Coupled Bayesian Sets

- semi-supervised learning approach to extract category instances (e.g. country(USA), city(New York) from web pages;
- based on the original Bayesian Sets
- can outperform algorithms such as the original Bayesian Set, the Naive Bayes classifier, the Bas-all and the coupled semi-supervised logistic regression algorithm (CPL);
- can be used to automatically generate new constraints to the set expansion task even when no mutually exclusiveness relationship is previously defined

Acknowledgements

Thanks to:

ECML/PKDD2012 audience! ☺

Also Thanks to:

- Department of Science and Technology, Government of India under Indo-Brazil cooperation programme;
- Brazilian research agencies CAPES and CNPq;
- Zoubin Ghahramani and K.A. Heller;
- All the Read The Web group;

contact: estevam.hruschka@gmail.com

http://rtw.ml.cmu.edu