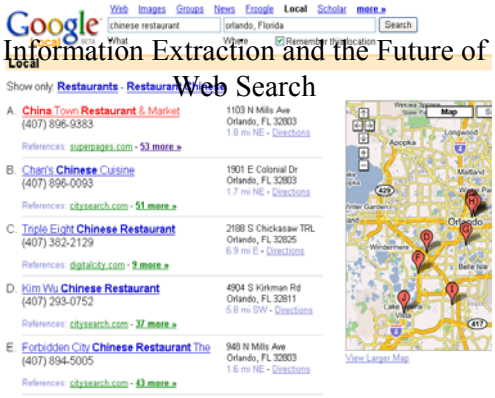
A Probabilistic Model of Redundancy in Information Extraction

Doug Downey, Oren Etzioni, Stephen Soderland

University of Washington
Department of Computer Science and Engineering
http://www.cs.washington.edu/research/knowitall

Information Extraction and the Future of Web Search



Motivation for Web IE

- What universities have active biotech research and in what departments?
- What percentage of the reviews of the Thinkpad T-40 are positive?

The answer is not on any single Web page!

Review: Unsupervised Web IE

Goal: Extract information on any subject automatically.

Review: Extraction Patterns

Generic extraction patterns (Hearst '92):

- "…**Cities** such as **Boston**, **Los Angeles**, and **Seattle**"
  ("**C** such as **NP1**, **NP2**, and **NP3**") =>
  IS-A(each(head(**NP**)), **C**), …
- "Detailed information for several **countries** such as **maps**, …" **ProperNoun(head(NP))**
- "I listen to pretty much all music but prefer **country** such as **Garth Brooks**"

Binary Extraction Patterns

R(I_1, I_2) ← I_1, **R** of I_2

Instantiated Pattern:

Ceo(Person, Company) ← <person> , **CEO** of <company>

"…Jeff Bezos, CEO of Amazon…"

"..Matt Damon, **star** of The Bourne Supremacy.."

"Erik Jonsson, CEO of Texas Instruments, mayor of Dallas from 1964-1971, and…"

Review: Unsupervised Web IE

Goal: Extract information on any subject automatically.

→*Generic extraction patterns*

Generic patterns can make mistakes.

→***Redundancy.***

7

Redundancy in Information Extraction

In large corpora, the same fact is often asserted multiple times:

"…and the rolling hills surrounding Sun Belt **cities such as** **Atlanta**"

"**Atlanta** **is a city** with a large number of museums, theatres…"

"…has offices in several major metropolitan **cities including** **Atlanta**"

Given a term *x* and a set of sentences about a class *C*, what is the probability that $x \in C$?

8

Redundancy – Two Intuitions

1) Repetition
2) Multiple extraction mechanisms

Phrase	Hits
"**Atlanta** and other cities"	980
"**Canada** and other cities"	286
"cities such as **Atlanta**"	5860
"cities such as **Canada**"	7

Goal: A formal model of these intuitions.

9

Outline

1. Modeling redundancy – the problem
2. URNS model
3. Parameter estimation for URNS
4. Experimental results
5. Summary

10

1. Modeling Redundancy – The Problem

Consider a single extraction pattern:

"*C* such as *x*"

Given a term *x* and a set of sentences about a class *C*, what is the probability that $x \in C$?

11

1. Modeling Redundancy – The Problem

Consider a single extraction pattern:

"*C* such as *x*"

If an extraction *x* appears *k* times in a set of *n* sentences containing this pattern, what is the probability that $x \in C$?

12

Modeling with *k*

`Country(x)`

extractions, $n = 10$

"…countries such as Saudi Arabia…"
"…countries such as the United States…"
"…countries such as Saudi Arabia…"
"…countries such as Japan…"
"…countries such as Africa…"
"…countries such as Japan…"
"…countries such as the United Kingdom…"
"…countries such as Iraq…"
"…countries such as Afghanistan…"
"…countries such as Australia…"

13

Modeling with *k*

`Country(x)`

extractions, $n = 10$

	k	$P_{noisy-or}$
Saudi Arabia	2	0.99
Japan	2	0.99
United States	1	0.9
Africa	1	0.9
United Kingdom	1	0.9
Iraq	1	0.9
Afghanistan	1	0.9
Australia	1	0.9

Noisy-Or Model :

$$P_{noisy-or}\left(x \in C \,|\, x \text{ appears } k \text{ times}\right) = 1-(1-p)^k$$

p is the probability that a single sentence is true.

$p = 0.9$

Important:

–Sample size (n)

–Distribution of C

} Noisy-or ignores these

14

Needed in Model: Sample Size

`Country(x)`

extractions, $n = 10$

	k	$P_{noisy-or}$
Saudi Arabia	2	0.99
Japan	2	0.99
United States	1	0.9
Africa	1	0.9
United Kingdom	1	0.9
Iraq	1	0.9
Afghanistan	1	0.9
Australia	1	0.9

`Country(x)`

extractions, n ~50,000

	k	$P_{noisy-or}$
Japan	1723	0.9999…
Norway	295	0.9999…
Israil	1	0.9
OilWatch Africa	1	0.9
Religion Paraguay	1	0.9
Chicken Mole	1	0.9
Republics of Kenya	1	0.9
Atlantic Ocean	1	0.9
New Zeland	1	0.9

As sample size increases, noisy-or becomes inaccurate.

15

Needed in Model: Distribution of *C*

`Country(x)`

extractions, n ~50,000

	k	$P_{noisy-or}$
Japan	1723	0.9999…
Norway	295	0.9999…
Israil	1	0.9
OilWatch Africa	1	0.9
Religion Paraguay	1	0.9
Chicken Mole	1	0.9
Republics of Kenya	1	0.9
Atlantic Ocean	1	0.9
New Zeland	1	0.9

$$P_{freq}\left(x \in C \,|\, x \text{ appears } k \text{ times}\right) = 1-(1-p)^{1000\,k/n}$$

16

Needed in Model: Distribution of *C*

`Country(x)`

extractions, n ~50,000

	k	P_{freq}
Japan	1723	0.9999…
Norway	295	0.9999…
Israil	1	0.05
OilWatch Africa	1	0.05
Religion Paraguay	1	0.05
Chicken Mole	1	0.05
Republics of Kenya	1	0.05
Atlantic Ocean	1	0.05
New Zeland	1	0.05

$$P_{freq}\left(x \in C \,|\, x \text{ appears } k \text{ times}\right) = 1-(1-p)^{1000\,k/n}$$

17

Needed in Model: Distribution of *C*

`Country(x)`

extractions, n ~50,000

	k	P_{freq}
Japan	1723	0.9999…
Norway	295	0.9999…
Israil	1	0.05
OilWatch Africa	1	0.05
Religion Paraguay	1	0.05
Chicken Mole	1	0.05
Republics of Kenya	1	0.05
Atlantic Ocean	1	0.05
New Zeland	1	0.05

`City(x)`

extractions, n ~50,000

	k	P_{freq}
Toronto	274	0.9999…
Belgrade	81	0.98
Lacombe	1	0.05
Kent County	1	0.05
Nikki	1	0.05
Ragaz	1	0.05
Villegas	1	0.05
Cres	1	0.05
Northeastwards	1	0.05

Probability that $x \in C$ depends on the distribution of C.

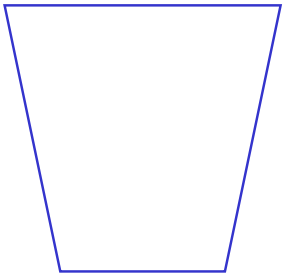
18

Outline

1. Modeling redundancy – the problem
2. **URNS model**
3. Parameter estimation for URNS
4. Experimental results
5. Summary

19

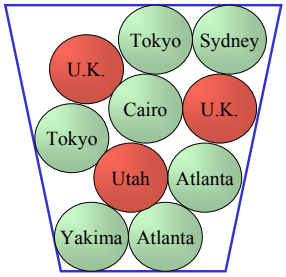
2. The URNS Model – Single Urn



20

2. The URNS Model – Single Urn

Urn for `City(x)`

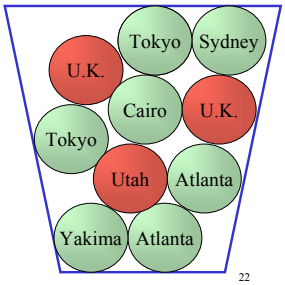


21

2. The URNS Model – Single Urn

Urn for `City(x)`

…`cities such as Tokyo`…



22

Single Urn – Formal Definition

C – set of unique target labels

E – set of unique error labels

num(b) – number of balls labeled by $b \in C \cup E$

num(B) –distribution giving the number of balls for each label $b \in B$.

23

Single Urn Example

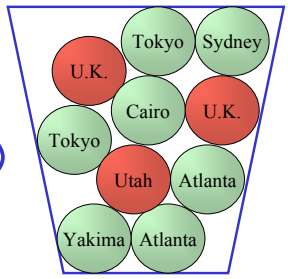
Urn for `City(x)`

num(`"Atlanta"`) = 2

num(*C*) = {2, 2, 1, 1, 1}

num(*E*) = {2, 1}

Estimated from data



24

Single Urn: Computing Probabilities

If an extraction x appears k times in a set of n sentences containing a pattern, what is the probability that $x \in C$?

25

Single Urn: Computing Probabilities

Given that an extraction x appears k times in n draws from the urn (with replacement), what is the probability that $x \in C$?

$$P(x \in C | x \text{ appears } k \text{ times in } n \text{ draws}) = \frac{\sum_{r \in num(C)} (\frac{r}{s})^k (1 - \frac{r}{s})^{n-k}}{\sum_{r' \in num(C \cup E)} (\frac{r'}{s})^k (1 - \frac{r'}{s})^{n-k}}$$

26

Uniform Special Case

Consider the case where $num(c_i) = R_C$ and $num(e_j) = R_E$ for all $c_i \in C$, $e_j \in E$

Then: $p = \frac{|C| R_C}{|E| R_E + |C| R_C}$

Then using a Poisson Approximation:

$$P(x \in C | x \text{ appears } k \text{ times in } n \text{ draws}) \approx \frac{1}{1 + \frac{|E|}{|C|} (\frac{p_E}{p_C})^k e^{n(p_C - p_E)}}$$

where $p_C = \frac{p}{|C|}$ and $p_E = \frac{1-p}{|E|}$. In practice, $p_C > p_E$.

Odds increase exponentially with k, but decrease exponentially with n.

27

The URNS Model – Multiple Urns

Correlation across extraction mechanisms is higher for elements of C than for elements of E.

28

Outline

1. Modeling redundancy – the problem
2. URNS model
3. **Parameter estimation for URNS**
4. Experimental results
5. Summary

29

3. Parameter Estimation for URNS

Simplifying Assumptions:

- Assume that *num*(C) and *num*(E) are Zipf distributed.
  - Frequency of ith most repeated label in $C \propto i^{-z_C}$
- Then *num*(C) and *num*(E) are characterized by five parameters:

$$z_C, z_E, |C|, |E|, p$$

30

Parameter Estimation

Supervised Learning
- Differential Evolution (maximizing conditional likelihood)

Unsupervised Learning
- Growing interest in IE without hand-tagged training data (e.g. DIPRE; Snowball; KnowItAll; Riloff and Jones 1999; Lin, Yangarber, and Grishman 2003)
- How to estimate *num*(*C*) and *num*(*E*)?

31

Unsupervised Parameter Estimation

Unsupervised Learning
- EM, with additional assumptions:
  - $|E| = 1{,}000{,}000$
  - $z_E = 1$
  - p is given ($p = 0.9$ for KnowItAll patterns)

32

EM Process

EM for Unsupervised IE:
- *E-Step:* Assign probabilities to extracted facts using Urns.
- *M-Step*:
  1. Estimate z_C by linear regression on log-log scale.
  2. Set $|C|$ equal to expected number of true labels extracted, plus *unseen* true labels (using Good-Turing estimation).

33

Outline

1. Modeling redundancy – the problem
2. Urns model
3. Parameter estimation for Urns
4. **Experimental results**
5. Summary

34

4. Experimental Results

Previous Approach: PMI (in KnowItAll, inspired by Turney, 2001)

$$PMI(\text{“<}\textbf{\textit{City}}\text{> }\textbf{\textit{hotels}}\text{”}, \text{“Tacoma”}) = \frac{Hits(\text{"Tacoma hotels"})}{Hits(\text{"Tacoma"})}$$

–Expensive: several hit-count queries per extraction

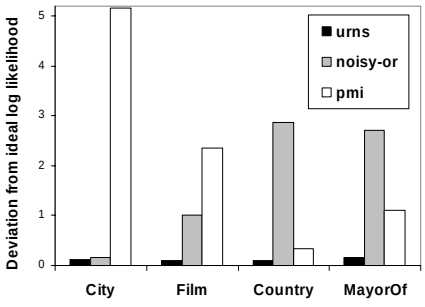
–Using Urns improves efficiency by ~8x

–‘Bootstrapped’ training data not representative

–Probabilities are polarized (Naïve Bayes)

35

Unsupervised Likelihood Performance



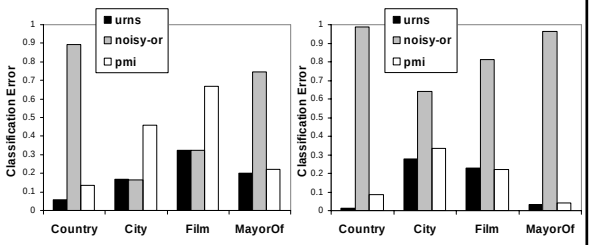
36

URNS Robust to Parameter Changes

Parameter	URNS improvement over: Noisy-or	PMI
$z_E = 1$, $\|E\| = 10^6$, $p = 0.9$	14x	19x
$z_E = 1.1$	15x	19x
$z_E = 0.9$	14x	18x
$\|E\| = 10^7$	13x	18x
$\|E\| = 10^5$	14x	18x
$p = 0.95$	9x	12x
$p = 0.80$	8x	11x

37

Classification Accuracy



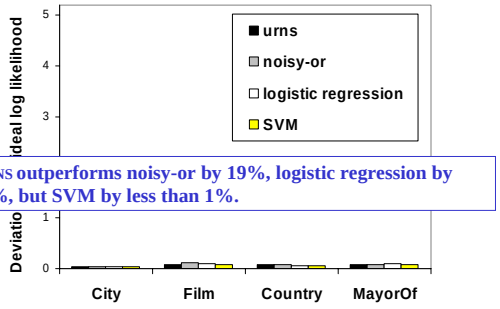
Loss Functions:

(False +) = (False –)

(False +) = 9(False –)

38

Supervised Results



URNS outperforms noisy-or by 19%, logistic regression by 10%, but SVM by less than 1%.

39

Modeling Redundancy – Summary

Given a term *x* and a set of sentences about a class *C*, what is the probability that $x \in C$?

40

Modeling Redundancy – Summary

URNS Model of Redundancy in Text Classification

Parameter learning algorithms

Substantially improved performance for Unsupervised IE

41

Pattern Learning

City =>

- **cities such as <*City*>**
- **<*City*> and other cities**
- **cities including <*City*>**
- **<*City*> is a city**, etc.

But what about:

- **<*City*> hotels**
- **headquartered in <*City*>**
- **the greater <*City*> area**, etc.

42

Pattern Learning (PL)

Seed Instances:
Moscow
Cleveland
London
Mexico City

Web Search Engine

Pattern Learning (PL)

Seed Instances:
Moscow
Cleveland
London
Mexico City

Web Search Engine

Context Strings: …near the city of Cleveland you can find the …

Pattern Learning (PL)

Seed Instances:
Moscow
Cleveland
London
Mexico City

Web Search Engine

Context Strings: …near the city of Cleveland you can find the …

Repeat as desired

The "best" patterns:
city of <City>

Large collection of context strings

A *pattern* is any substring of a context string that includes the seed.

Which patterns are "best"

Both **precision** and **recall** are important, but hard to measure.

Which patterns are "best"

Both **precision** and **recall** are important, but hard to measure.

$$EstimatedRecall = \frac{c}{S}$$

$$EstimatedPrecision = \frac{c+k}{c+n+m}$$

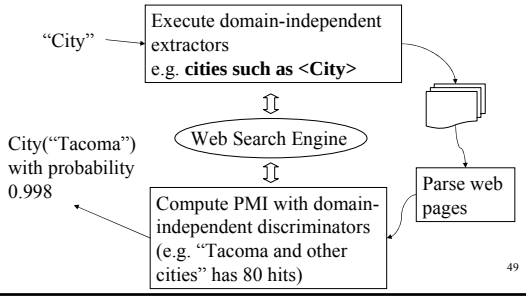
Where:

- The pattern is found for c target seeds and n non-target seeds.
- S is the total number of target seeds.
- k/m is a prior estimate of pattern precision.

Patterns as Extractors and Discriminators

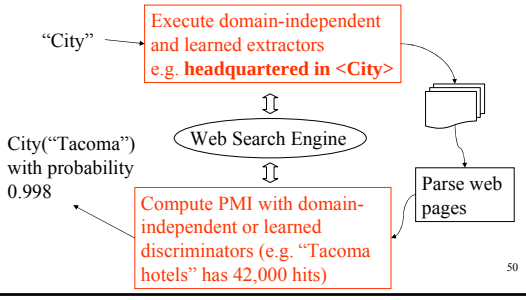
Pattern Learner (PL) → Patterns → Extractors (*increase coverage*) / Discriminators (*increase accuracy*)

KnowItAll



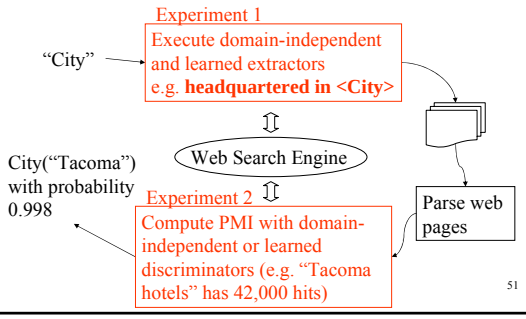
49

KnowItAll with Pattern Learning



50

KnowItAll with Pattern Learning



51

Experiment 1: Learned patterns as extractors

Baseline – KnowItAll with domain independent extractors.

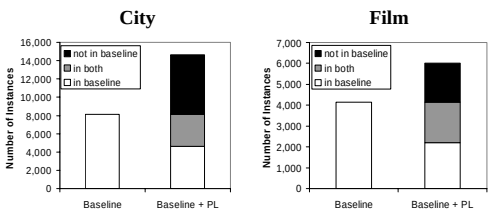
Baseline+PL – KnowItAll with both domain-independent and learned extractors.

In both cases, domain independent discriminators.

We compare **coverage** – i.e. the number of instances extracted at a fixed level of precision (0.90).

52

Experiment 1: Learned patterns as extractors



Adding PL improves coverage by **50%** to **80%**.

53

Experiment 1: Learned patterns as extractors

Pattern	Correct Extractions	Precision
the cities of <City>	5215	0.80
headquartered in <City>	4837	0.79
for the city of <City>	3138	0.79
in the movie <Film>	1841	0.61
<Film> the movie starring	957	0.64
movie review of <Film>	860	0.64

54

Experiment 2: Learned patterns as discriminators

Baseline – Uses domain independent discriminators.

Baseline+PL – Uses both domain independent and learned discriminators.

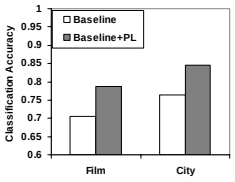
We compare the **classification accuracy** of the two methods (the fraction of extractions classified correctly as positive or negative) after running two discriminators on each of 300 extractions.

Experiment 2: Learned patterns as discriminators

Baseline – Uses domain independent discriminators.

Baseline+PL – Uses both domain independent and learned discriminators.

We compare the **classification accuracy** of the two methods (the fraction of extractions classified correctly as positive or negative) after running two discriminators on each of 300 extractions.



Adding PL reduces classification errors by **28%** to **35%**

Selecting discriminators

In Experiment 2, for each extraction we executed:

- a fixed pair of discriminators
- choosing those with the highest precision

This approach can be improved.

Selecting discriminators

The baseline ordering can be improved in several ways:

- Precision *and* recall are important for accuracy.

Selecting discriminators

The baseline ordering can be improved in several ways:

- Precision *and* recall are important for accuracy.
- Discriminators can perform better on some extractions than on others:
  - E.g. *rare* extractions:
    - A high-precision but rare discriminator might falsely return a PMI a zero (e.g. "cities such as Fort Calhoun" has 0 hits)
    - Using a more prevalent discriminator on rare facts could improve accuracy (e.g. "Fort Calhoun hotels" has 20 hits).

Selecting discriminators

The baseline ordering can be improved in several ways:

- Precision *and* recall are important for accuracy.
- Discriminators can perform better on some extractions than on others:
  - E.g. *rare* extractions:
    - A high-precision but rare discriminator might falsely return a PMI a zero (e.g. "cities such as Fort Calhoun" has 0 hits)
    - Using a more prevalent discriminator on rare facts could improve accuracy (e.g. "Fort Calhoun hotels" has 20 hits).
- The system should prioritize uncertain extractions.

The Discriminator Selection Problem

Goal: given a set of extractions and discriminators, find a **policy** that maximizes expected accuracy.

- Known as "active classification." Assume discriminators are conditionally independent (as in Guo, 2002).
  - The general optimization problem is NP-hard.
  - The *MU Heuristic* is optimal in important special cases and improves performance in practice.

61

The *MU Heuristic*

Greedily choose the action with maximal marginal utility *MU*:

$$MU = \frac{\text{Expected increase in accuracy}}{\text{cost of action}}$$

We can compute *MU* given

- the discriminator's precision and recall (adjusted according to the extraction's hit count)
- the system's current belief in the extraction.

(similar to Etzioni 1991).

62

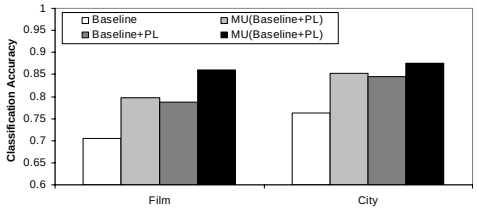
Experiment 3: Testing the *MU* Heuristic

As in experiment 2, the Baseline and Baseline+PL configurations execute two discriminators (ordered by precision) on each of 300 extractions.

The *MU* configurations are constrained to execute the same total number of discriminators (600), but can dynamically choose to execute the discriminator and extraction with highest marginal utility.

63

Experiment 3: Testing the *MU* Heuristic



Ordering by *MU* further reduces classification errors by **19%** to **35%**, for a total error reduction of **47%** to **53%**.

64

Summary

Pattern Learning

- Increased coverage by **50%** to **80%**.
- Decreased errors **28%** to **35%**.

Theoretical Model

- decreased errors an additional **19%** to **35%**.

65

Extensions to PL

Complex patterns

- Syntax (Snow and Ng 2004), Classifiers (Snowball)
- Tend to require good training data

Iteration (Patterns->Seeds->Patterns->….)

- (Brin 1998, Agichtein and Gravano 2000, Riloff 1999)
- Scope creep…URNS?

66

Backup

Future Work

Additional Experiments

Normalization/Negative Evidence

- Don't mistake cities for countries, etc (e.g. Lin et al 2003, Thelen & Riloff 2002)

Learning extraction patterns

- E.g. DIPRE, Snowball

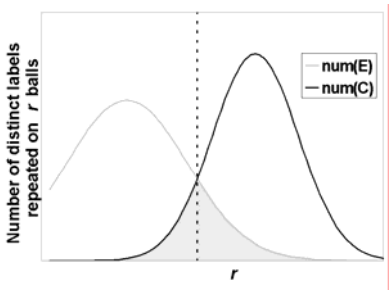
Other applications

- E.g. PMI applied to synonymy (Turney, 2001)

URNS adjusts for sample size and distribution of *C* and *E*

k	*p*	*n*	R_C/R_E	$P_{noisy\text{-}or}$	P_{urns}
3	0.9	10,000	90	0.999	0.999
3	0.9	10,000	**9**	**0.999**	**0.930**
3	0.9	**20,000**	9	**0.999**	**0.196**

When is URNS effective?



URNS works when the **confusion region** is small.

The URNS Model – Multiple Urns

Phrase	Hits	
"Atlanta and other cities"	980	6x
"cities such as Atlanta"	5860	
"Canada and other cities"	286	0.02x
"cities such as Canada"	7	
"Texas and other cities"	4710	0.002x
"cities such as Texas"	9	

Correlation between counts for different extractors is informative.

Multi-urn Assumptions

Modeling the Urns:

- z_C, z_E, $|C|$, $|E|$ the same for all urns.
- Different extraction precisions *p*.

Modeling correlation between Urns:

- Relative frequencies are perfectly correlated for elements of *C*, and *some* elements of *E*.
- The remaining elements of *E* appear for only one kind of extraction mechanisms.

Multi-urn Assumptions

$$P(x \in C \mid x \text{ appears } (k_1,\ldots,k_{|M|}) \text{ times in } (n_1,\ldots,n_{|M|}) \text{ draws}) = \frac{\sum_{c_i \in C} \prod_{m \in M} P(A_m(c_i, k_m, n_m))}{\sum_{x \in C \cup E} \prod_{m \in M} P(A_m(x, k_m, n_m))}$$

$A_m(x, k, m)$ = Event that extraction x is seen k times in urn m.

With our assumptions, we can obtain the above expression in closed form.

73

Recall – Distribution of *C*

Country(x) extractions, n ~50,000	k	$P_{noisy-or}$
Japan	1723	0.9999…
Norway	295	0.9999…
Israil	1	0.05
OilWatch Africa	1	0.05
Religion Paraguay	1	0.05
Chicken Mole	1	0.05
Republics of Kenya	1	0.05
Atlantic Ocean	1	0.05
New Zeland	1	0.05

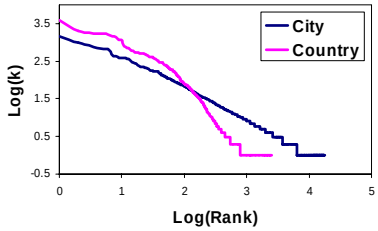
City(x) extractions, n ~50,000	k	P_{freq}
Toronto	274	0.9999…
Belgrade	81	0.98
Lacombe	1	0.05
Kent County	1	0.05
Nikki	1	0.05
Ragaz	1	0.05
Villegas	1	0.05
Cres	1	0.05
Northeastwards	1	0.05

Probability that $x \in C$ depends on the distribution of C.

74

Untagged Data

A mixture of samples from *num*(*C*) and *num*(*E*):



Challenge: Estimate *num*(*C*), *num*(*E*).

75

Related Work

Redundancy in IE

- Heuristics/noisy-or models (e.g. Riloff & Jones 1999; Brin 1998; Agichtien & Gravano 2000; Lin *et al*. 2003)
- Supervised models (Skounakis & Craven, 2003)
- Do not model *n*, *num*(*C*), *num*(*E*)

BLOG models (Milch *et al*. 2004)

- Our focus is on IE/Text Classification; we give algorithms, experimental results

76

Related Work

CRFs for confidence estimation (Culotta & McCallum, 2004)

- Our interest is *combining* evidence from multiple extractions.

77

Supervised Results

	City	*Film*	*Mayor*	*Country*	*Average*
noisy-or	0.0439	0.1256	0.0857	0.0795	0.0837
logistic regression	0.0466	0.0893	**0.0655**	0.1020	0.0759
SVM	0.0444	0.0865	0.0659	**0.0769**	0.0684
URNS	**0.0418**	**0.0764**	0.0721	0.0823	**0.0681**

Deviation from the ideal log-likelihood.

78