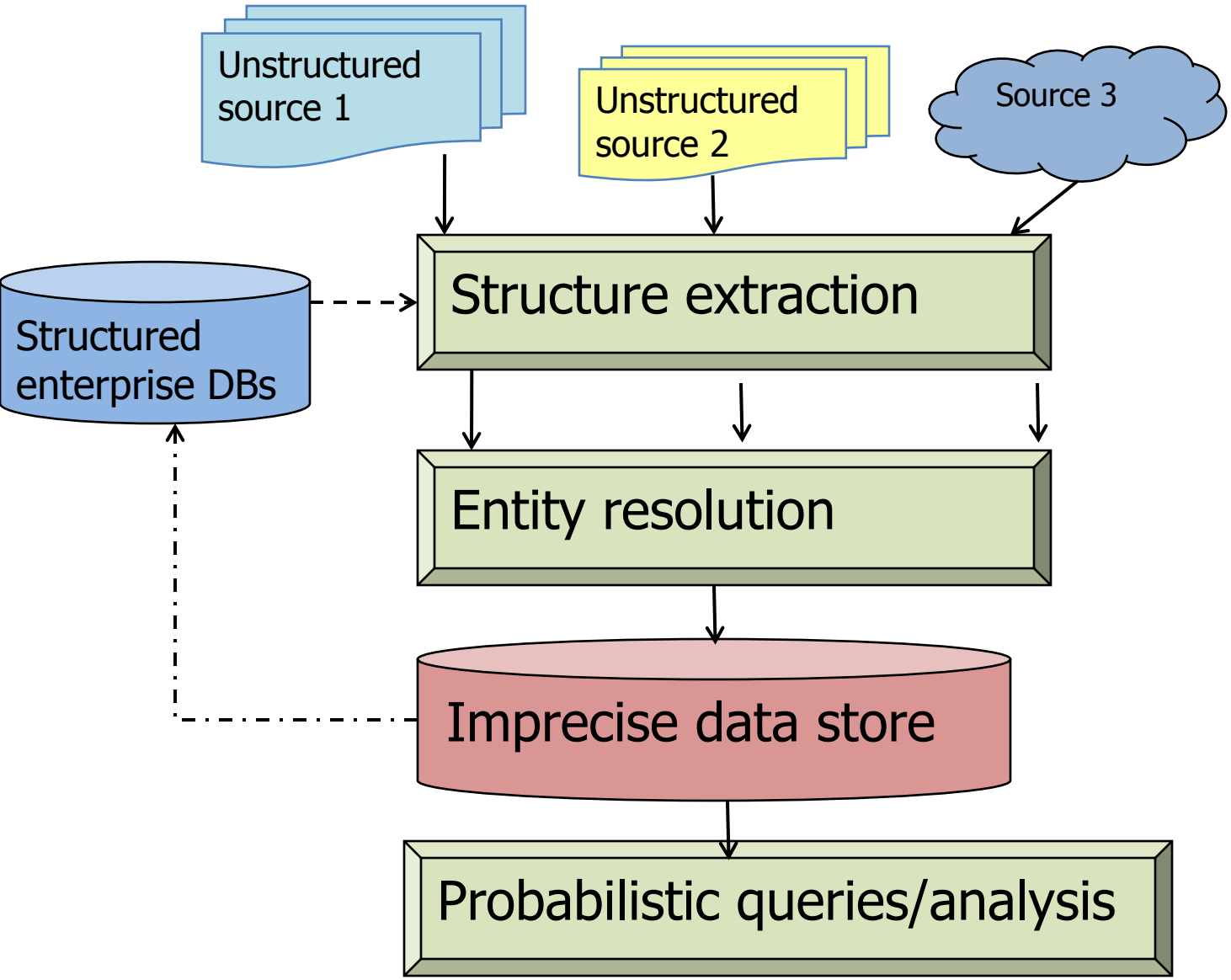
Queries over unstructured enterprise data: probabilistic methods to the rescue

Sunita Sarawagi
IIT Bombay
http://www.cse.iitb.ac.in/~sunita

Unstructured data in Enterprises

- Customer emails
  - Link customer emails to a sales transaction
- Phone conversation transcripts
  - Customer moods & satisfaction, products mentioned
- Reviews and blogs
  - Relate product name, attribute names, opinion to sales
- Claim forms
  - Repair records from insurance claim forms
- Names and addresses
  - Relate customer records across many databases
  - Merchant records from sales invoices for expense reimbursement

Making sense of unstructured data



Structure Extraction

IE from free format text

According to Robert Callahan, president of Eastern's flight attendants union, the past practice of Eastern's parent, Houston-based Texas Air Corp., has involved ultimatums to unions to accept the carrier's terms

- Extract person, location, organization names

Segmenting text records

Useful for data warehousing, data cleaning, web data integration

Address

House number	Building	Road	City	State	Zip
4089	Whispering Pines	Nobel Drive	San Diego	CA	92122

Citation

Ronald Fagin, Combining Fuzzy Information from Multiple Systems, Proc. of ACM SIGMOD, 2002

Segment(s_i)	Sequence	Label(s_i)
S_1	Ronald Fagin	Author
S_2	Combining Fuzzy Information from Multiple Systems	Title
S_3	Proc. of ACM SIGMOD	Conference
S_4	2002	Year

Table queries over lists on the web

Structured Query	
Cornell University	Ithaca
State University of New York	Stony Brook
New York University	New York

- New York University (NYU), New York City, founded in 1831.
- Columbia University, founded in 1754 as King's College.
- Binghamton University, Binghamton, established in 1946.
- State University of New York, Stony Brook, New York, founded in 1957
- Syracuse University, Syracuse, New York, established in 1870
- State University of New York, Buffalo, established in 1846
- Rensselaer Polytechnic Institute (RPI) at Troy.

Extraction

Structured Query	
Cornell University	Ithaca
State University of New York	Stony Brook
New York University	New York

- New York University (NYU), New York City, founded in 1831.
- Columbia University, founded in 1754 as King's College.
- Binghamton University, Binghamton, established in 1946.
- State University of New York, Stony Brook, New York, founded in 1957
- Syracuse University, Syracuse, New York, established in 1870
- State University of New York, Buffalo, established in 1846
- Rensselaer Polytechnic Institute (RPI) at Troy.

Gupta and Sarawagi, Answering Table Augmentation Queries using Unstructured Lists on the Web,VLDB 2009

Available clues..

- Surface patterns, regular expression:
  - Pattern: X. [X.] Xx*→ People name
  - Pattern: dddd → Year
- Commonly occurring words:
  - Co. Ltd → company name
- Ordering of words:
  - Text after “Mr.” is person name
  - Text after comma is location
- Order of attributes:
  - City names before state names
- Match with existing entities
  - City names in a database or Ontology

Putting the clues together

- Manually-developed set of rules
  - Makes hard decisions on a subset of clues
  - Tedious, lots and lots of special cases
  - Ad hoc ways of combining varied set of clues
- Statistical learning-based approach (lots!)
  - Generative: HMMs (1990s)
    - Intuitive but not too flexible
  - Conditional: CRFs (2000s)
    - Flexible feature set.

Extraction as sequence segmentation

R. Fagin and J. Helpbern. Belief Awareness Reasoning

t	1	2	3	4	5	6	7	8
x	R.	Fagin	and	J.	Helpbern	Belief	Awareness	Reasoning

l,u	l_1=1, u_1=2		l_1=u_1=3	l_1=4, u_1=5		l_1=6, u_1=8		
x	R.	Fagin	and	J.	Helpbern	Belief	Awareness	Reasoning
y	**Author**		**Other**	**Author**		**Title**		

Similarity to author's column in database

$$\mathbf{f}(y_j, \mathbf{x}, l_j, u_j, y_{j-1})$$

Features describe the **segment** from l_j to u_j

Features

- Feature vector for each *segment* $S_j=(l_j,u_j)$

$$\mathbf{f}(y_j, \mathbf{x}, l_j, u_j, y_{j-1})$$

User provided

j-th label

Start of S_j

end of S_j

previous label

- Examples: x: R. Fagin and J. Helpbern. Belief Awareness

$$f_2(y_j, \mathbf{x}, 1, 2, y_{j-1}) = 2 \quad \text{(segment length)}$$

$$f_3(y_j, \mathbf{x}, 1, 2, y_{j-1}) = 1 \text{ if } y_j \text{ is Person \& } y_{j-1} \text{ is Start}$$

$$f_5(P, \mathbf{x}, 1, 2, y_{j-1}) = 1 \text{ if} (x_1 x_2) = \text{X.Xx}_+$$

$$f_4(P, \mathbf{x}, 1, 2, y_{j-1}) = \max_{e \in D} \text{cosine}(e, \text{"R.Fagin"})$$

***D* is a dictionary of known entity names**

Parameters: weight of features

$$\mathbf{W} = W_1 W_2 \ldots W_{|\mathbf{f}|}$$

Learned from labeled examples.

Probability of a segmentation

Given a word sequence $\mathbf{x} : x_1 \ldots x_n$,
segmentation $\mathbf{S} : S_1, \ldots, S_k$ where $S_j = (y_j, l_j, u_j)$

$$\text{Score}(\mathbf{x}, \mathbf{S}) = \sum_{j=1}^{k} \mathbf{W}.\mathbf{f}(y_j, \mathbf{x}, l_j, u_j, y_{j-1})$$

$$\Pr(\mathbf{S}|\mathbf{x}) = \frac{\exp(\text{Score}(\mathbf{x}, \mathbf{S}))}{\sum_{\mathbf{S}'} \exp(\text{Score}(\mathbf{x}, \mathbf{S}'))}$$

Deploying

Find $\mathbf{S} : S_1, S_2 \ldots, S_k$ given $\mathbf{x} : x_1 \ldots x_n$ as

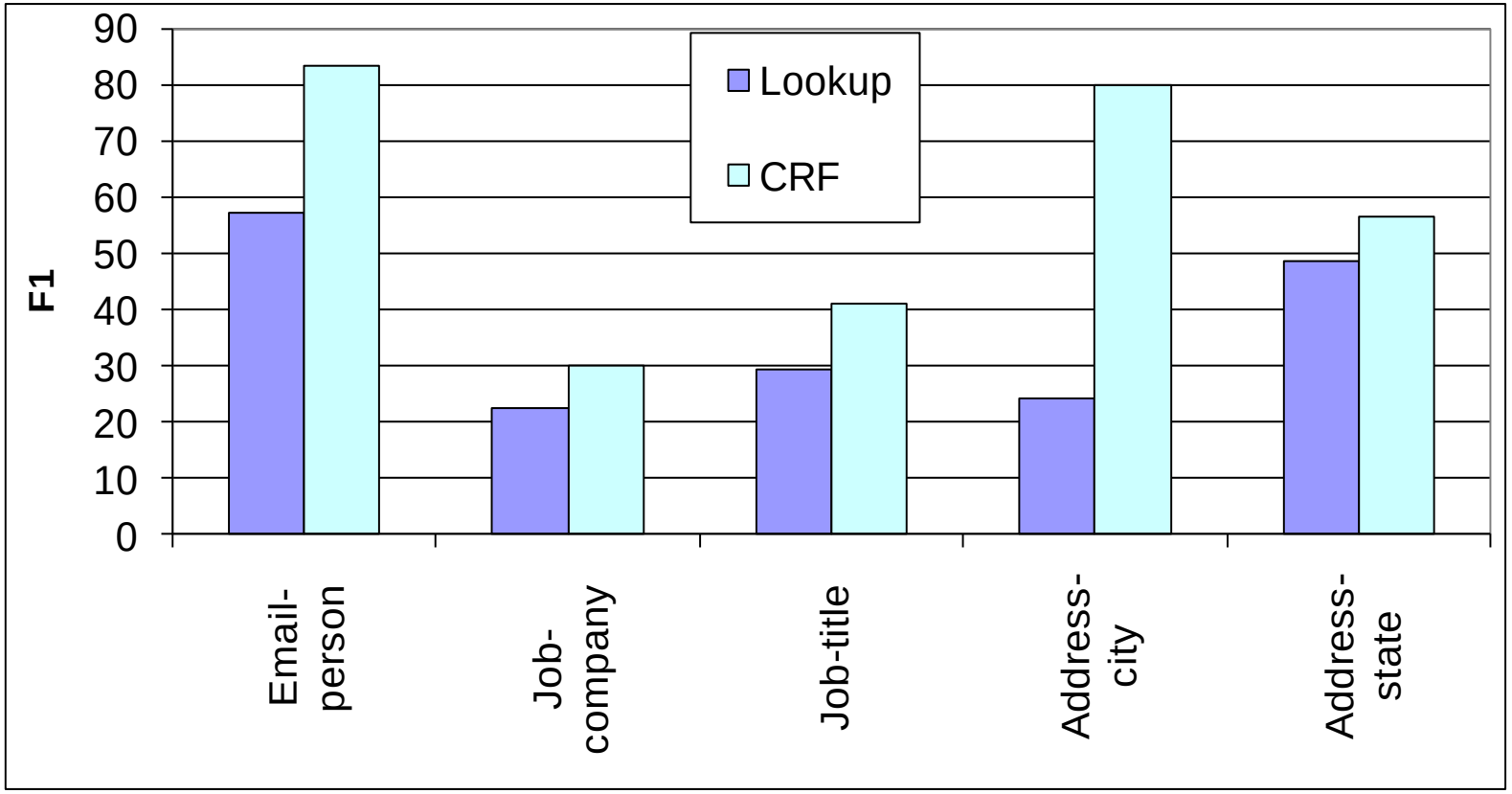
$$\text{argmax}_{\mathbf{S}} \sum_{j=1}^{k} \mathbf{W}.\mathbf{f}(y_j, \mathbf{x}, l_j, u_j, y_{j-1})$$

where, $S_j = (y_j, l_j, u_j), \quad l_j = u_{j-1} + 1$

Can be found efficiently using dynamic programming.

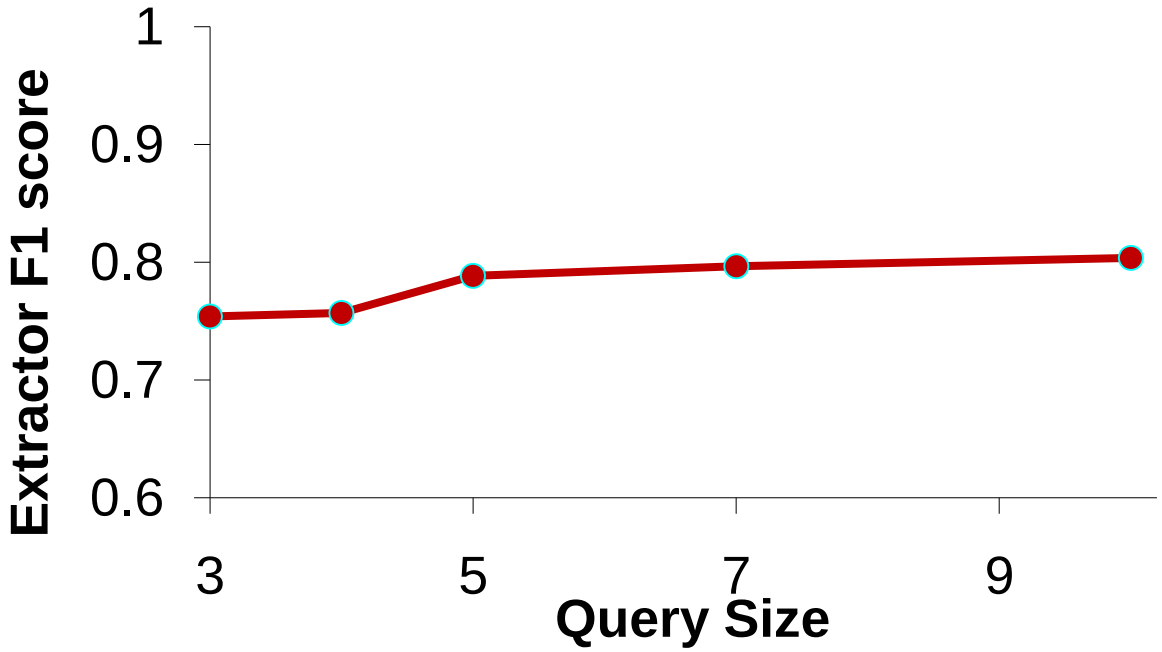
Sunita Sarawagi. Information extraction. *FnT Databases*, 1(3), 2008

Accuracy on some tasks



William W. Cohen and Sunita Sarawagi. Exploiting dictionaries in named entity extraction: Combining semi-markov extraction processes and data integration methods.*SIGKDD* 2004.

Extraction performance in the open domain



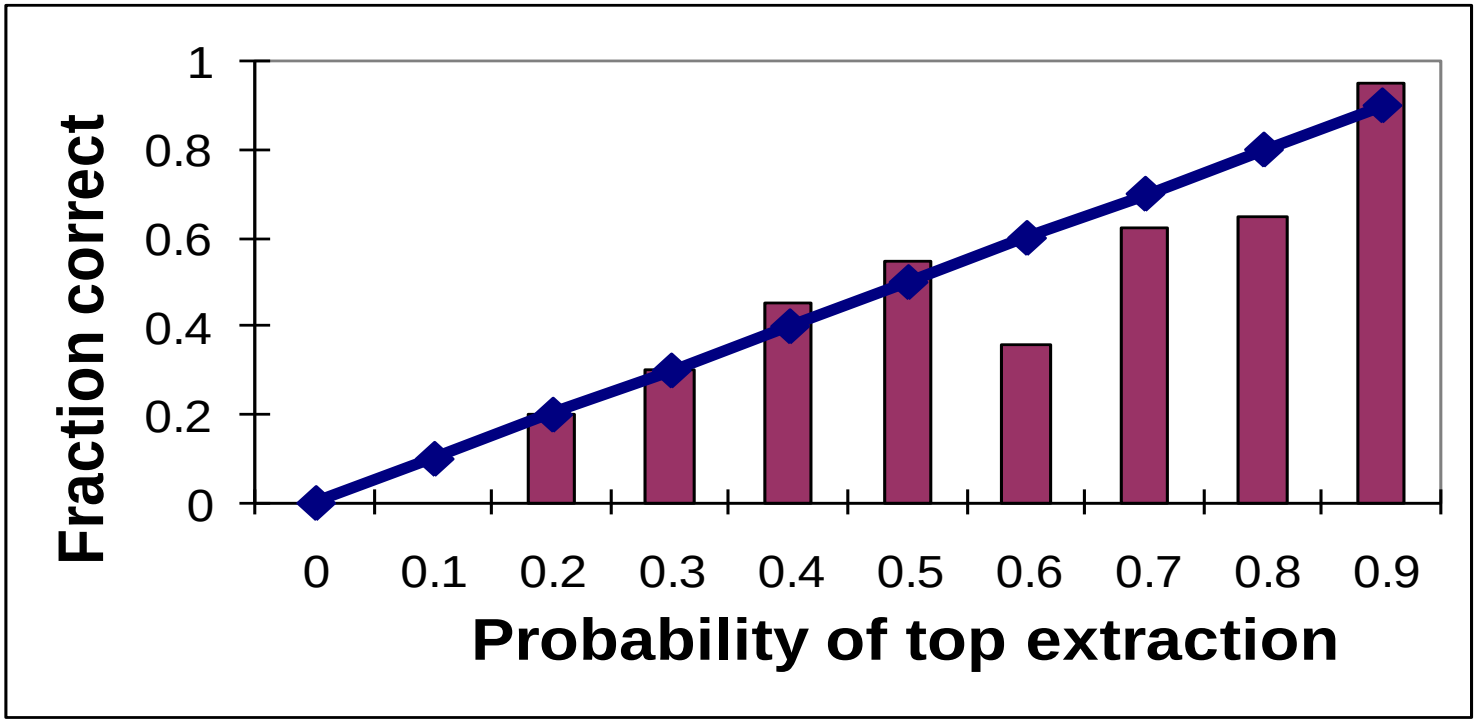
- Features: HTML tags, delimiters, segment length, alignment features
- Training data: Just the structured query records
- Accuracy: Close to 80% even with three query records.

Representing noisy extractions as imprecise databases

Imprecision of extraction

- No automated method can guarantee 100% extraction accuracy
- Imprecise databases
  - Can we capture extraction errors as well-calibrated confidence values attached with an extraction?
  - Can we represent these confidences compactly in a relational database and process queries efficiently over them?

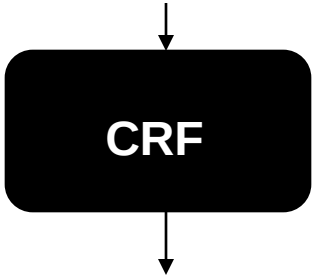
Well-calibrated Probability = Confidence



- Natural to ask, very difficult to obtain
  - Poor: Rule-based systems, HMMs
  - Good: CRFs

Extraction probability via CRF

Input: “52-A Bandra West Bombay 400 062”



HNO	AREA	CITY	PINCODE	PROB
52	Bandra West	Bombay	400 062	0.1
52-A	Bandra	West Bombay	400 062	0.2
52-A	Bandra West	Bombay	400 062	0.5
52	Bandra	West Bombay	400 062	0.2

Imprecision of extraction

- No automated method can guarantee 100% extraction accuracy
- Imprecise databases
  - Can we capture extraction errors as well-calibrated confidence values attached with an extraction?
  - Can we represent these confidences compactly in a relational database and process queries efficiently over them?

Segmentation-per-row model

(Rows: Uncertain; Columns: Exact)

HNO	AREA	CITY	PINCODE	PROB
52	Bandra West	Bombay	400 062	0.1
52-A	Bandra	West Bombay	400 062	0.2
52-A	Bandra West	Bombay	400 062	0.5
52	Bandra	West Bombay	400 062	0.2

Exact but impractical. We can have too many extraction possibilities!

One-row Model

Each column is a multinomial distribution

(Row: Exact; Columns: Independent, Uncertain)

HNO	AREA	CITY	PINCODE
52 (0.3)	Bandra West (0.6)	Bombay (0.6)	400 062 (1)
52-A (0.7)	Bandra (0.4)	West Bombay (0.4)	

e.g. P(52-A, Bandra West, Bombay, 400 062)
= 0.7 x 0.6 x 0.6 x 1.0 = 0.252

Efficient, compact model, closed form solution, but crude.

Multi-row Model

Rows: Uncertain; Columns: Independent, Uncertain

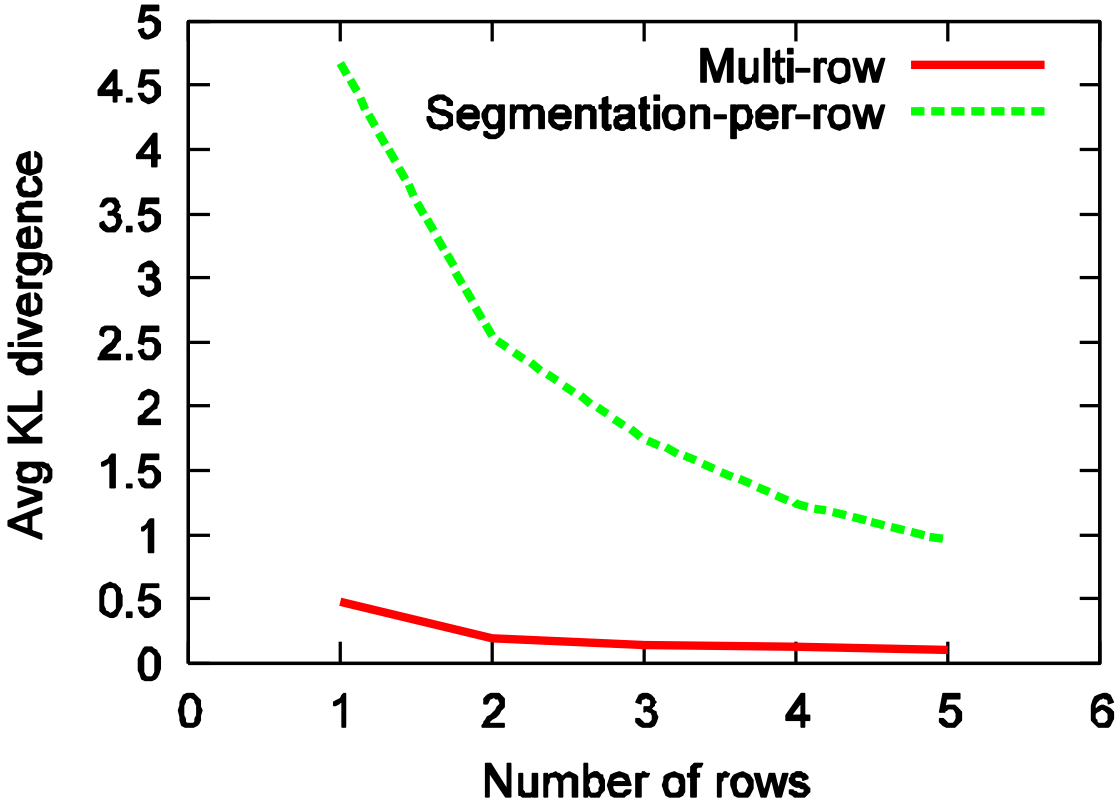
HNO	AREA	CITY	PINCODE	Prob
52 (0.2) 52-A (0.8)	Bandra West (1)	Bombay (1)	400 062 (1)	0.6
52 (0.5) 52-A (0.5)	Bandra (1)	West Bombay(1)	400 062 (1)	0.4

e.g. P(52-A, Bandra West, Bombay, 400 062)
= 0.833 x 1.0 x 1.0 x 1.0 x 0.6 + 0.5 x 0.0 x 0.0 x 1.0 x 0.4
= 0.50

EM like algorithm to transform CRF to this form

Rahul Gupta and Sunita Sarawagi. Curating probabilistic databases from information extraction models. In *VLDB*, 2006

Effectiveness of the multi-row model in approximating a CRF



- KL very high at m=1. One-row model clearly inadequate.
- Even a two-row model is sufficient in many cases.

Uncertainty in duplicate resolution

The de-duplication problem

- Given a large list of records, group together those referring to the same entity
  - People names and addresses in a warehouse
  - Product parts in an inventory database
- Impossible in many cases to resolve when two mentions refer to the same entity
  - Alistair MacLean and A Mclean duplicates?
- Even more challenging to resolve when group of mentions refer to the same entity.
  - Alistair MacLean
  - A Mclean
  - Alistair Mclean

Probability of two records being duplicates

- User specifies domain specific similarity features between two records
  - Cosine similarity
  - Edit-distance
  - 3-gram Jaccard
- Given record pair (t,t'), transform it into the similarity vector.
  - Example: ("Alistair MacLean", "A Mclean")→(0,8,3/16)
- Train a probabilistic classifier to predict Pr(y|t,t') from examples of duplicates and non-duplicates
  - Logisitic regression well-calibrated
  - Naïve Bayes, SVMs poorly calibrated

Probability over entity groupings

Alistair MacLean
A Mclean
Alistair Mclean
Alex Bell
Alexandar Bell
Alex Green
A Bell

Alistair MacLean	Alistair MacLean	Alistair MacLean
A Mclean	A Mclean	Alistair Mclean
Alistair Mclean	Alistair Mclean	Alex Bell
Alex Bell	Alex Bell	Alexandar Bell
Alexandar Bell	Alexandar Bell	Alex Green
Alex Green	A Bell	A Bell
A Bell	Alex Green	A Mclean
0.55	0.2	0.1

- Semantics of a grouping: G= g_1, g_2,…,g_n
  - All members in any g_i are duplicates of each other. No member outside g_i is a duplicate of any of its member
- Multiplying the probability of all duplicate and non-duplicate pairs yields poorly calibrated models.
  - Pairs are not independent of each other!

Scoring duplicate groups

$$\text{Probability of G: } g_1, \ldots, g_n = \frac{\exp(\text{score}(G))}{\sum_{G'} \exp(\text{score}(G'))}$$

$$\text{score}(G) = \sum_i \text{gscore}(g_i)$$

All pairs scores

$$\text{gscore}(g_i) = \sum_{t,t' \in g_i} \text{sim}(t,t') - \lambda \sum_{t \in g_i, t' \notin g_i} \text{sim}(t,t')$$

A more robust scoring function

$$\text{gscore}(g_i) = \min_{t,t' \in g_i} \text{sim}(t,t') - \lambda \max_{t \in g_i, t' \notin g_i} \text{sim}(t,t')$$

Queries over imprecise duplicates

- Find the most likely de-duplication grouping.
  - NP-hard for most scoring functions, akin to typical clustering and graph partitioning problems
- Find the K largest groups
  - Example:
    - Find the three most frequently cited organization in the past six months of news stories
    - The ten most prolific authors in a citation corpus
  - Very challenging..
    - NP-hard to compute the score of a given K clusters

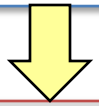
K largest groups w/o full deduplication

- Prune away tuples guaranteed not to be part of the answer
  - Depend on efficient upper and lower bounds to the pair similarity function
- Linearly embed all records based on sim(t,t')
- Groups → segmentation of the linear order

Sunita Sarawagi, Vinay S Deshpande, and Sourabh Kasliwal. Efficient top-k count queries over imprecise duplicates. In *EDBT*, 2009

Linear embedding.

Alistair MacLean
A Mclean
Alistair Mclean
Alex Bell
Alexandar Bell
Alex Green
A Bell



1	A Mclean
2	Alistair MacLean
3	Alistair Mclean
4	Alex Green
5	Alex Bell
6	Alexandar Bell
7	A Bell

Alistair MacLean
A Mclean
Alistair Mclean
Alex Bell
Alexandar Bell
Alex Green
A Bell

0.55

Alistair MacLean
A Mclean
Alistair Mclean
Alex Bell
Alexandar Bell
A Bell
Alex Green

0.2

Alistair MacLean
Alistair Mclean
Alex Bell
Alexandar Bell
Alex Green
A Bell
A Mclean

0.1

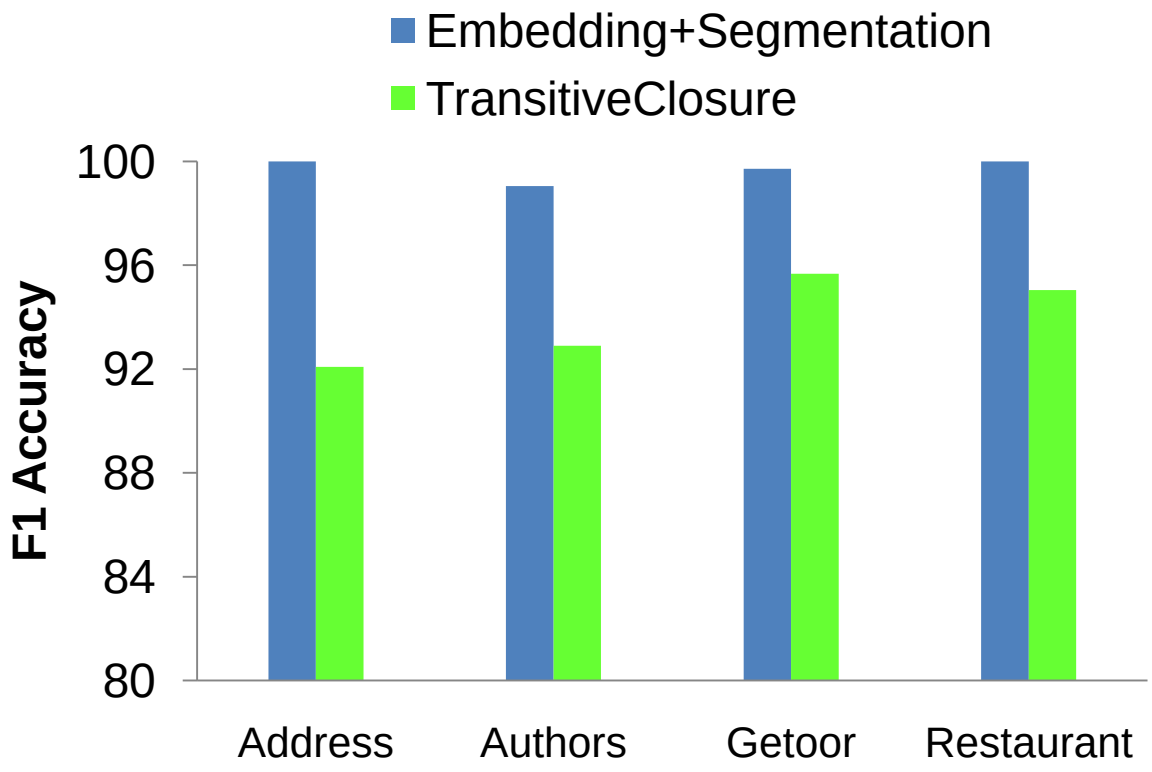
All three groupings are segmentation of this linear order

Data reduction with increasing K

K	Iteration-1				Iteration-2			
	n	m	M	n'	n	m	M	n'
1	67.22	1	11970	1.70	1.56	1	11970	0.98
5	67.22	5	6896	5.55	4.60	5	6896	3.17
10	67.22	10	6101	6.49	5.33	10	6573	3.54
50	67.22	50	3396	13.67	10.77	51	3860	6.93
100	67.22	100	2674	17.59	13.80	101	2838	10.06
500	67.22	543	1166	31.34	24.69	547	1308	19.39
1000	67.22	1206	720	38.02	30.06	1166	802	25.50

Data reduces to one-tenth with K=100 and to one-hundredth with K=1.

Accuracy of linear embedding



- Embedding has close to 100% accuracy on real-life dataset

Summary

Information extraction and entity resolution

- Very challenging to automate
  - Success depends on being able to combine soft clues from diverse sources
  - Conditional Random Fields: a unified, elegant solution
- Impossible to guarantee 100% accuracy→ Reflect imprecision to the query output
  - IE: CRFs provide well-calibrated probabilities
    - Transformation to row/column uncertainty models for storing in a relational database
  - De-deduplication: existing models expensive
    - Transformation to a linear embedding for efficient query processing.

What next?

- Combine uncertainty from multi-stage operations
  - Extraction, entity resolution, canonicalization.
  - Uncertainty in the source, correlation between sources
- Sound probability models for deduplication
  - Existing models are not well-calibrated.

- Decision making tools (Advanced querying, OLAP, forecasting, clustering, classification) over uncertain data.

Thank you.