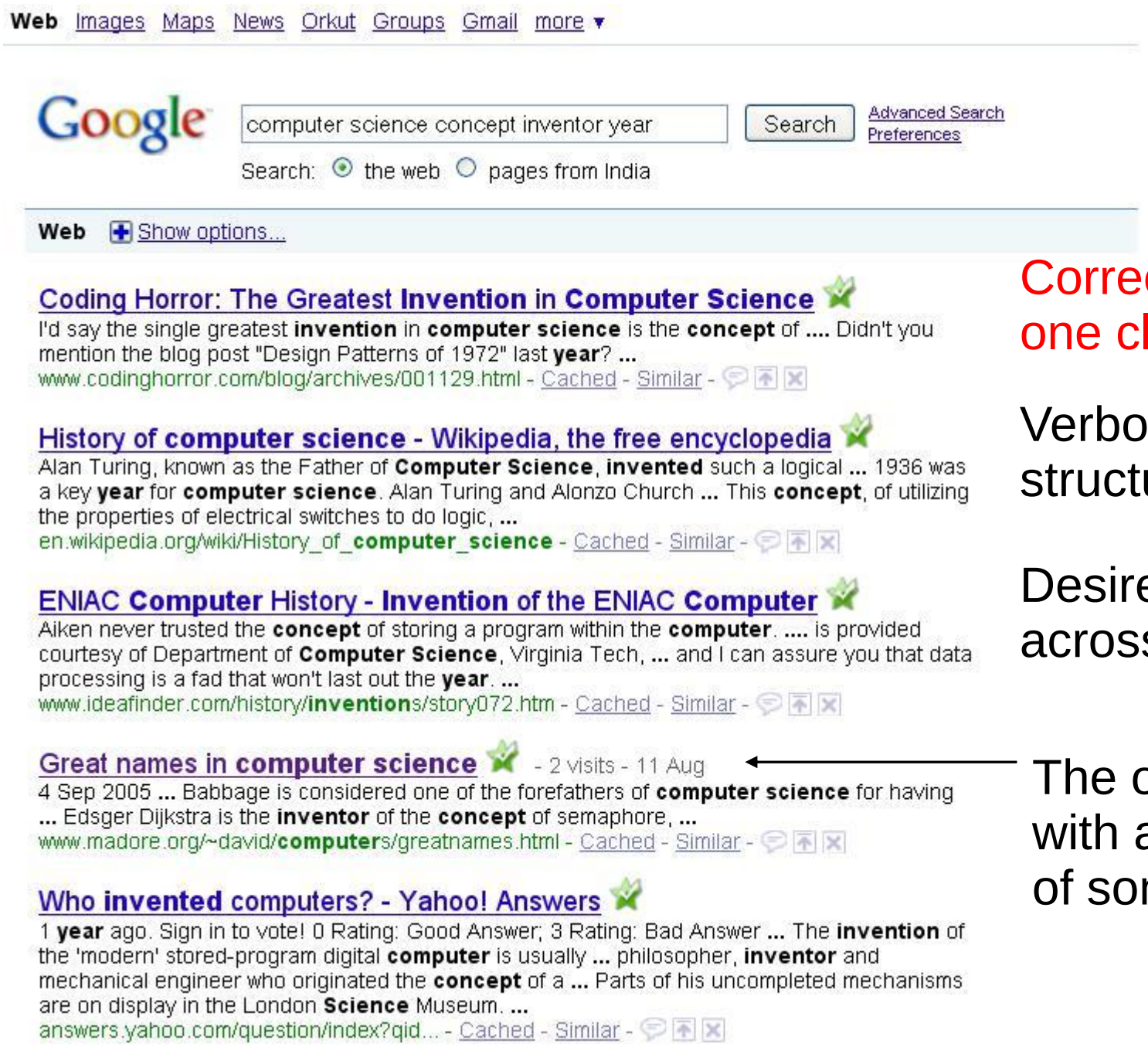
Answering Table Augmentation Queries using Unstructured Lists on the Web

Rahul Gupta and Sunita Sarawagi

Dept. of Computer Science and Engineering, IIT Bombay

Example: Compiling a List of famous CS inventors and inventions

Person	Concept/Invention
Alan Turing	Turing Machine
Seymour Cray	Supercomputer
E. F. Codd	Relational Databases
Tim Berners-Lee	WWW
Charles Babbage	Babbage Engine



Correct answer is not one click away.

Verbose articles, not structured tables

Desired records spread across many documents

The only document with an unstructured list of some desired records

The only list in one of the retrieved pages

Harold Abelson

(web page) Professor of Computer Science and Engineering at the Mas
Sussman, Abelson is the author of *Structure and Interpretation of Comp*

Eric Allman

(web page) Eric Allman is the main author of the sendmail program, wh
(emails), although certain alternatives have become popular, such as D
McKusick's partner.

Charles Babbage

Born: Monday, December 26, 1791, in London (England). Died: Wedne
considered one of the forefathers of computer science for having desig
(with the help of Ada Lovelace) the analytical engine, which, although it
(mechanical) computer. See also Babbage's biography on the MacTuto

John W. Backus

Born: Wednesday, December 3, 1924, in Philadelphia, Pennsylvania (U
which gave birth to the language FORTRAN (the oldest programming la
Calculus, and, of course, assembler). John Backus is the 1977 recipier
of the IEEE Computer Society's Pioneer Award.

Tim Berners-Lee

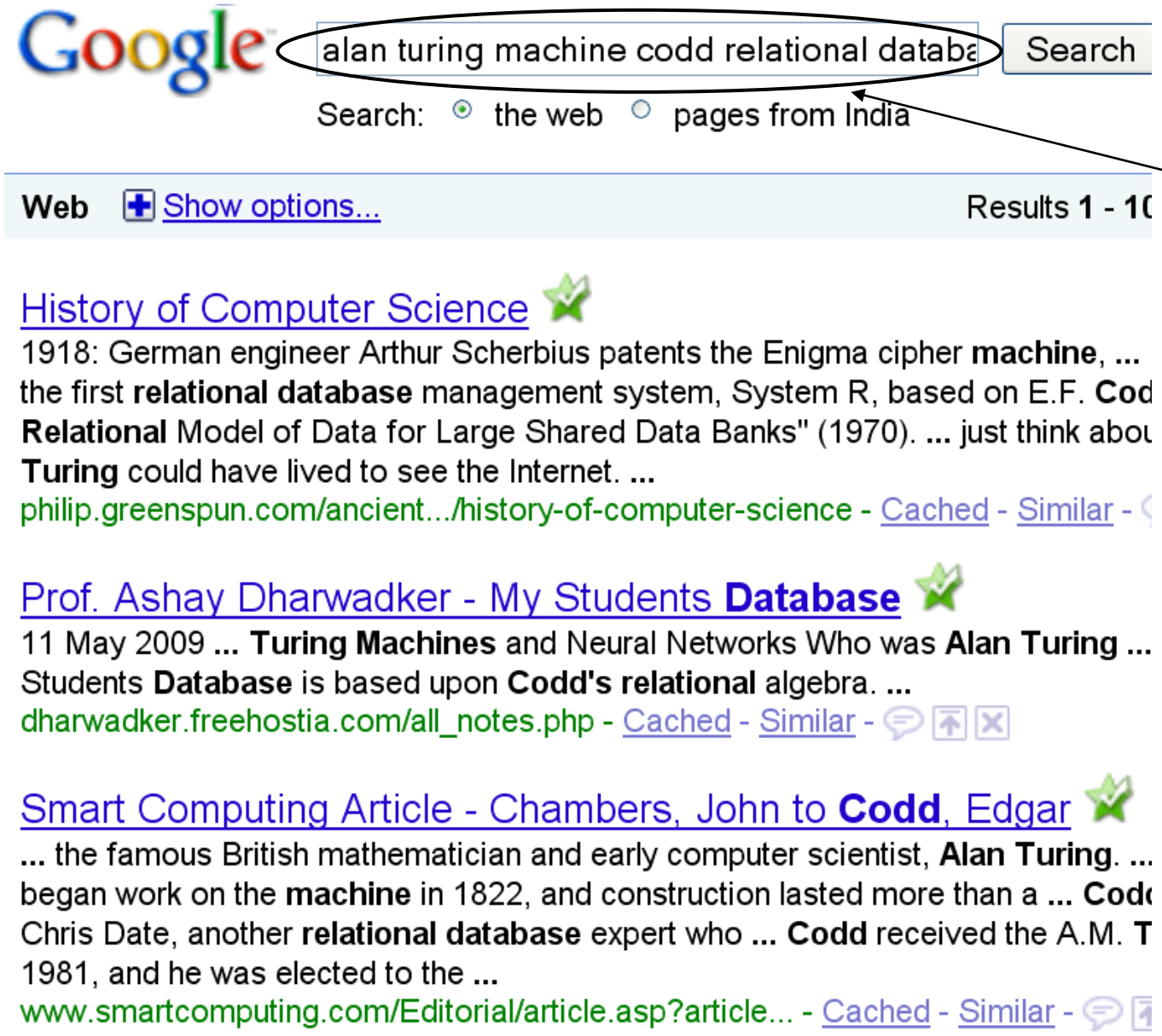
(web page) Born: Wednesday, June 8, 1955, in London (UK). Tim Bern

Highly relevant Wikipedia table not retrieved in the top-k

Person	Achievement	Ach. Date
John Atanasoff	Built the first electronic digital computer, the Atanasoff–Berry Computer, though it was neither programmable nor Turing-complete.	1939
Charles Babbage	Designed the Analytical Engine and built a prototype for a less powerful mechanical calculator.	1822 1837
John Backus	Invented FORTRAN (*For*mula *Tran*slation), the first practical high-level programming language, and he formulated the Backus-Naur form that described the formal language syntax.	1954 1963
George Boole	Formalized Boolean algebra, the basis for digital logic and computer science.	1830~

Ideal answer should be integrated from these incomplete sources

Attempt 2: Include samples in query



Known examples

Documents relevant only to the keywords

Ideal answer still spread across many documents

Related applications: Augmenting Freebase/Wikipedia Tables

Did you know?

More than 20,000 brands of beer

Worldwide, 20,000 brands of beer are brewed in 180 styles, from ales, lagers, pilsner and stouts to bitters, cream ales and iced beers.

Beer has been a popular beverage for a long time. Babylonian clay tablets show detailed recipes of beer making in 4300 BC. Beer was also brewed by the ancient Chinese, Assyrians and Incas.

An Egyptian text of 1600 BC gives 100 medical prescriptions using beer. A few years ago, the New Castle Brewery in England brewed 1,000 bottles

Table Augmentation Problem

- A user provides a few (~3) structured records.
- Goal is to return a single table with more such records ranked by relevance.
- Large number of relevant records can be extracted from multiple semi/unstructured sources like HTML lists/tables.
  - Have to integrate across multiple sources.
- Multi-attribute version of Google Sets.
- Goal similar to Google-Squared (launched mid-May 2009).

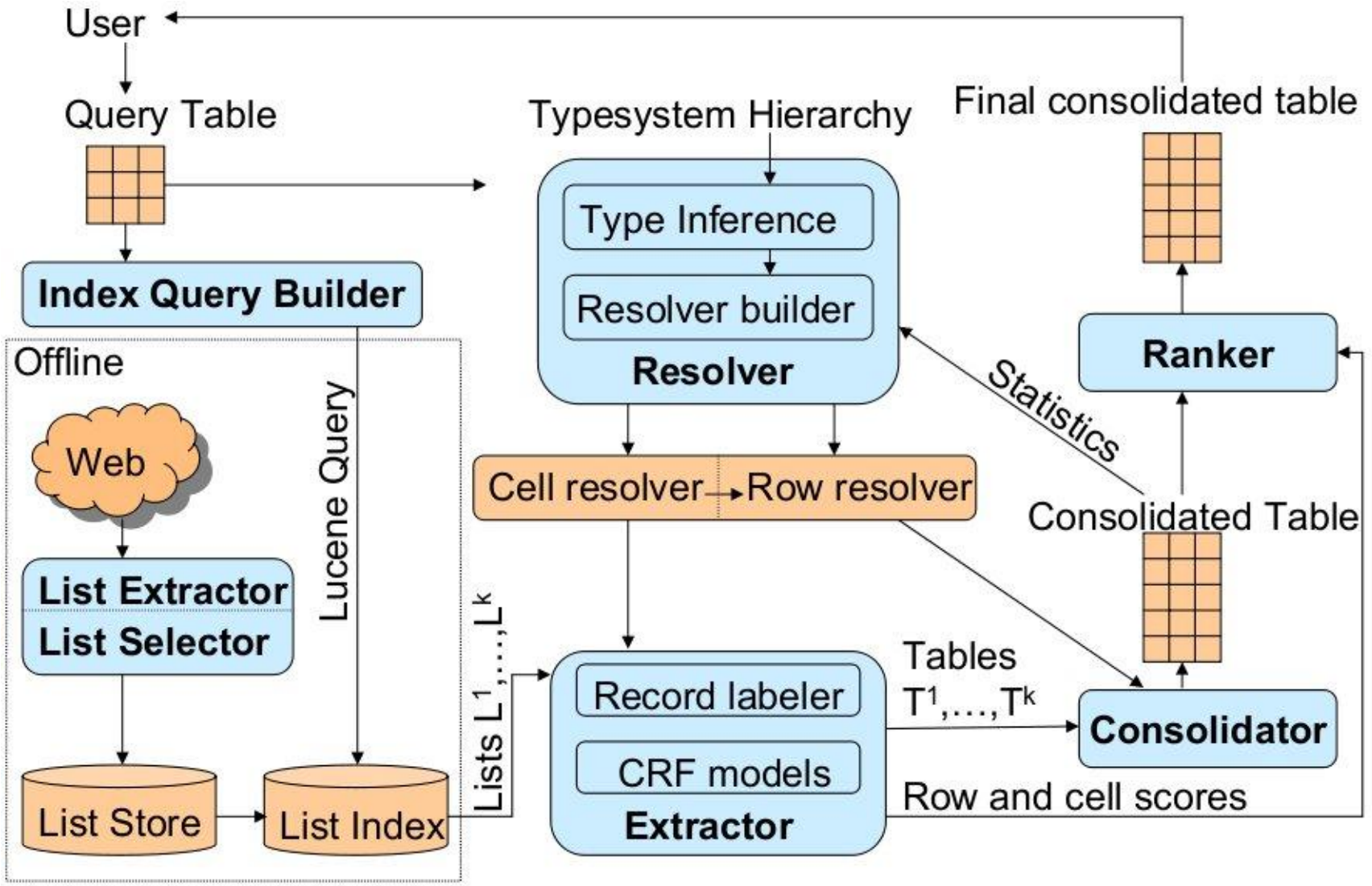
Our Contributions

- A new web-scale task: Table Augmentation.
- An end-end World Wide Tables (WWT) system that answers table augmentation queries.
- Achieves a runtime of ~30s. Recall ~ 55% when reconstructing Wikipedia tables from 3 samples.

This talk: Answering queries only using HTML list sources

- Many lists have records that are absent in tables
- Lists harder to process than tables
- Current WWT system uses both lists and tables

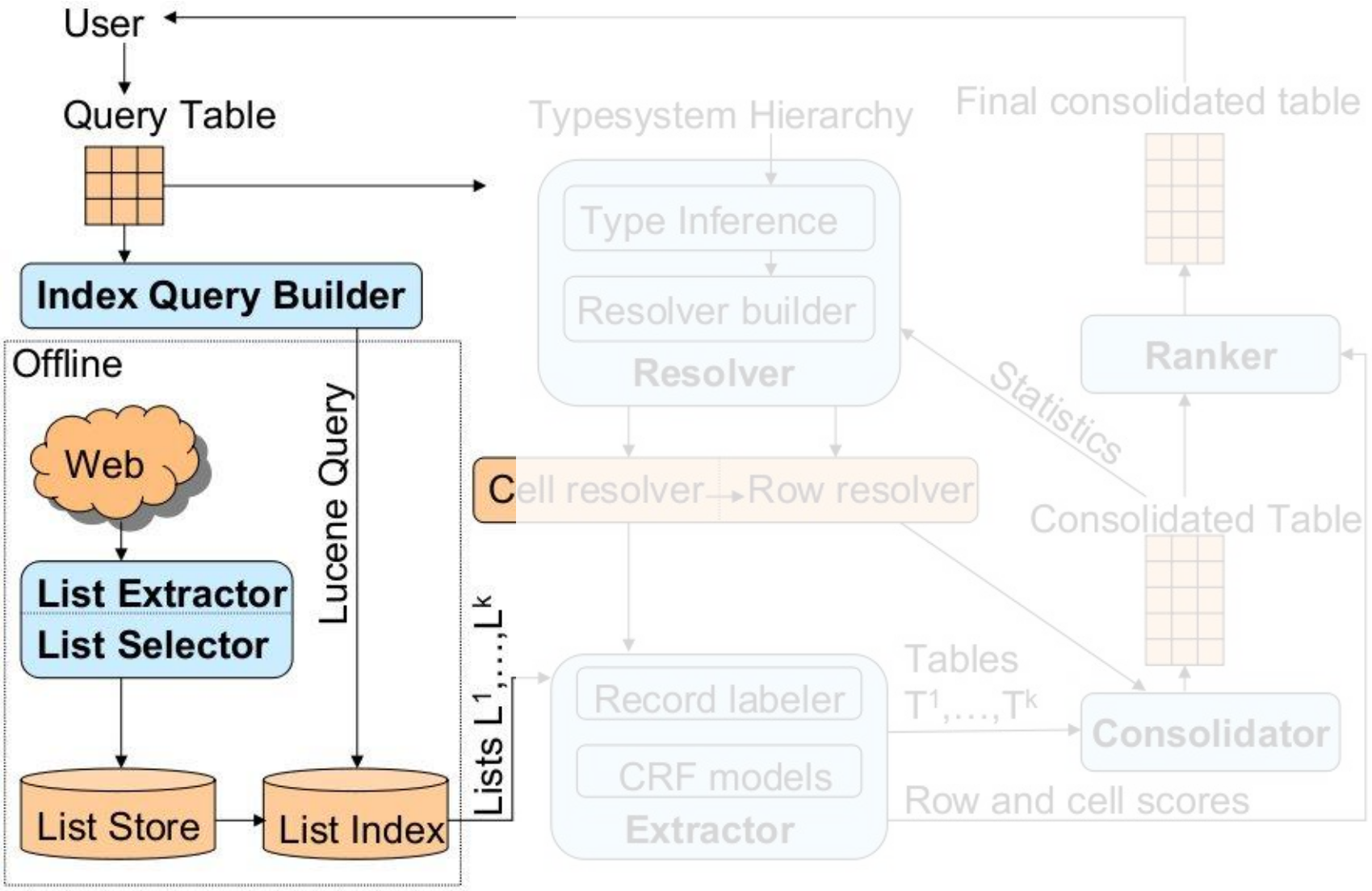
WWT: Architecture



Offline List Extraction and Indexing

- Retrieving and processing HTML lists instead of documents
  - We indexed 16M lists extracted from 500M documents.

Step 0: Index Probe



Step 1: Extraction

Extracting required columns from list records

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.

↖ Lists are often human generated.

Step 1: Extraction

Extracting required columns from list records

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.

Rule-based extractor insufficient. Statistical extractor needs training data.

Generating that is also not easy!

Extraction: Contributions

- We adapt Conditional Random Fields for extraction
  - No explicitly labeled training data. We present robust techniques to generate reliable training data from the small query.
  - Exploit regularity inside a source using multiple sequence alignment.
  - Use content overlap across sources to strengthen sources with less labeled data.

Step 2: Consolidation

Merging the extracted tables into one

Cornell University	Ithaca
State University of New York	Stony Brook
New York University	New York City
Binghamton University	Binghamton

+

SUNY	Stony Brook
New York University (NYU)	New York
RPI	Troy
Columbia University	New York
Syracuse University	Syracuse

=

Cornell University	Ithaca
State University of New York OR SUNY	Stony Brook
New York University OR New York University (NYU)	New York City OR New York
Binghamton University	Binghamton
RPI	Troy
Columbia University	New York
Syracuse University	Syracuse

Merging duplicates

Consolidation: Contributions

- Design a Bayesian Network for resolution
  - Parameters set automatically on a per-source basis.
  - Naturally handles missing columns.

Step 3: Ranking

Ordering consolidated records by relevance

		Support	Total Row Confidence
-	NYC	9	0.95
Cornell University	Ithaca	2	0.90
State University of New York OR SUNY	Stony Brook	2	0.80
New York University OR New York University (NYU)	New York City OR New York	3	0.82
Binghamton University	Binghamton	1	0.90
RPI	Troy	1	0.94
Columbia University	New York	2	0.91
Syracuse University	Syracuse	1	0.85

- Reward records supported by multiple sources
- Penalize records with only common ‘spam’ columns
  e.g. City, State
- Reward records confidently extracted by the statistical extractor

Extraction: Labeled data generation

Lists are unlabeled. Labeled records needed to train a CRF

A fast but naïve approach for generating labeled records

New York University	New York
Monroe College	Brighton
State University of New York	Stony Brook

Query about colleges in NY

- New York Univ. in NYC
- Columbia University in NYC
- Monroe Community College in Brighton
- State University of New York in Stony Brook, New York.

Fragment of a relevant list source

Extraction: Labeled data generation

A fast but naïve approach

New York University	New York
Monroe College	Brighton
State University of New York	Stony Brook

- New York Univ. in NYC
- Columbia University in NYC
- Monroe Community College in Brighton
- State University of New York in Stony Brook, New York.

Another match for New York

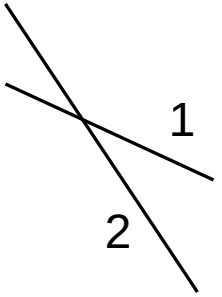
Another match for New York University

- In the list, look for matches of every query cell.

Extraction: Labeled data generation

A fast but naïve approach

New York University	New York
Monroe College	Brighton
State University of New York	Stony Brook

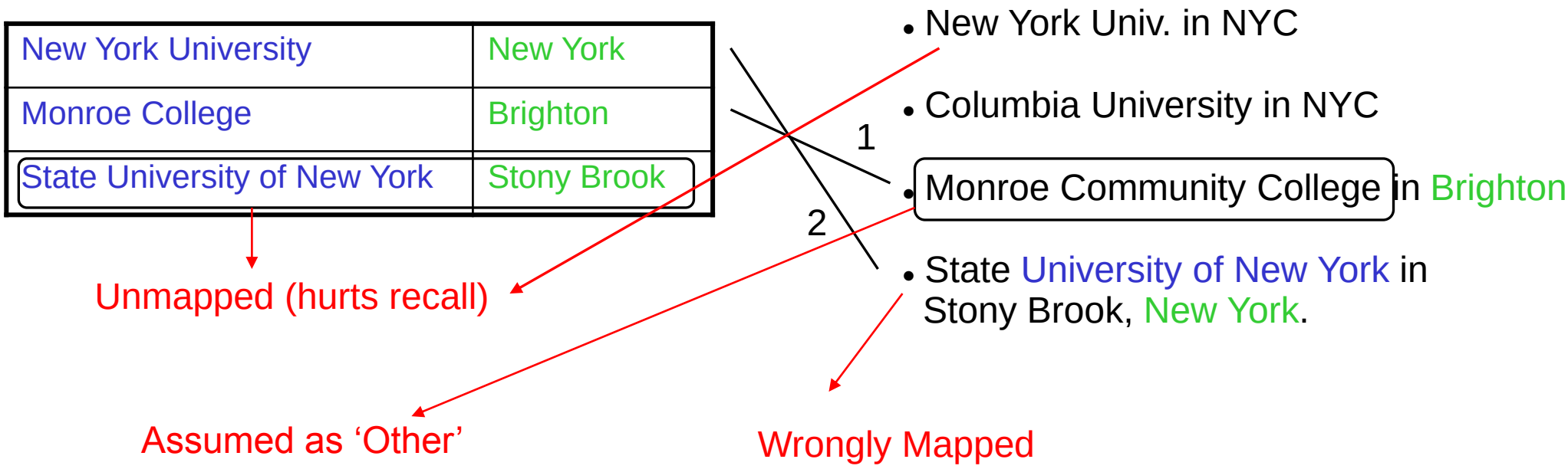


- New York Univ. in NYC
- Columbia University in NYC
- Monroe Community College in Brighton
- State University of New York in Stony Brook, New York.

- In the list, look for matches of every query cell.
- Greedily map each query row to the best match in the list

Extraction: Labeled data generation

A fast but naïve approach



- Hard matching criteria has significantly low recall
  - Missed segments.
  - Does not use natural clues like Univ = University
- Greedy matching can be lead to really bad mappings

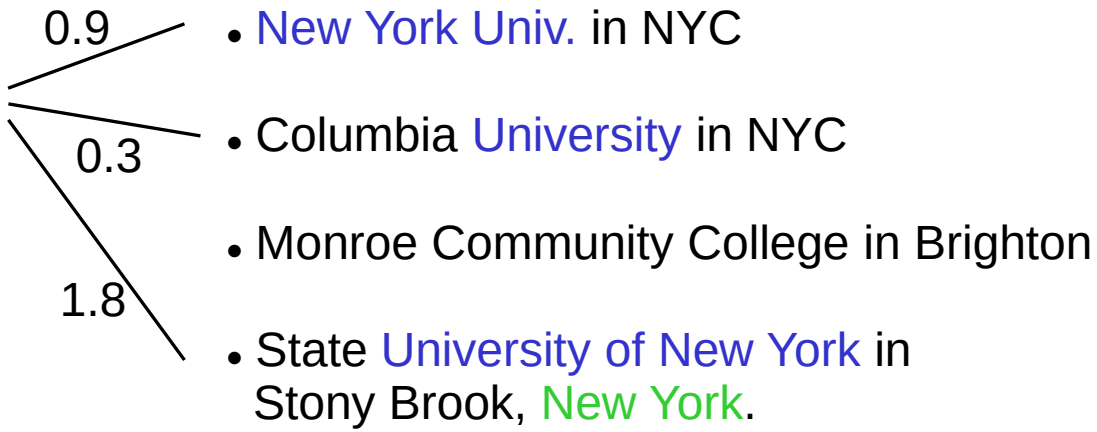
Generating labeled data: Soft approach

New York University	New York
Monroe College	Brighton
State University of New York	Stony Brook

- New York Univ. in NYC
- Columbia University in NYC
- Monroe Community College in Brighton
- State University of New York in Stony Brook, New York.

Generating labeled data: Soft approach

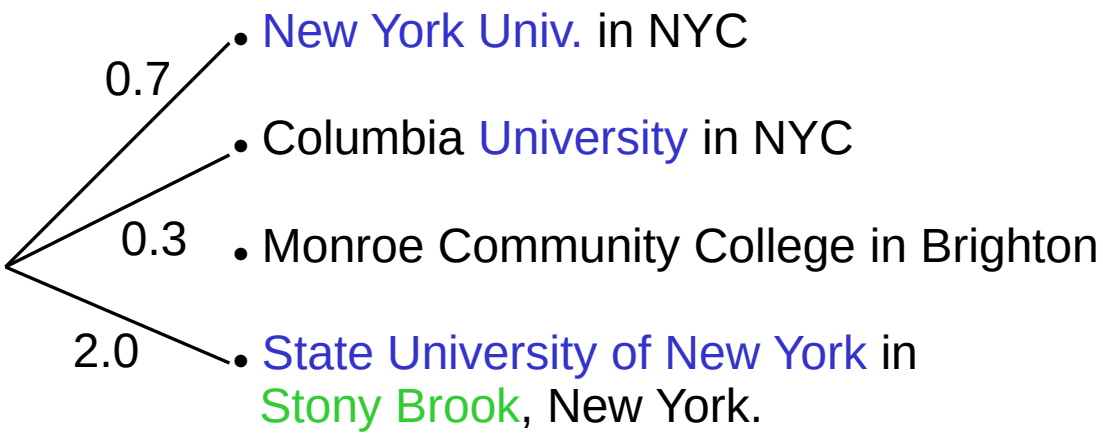
New York University	New York
Monroe College	Brighton
State University of New York	Stony Brook



- Compute the best match score for each query and source row
  - Score not hard but a continuous value
  - Computed as probabilities P(cell,string) by cell-string resolvers.
  - Resolver uses similarity functions specific to column-type

Generating labeled data: Soft approach

New York University	New York
Monroe College	Brighton
State University of New York	Stony Brook

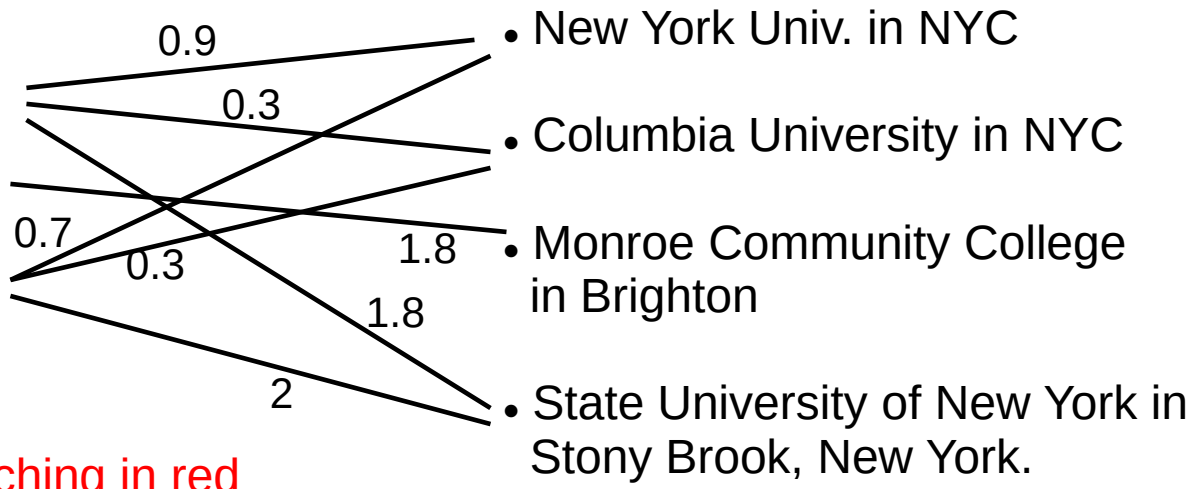


Compute the best match score for each query and source row

- Score not hard but a continuous value
- Computed as probabilities P(cell,string) by cell resolvers
- Resolver uses similarity functions specific to column-type

Generating labeled data: Soft approach

New York University	New York
Monroe College	Brighton
State University of New York	Stony Brook



Greedy matching in red

- Compute the maximum weight matching
  - Better than greedily choosing the best match for each row
- Soft string-matching increases the labeled candidates significantly
  - Vastly improves recall, leads to better extraction models.

Extractor (Contd.)

- Use CRF on the generated labeled data
- Feature Set: delimiter and HTML tokens in a window around labeled segments.

Extractor: Overlap across sources

Some sources have too few labeled segments or target columns

New York University	New York City
University of Buffalo	Buffalo
RPI	Troy

Query

- New York Univ. in New York City.
- Columbia University, New York City
- Binghamton University.
- Cornell University in Ithaca.
- Syracuse University in Syracuse.
- RPI in Troy.

(Strongly labeled source)

- SUNY, Buffalo, founded 1846
- SUNY, Albany, founded 1844
- SUNY, Stony Brook, founded 1957
- Cornell University, Ithaca, founded 1865
- NYU, NYC, founded 1831
- Syracuse University, Syracuse, founded 1870

(Weakly labeled source)

Extractor: Overlap across sources

New York University	New York City
University of Buffalo	Buffalo
RPI	Troy

New York Univ.	New York City
Columbia University	New York City
Binghamton University	-
Cornell University	Ithaca
Syracuse University	Syracuse
RPI	Troy

-	Buffalo
-	Albany
-	Stony Brook
-	Ithaca
-	NYC
-	Syracuse

Useless without school names

But sources are related, so they have content overlap

Exploiting Overlap: Staged Extraction

1. Order lists from strong to weak.
2. Build the model for next list as before.
3. Extract high confidence records from the list

- New York Univ. in New York City.
- Columbia University, New York City.
- Binghamton University.
- Cornell University in Ithaca.
- Syracuse University in Syracuse.
- RPI in Troy.

New York Univ.	New York City	✓
Columbia University	New York City	
Binghamton University	-	
Cornell University	Ithaca	✓
Syracuse University	Syracuse	✓
RPI	Troy	✓

Exploiting Overlap: Staged Extraction

4. Merge high confidence records with the query (done by consolidator)

High confidence records avoid polluting the query.

Query

New York University	New York City
University of Buffalo	Buffalo
RPI	Troy

+

New York Univ.	New York City
Cornell University	Ithaca
Syracuse University	Syracuse
RPI	Troy

=

New York University OR New York Univ.	New York City
Cornell University	Ithaca
Syracuse University	Syracuse
University of Buffalo	Buffalo
RPI	Troy

(Enhanced Query)

Exploiting Overlap: Staged Extraction

5. Re-label weak sources with enhanced query.

Repeat for all weak sources

New York University OR New York Univ.	New York City
Cornell University	Ithaca
Syracuse University	Syracuse
University of Buffalo	Buffalo
RPI	Troy

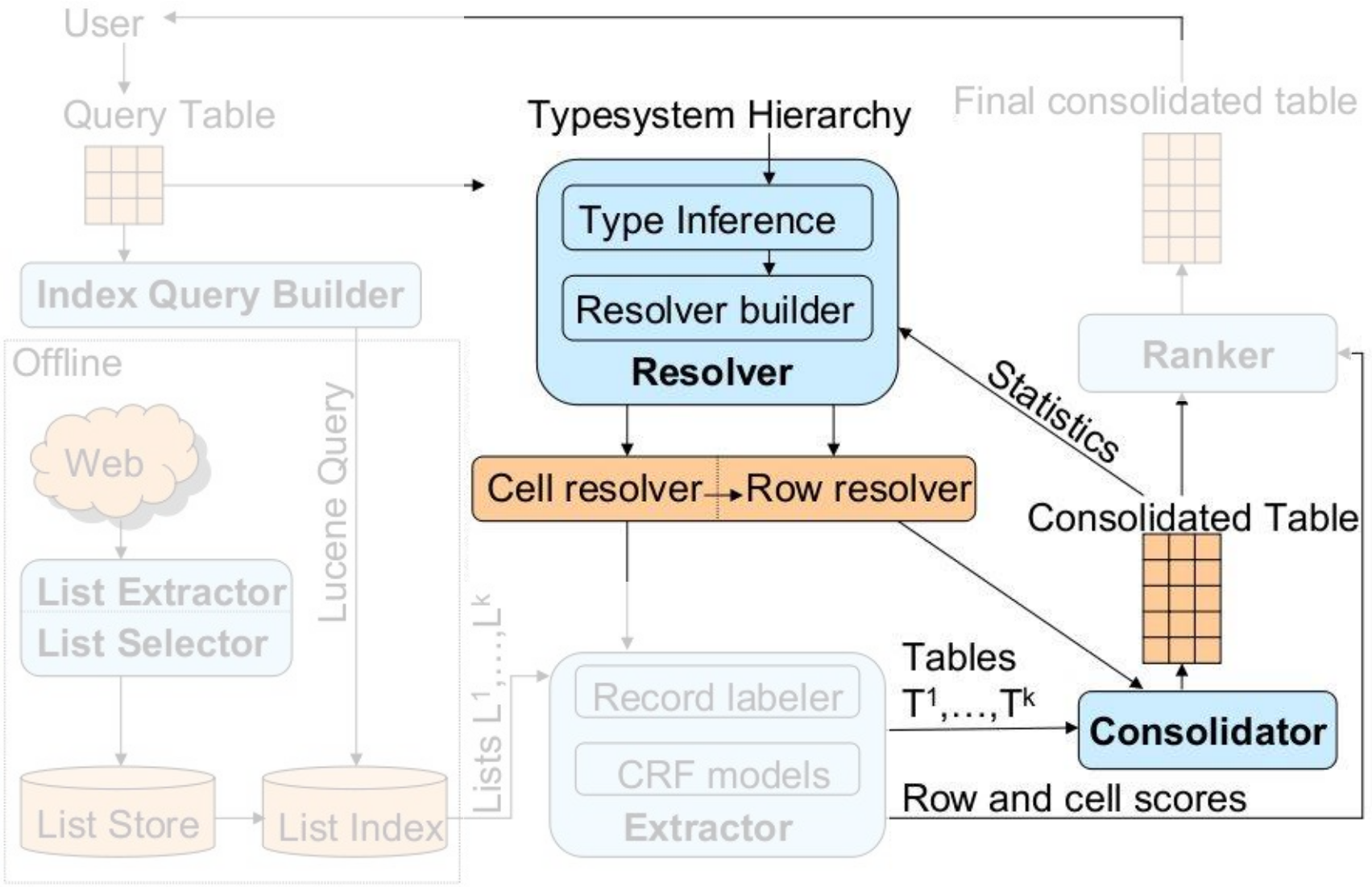
(Enhanced Query)

- SUNY, Buffalo, founded 1846
- SUNY, Albany, founded 1844
- SUNY, Stony Brook, founded 1957
- Cornell University, Ithaca, founded 1865
- NYU, NYC, founded 1831
- Syracuse University, Syracuse, founded 1870

SUNY	Buffalo
SUNY	Albany
SUNY	Stony Brook
Cornell University	Ithaca
NYU	NYC
Syracuse University	Syracuse

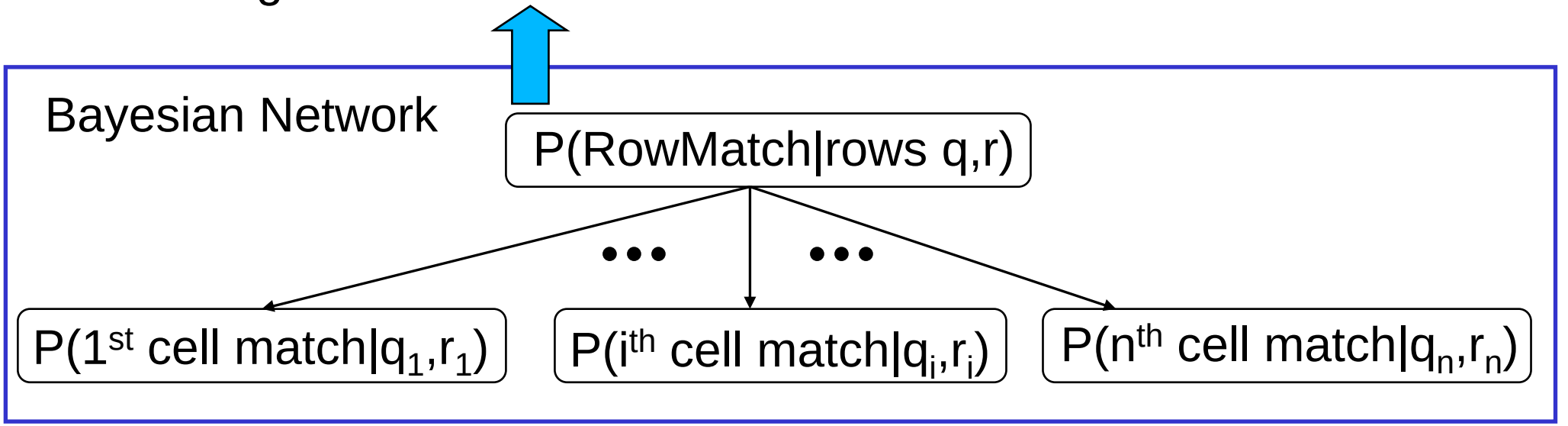
(Re-labeled weak source)

Resolver


User
Query Table
Typesystem Hierarchy
Final consolidated table
Type Inference
Resolver builder
Resolver
Index Query Builder
Ranker
Offline
Statistics
Web
Lucene Query
Cell resolver
Row resolver
Consolidated Table
List Extractor
List Selector
Lists L1,…,Lk
Record labeler
Tables
T1,…,Tk
Consolidator
CRF models
Row and cell scores
List Store
List Index
Extractor

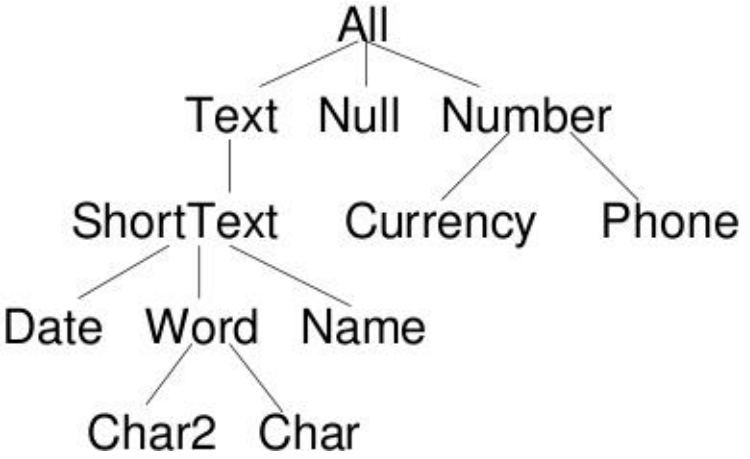
Resolver

Used during consolidation



Cell-level probabilities

- Parameters automatically set using list statistics
- Derived from user-supplied type-specific similarity functions



Ranking: Additive Criteria

- Just sort by support across lists
  - Junk records that only have spam columns (city/state) come on top. (NY, NYC)
  - All columns assumed equally important.
  - Ignores confidence of extraction (Rochester vs Cornell)
  - Also, confidence decreases when more columns present

School	Location	State	Merged Row Confidence	Support
-	-	NY	0.99	9
-	NYC	New York	0.95	7
New York Univ. OR New York University	New York City OR New York	New York	0.85	4
University of Rochester OR Univ. of Rochester,	Rochester	New York	0.50	2
University of Buffalo	Buffalo	New York	0.70	2
Cornell University	Ithaca	New York	0.76	1

Ranking: 'Cell-SoftMax' Criteria

- Score(Row r): Each non-null cell c contributes

Importance(column c) x Cell-extraction-confidence(c, support)

More for text, long strings

Obtained from CRFs. Support included via noisy-OR

Favors more non-null cells

School	Location	State	Merged Row Confidence	Support
New York Univ. OR New York University (0.90)	New York City OR New York (0.95)	New York (0.98)	0.85	4
University of Buffalo (0.88)	Buffalo (0.99)	New York (0.99)	0.70	2
Cornell University (0.92)	Ithaca (0.95)	New York (0.99)	0.76	1
University of Rochester OR Univ. of Rochester, (0.80)	Rochester (0.95)	New York (0.99)	0.50	2
-	-	NY (0.99)	0.99	9
-	NYC (0.98)	New York (0.98)	0.95	7

Gain in recall at 10% error vs Additive: +10%

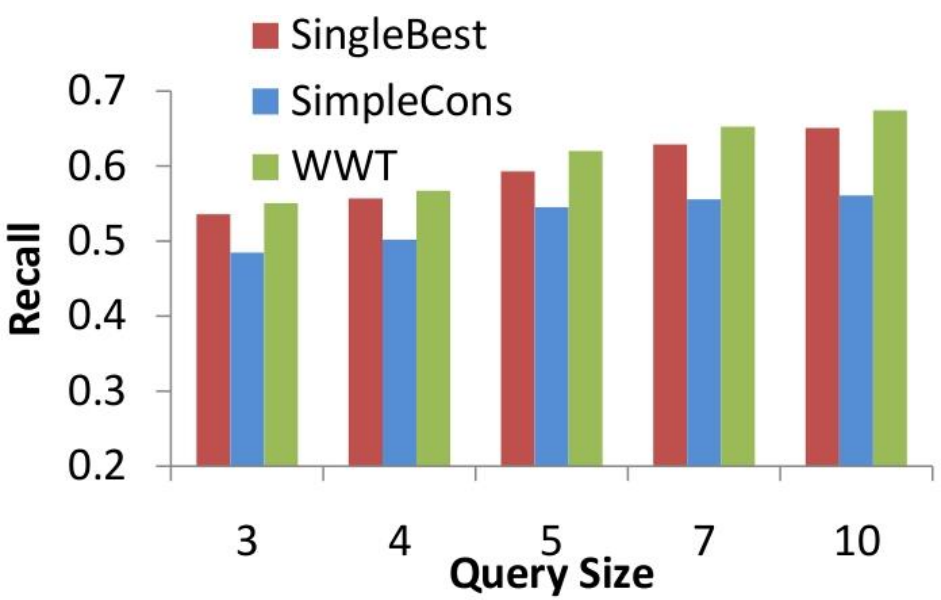
Experiments

- Aim: Reconstruct Wikipedia tables from only a few sample rows.
- Metric: Retrieve as many rows as the ground truth table. Measure recall.
- Sample queries
  - West Wing: Character name, Actor name, Season
  - Oil spills: Tanker, Region, Time
  - Golden Globe Awards: Actor, Movie, Year
  - Bond Cars: Brand, Movie

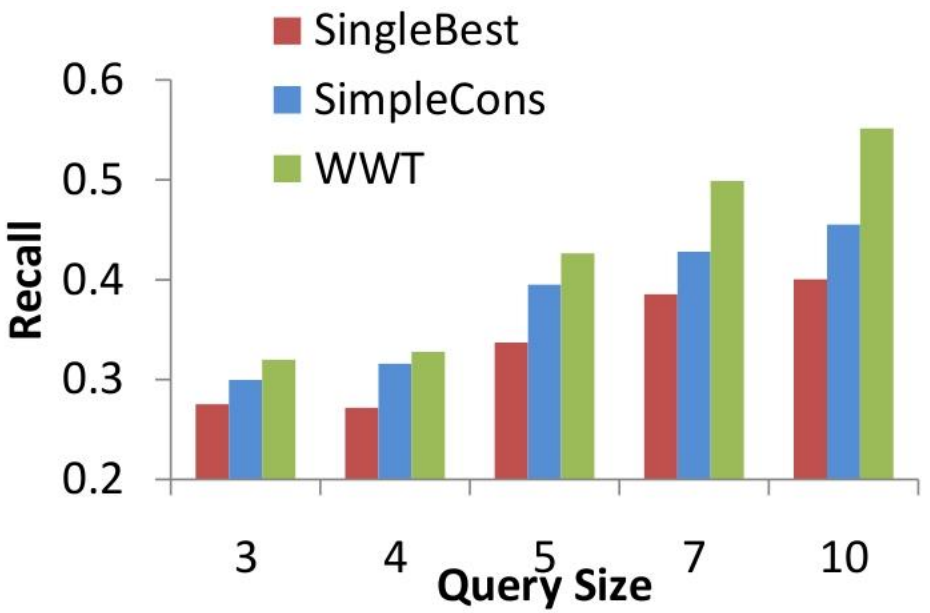
Experiments: Dataset

- Corpus:
  - 16M lists from 500M pages.
  - 45% of lists retrieved by index probe are irrelevant.
- Query workload
  - 65 queries. Ground truth hand-labeled by 10 users over 1300 lists.
  - 27% queries not answerable with one list (difficult).
  - True consolidated table = 75% of Wikipedia table, 25% new rows not present in Wikipedia.

Overall performance



All Queries

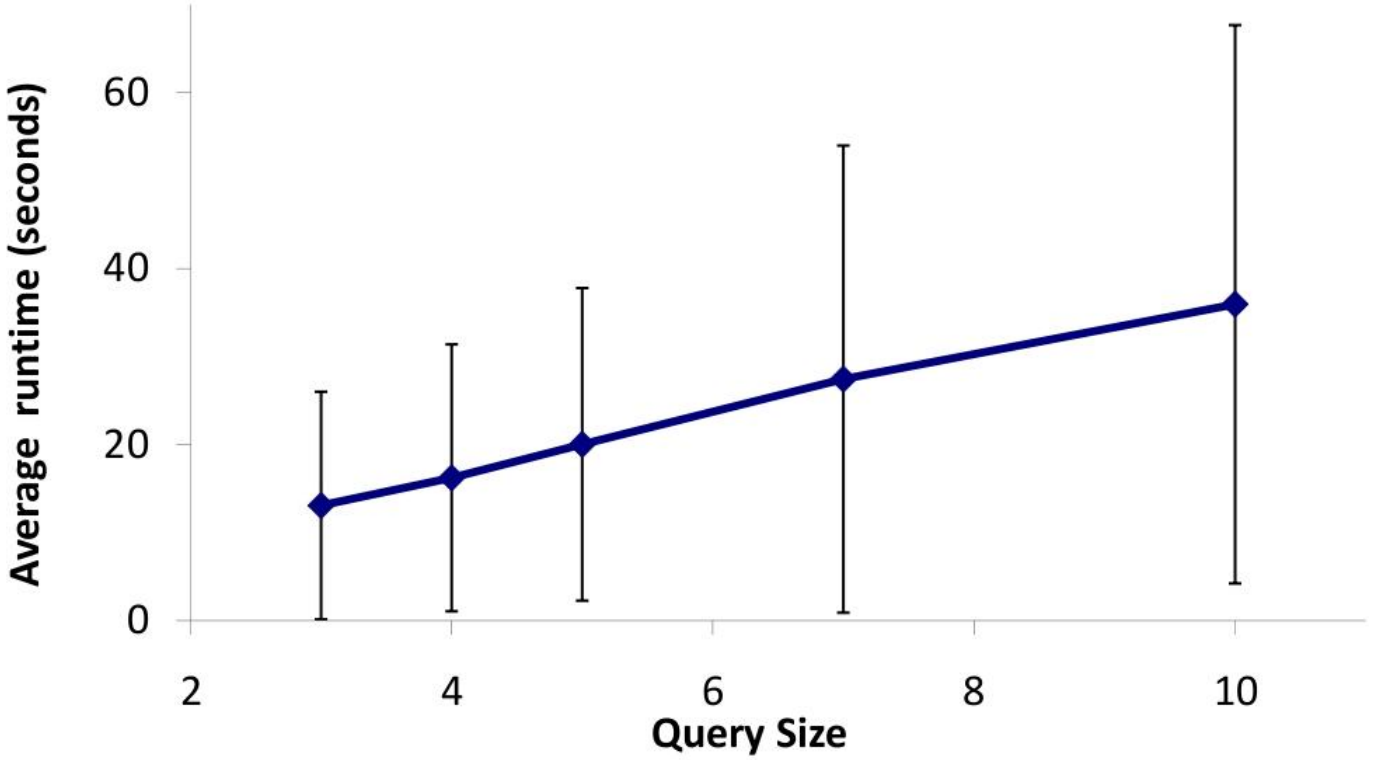


Difficult Queries

- Justify sophisticated consolidation and resolution. So compare with:
  - Processing only the magically known single best list
    => no consolidation/resolution required.
  - Simple consolidation. No merging of approximate duplicates.

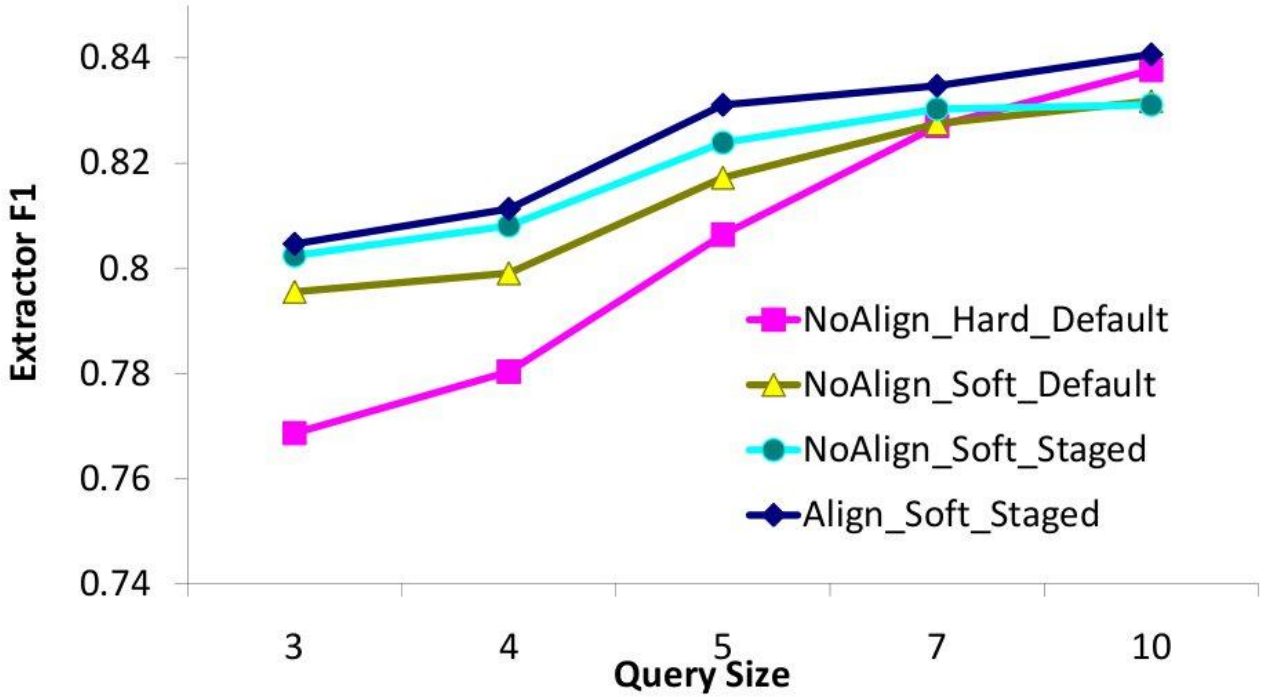
- WWT has > 55% recall, beats others. Gain bigger for difficult queries.

Running time



- < 30 seconds with 3 query records.
  - Can be improved by processing sources in parallel.
- Variance high because time depends on number of columns, record length etc.

Extraction performance



- Benefits of soft training data generation, alignment features, staged-extraction on F1 score.
- Biggest boost from soft training data generation
- Not much boost from staging as we usually have only one stage. Boost significant (+4%) for difficult queries where we have multiple stages.

Related Work

- Google-Squared
  - Developed independently. Launched in May
  - User provides keyword query, e.g. “list of Italian joints in Manhattan”. Schema inferred.
  - Technical details not public.
- Prior methods for extraction and resolution.
  - Assume labeled data/pre-trained parameters
  - We generate labeled data, and automatically train resolver parameters from the list source.

Summary

- Table Augmentation Problem
  - Ad-hoc structured queries on the web
  - Applications in building Wikipedia/Freebase tables.
- WWT system
  - Soft-approach for generating labeled data
  - Exploit content regularity and overlap during extraction using alignments and staged-extraction.
  - Bayesian network for resolution.
  - Need a good ranker. We propose many schemes.

Future Work

- Joint processing of sources
  - Joint training of extractors to fully exploit overlap
  - Joint labeled data generation for consistency across correlated sources.
  - Exploiting ontologies.
- Alternative querying mechanisms:
  - Even three query records might be too much
  - How much can we infer from keyword queries?

(A final WWT table)

Thanks!	Questions?
Merci	Avez-vous des questions?
Grazie	Domande?
Gracias	¿Hay preguntas?
Danke	Haben Sie noch irgenwelche Fragen?

(Note: Please blame the extractor for any wrongly extracted translation)