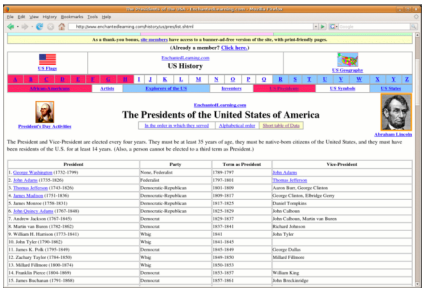
# Lecture 14: Some Topics in Web Search Research




## Administration

- Some changes to the syllabus
  - PA6 goes out today; due March 15
  - PA7 goes out March 15; due March 22
  - Final Project goes out March 17 (as before)
  - FP Proposals due March 24 (earlier!)
  - Midterm #2 on March 29 (later!)
- Upcoming lectures:
  - Topics in Web Search Research (today!)
  - Auctions
  - Recommendation systems
  - Maybe some surprises

2

## Topics in Web Research

- There's a lot going on
  - Text Mining, Search Log Mining
  - Opinion Extraction
  - Recommender Systems
  - Social Search, Mobile Search
  - Search Engine Architecture
  - Developing World Access to the Web
  - All Things Wikipedia
  - Browser Security
- Even XML !!!

3

## Topics in Web Research (2)

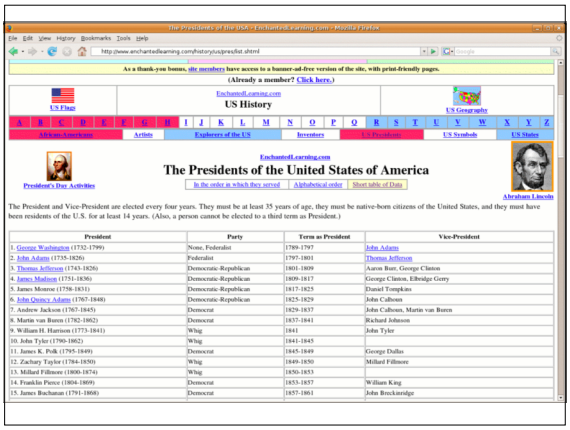
- Rather than talk about 10 things for 3 slides each, we'll discuss a few in depth
- My work deals w/ *information extraction*
  - Getting structured, queryable information from Web text
  - Perhaps slightly more database-centric than a lot of Web work

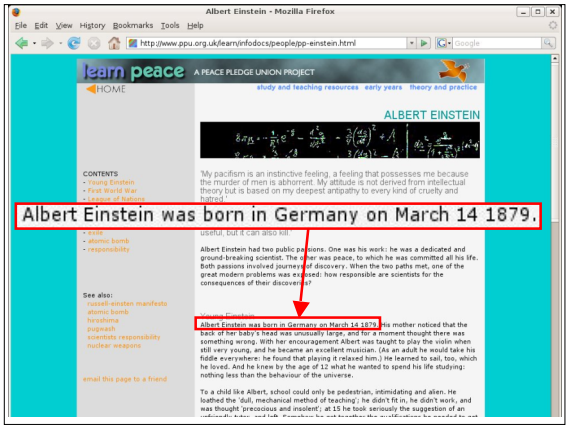
4

## The Modern Web

- Web pages contain structure that is obvious to humans, though not machines
- Search engines are blind to it
  - Search interfaces sort URLs by relevance
  - That's it

5






## What about databases?

- Import interfaces assume data is formally-structured and defined
- Bad assumptions about scale

8

## Vision: The Structured Web

- What if we had an automatically-extracted structured version of all information on the Web?
- *A "Database of Everything"?*

9

## Highly Expressive Web Search

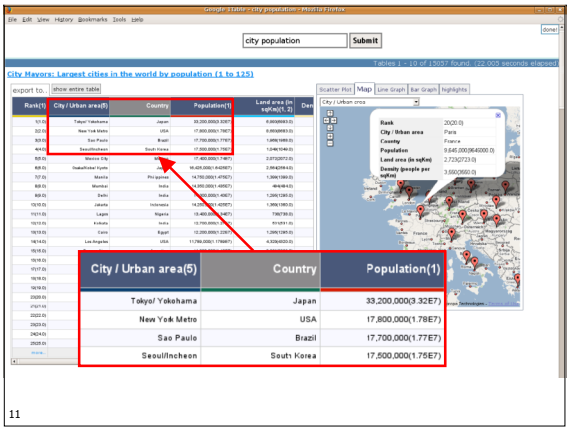
- "List some scientists, their inventions, and their years of birth"
  - `q(?scientist, ?invention, ?dob):-`
    `invented(?scientist, ?invention),`
    `born-in(?scientist, <year>?dob)`

*[CIDR07, "Structured Querying…", Cafarella et al]*

| scientist | invention | dob | prob |
|---|---|---|---|
| Kepler | log tables | 1571 | .8 |
| Heisenberg | matrix mechanics | 1901 | .8 |
| Galileo | telescope | 1564 | .7 |
| Newton | calculus | 1643 | .7 |

- Vision: Combine breadth of search engine with expressiveness of relational database

10




11

## Web Data Integration

- "Compile database of all VLDB PC members"

*[Under Submission, "Data Integration…", Cafarella et al]*

- Vision: Easy reuse of Web's wealth of data (whether or not in XML)

12

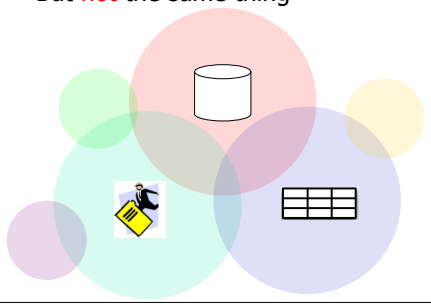
## Outline

- Introduction
- The Structured Web
  - Tables
  - Multiple Tables
  - Text
- Future Work
- Conclusions

13

## A Note

- Structured Web overlaps w/Deep Web…
- But *not* the same thing



14

## Research Methodology

- Information Extraction is venerable field, but doesn't scale to the Web. We need:
  - Domain-independence
  - Scalable computer systems
- Draw on databases, AI, systems

15

## Outline

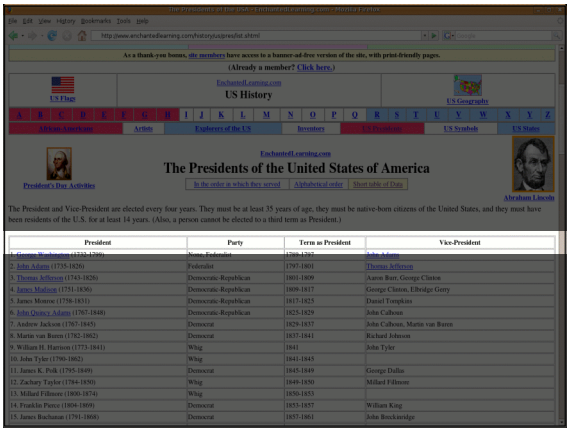
- Introduction
- The Structured Web
  - Tables
  - Multiple Tables
  - Text
- Future Work
- Conclusions

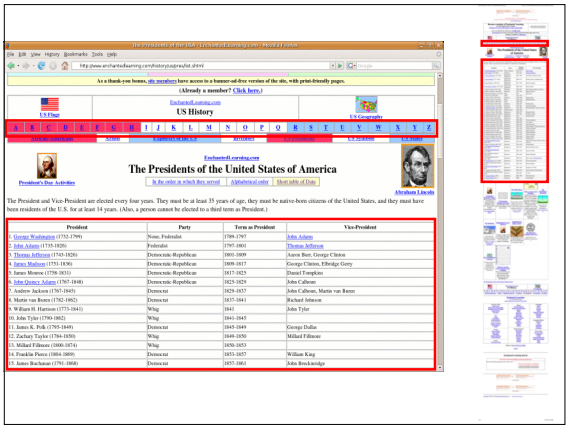
16

## Outline

- Introduction
- The Structured Web
  - Tables
  - Multiple Tables
  - Text
- Future Work
- Conclusions

17



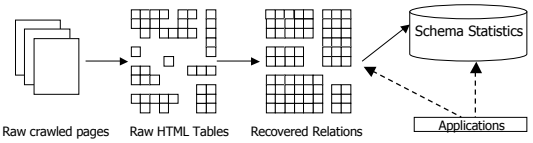


This page contains 16 distinct HTML tables, but only one relational database

Each relational database has its own schema, usually with labeled columns.

## WebTables

- WebTables system automatically extracts dbs from web crawl
  *[WebDB08, "Uncovering…", Cafarella et al]*
  *[VLDB08, "WebTables: Exploring…", Cafarella et al]*




- An extracted relation is one table plus labeled columns
- Estimate that our crawl of 14.1B raw HTML tables contains ~154M good relational dbs

21

## Relation Recovery

**Step 1. Relational Filtering**
Recall 81%, Precision 41%

**Step 2. Metadata Detection**
Recall 85%, Precision 89%

- Output
  - 271M databases, about 125M are good
  - Five orders of magnitude larger than previous largest corpus *[WWW02, "A Machine Learning…", Wang & Hu]*
  - 2.6M unique relational schemas
- What can we do with those schemas?
  *[VLDB08, "WebTables: Exploring…", Cafarella et al]*

22

## Schema Statistics

Recovered Relations

| make | model | year |
|---|---|---|
| Toyota | Camry | 1984 |

| make | model | year |
|---|---|---|
| Mazda | Protégé | 2003 |
| Chevrolet | Impala | 1979 |

| make | model | year | color |
|---|---|---|---|
| Chrysler | Volare | 1974 | yellow |
| Nissan | Sentra | 1994 | red |

| name | addr | city | state | zip |
|---|---|---|---|---|
| Dan S | 16 Oak St. | Seattle | WA | 98195 |
| Alon H | 129 Elm | Belmont | CA | 94011 |

| name | last-modified |
|---|---|
| Readme.txt | 182 | Apr 26, 2005 |
| cac.xml | 813 | Jul 23, 2008 |

| Schema | Freq |
|---|---|
| {make, model, year} | 2 |
| {make, model, year, color} | 1 |
| {name, addr, city, state, zip} | 1 |
| {name, size, last-modified} | 1 |

- Schema stats useful for computing attribute probabilities
  - p("make"), p("model"), p("zipcode")
  - p("make" | "model"), p("make" | "zipcode")

23

## App #1: Schema Autocomplete

- Input: topic attribute (*e.g.*, `make`)
- Output: relevant schema
  `{make, model, year, price}`
  - "tab-complete" for your database

- For input set *I*, output *S*, threshold *t*
  - while $p(S\text{-}I \mid I) > t$
    - *newAttr* = max $p(newAttr, S\text{-}I \mid I)$
    - $S = S \cup newAttr$
    - emit *newAttr*

24

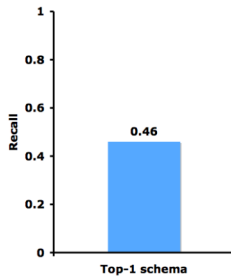
## App #1: Schema Autocomplete

| | |
|---|---|
| name | name, size, last-modified, type |
| instructor | instructor, time, title, days, room, course |
| elected | elected, party, district, incumbent, status, … |
| ab | ab, h, r, bb, so, rbi, avg, lob, hr, pos, batters |
| sqft | sqft, price, baths, beds, year, type, lot-sqft, … |

25

## App #1: Experiments

- Asked experts for schemas in 10 areas
- What was autocompleter's recall?




26

## App #2: Synonym Discovery

- Input: topic attribute (*e.g.*, `address`)
- Output: relevant synonym pairs (`telephone` = `tel-#`)
  - Used for schema matching *[VLDB01, "Generic Schema Matching…", Madhavan et al]*
  - Linguistic thesauri are incomplete; hand-made thesauri are burdensome
- For attributes *a*, *b* and input domain *C*, when *p(a,b)=0*

$$syn(a,b) = \frac{p(a)p(b)}{\epsilon + \Sigma_{z \in A} (p(z|a,C) - p(z|b,C))^2}$$

27

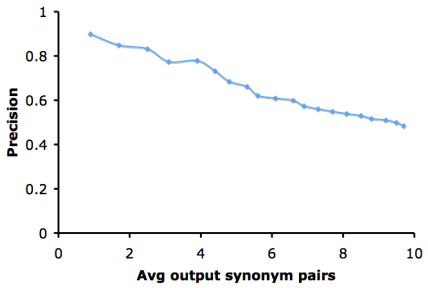
## App #2: Synonym Discovery

| | |
|---|---|
| name | e-mail\|email, phone\|telephone, e-mail_address\|email_address, date\|last_modified |
| instructor | course-title\|title, day\|days, course\|course-#, course-name\|course-title |
| elected | candidate\|name, presiding-officer\|speaker |
| ab | k\|so, h\|hits, avg\|ba, name\|player |
| sqft | bath\|baths, list\|list-price, bed\|beds, price\|rent |

28

## App #2: Experiments

- For each input attr, repeatedly emit best synonym pair (until min threshold reached)




29

## WebTables Contributions

- Largest collection of databases and schemas, by far
- Large-scale extracted schema data for first time; enables novel applications

30

## Outline

- Introduction
- The Structured Web
  - Tables
  - Multiple Tables
  - Text
- Future Work
- Conclusions

31

## Multiple Tables

- Can we combine tables to create new data sources?
- Data integration for the Structured Web
- Many existing "mashup" tools, which ignore realities of Web data
  - A lot of useful data is not in XML
  - User cannot know all sources in advance
  - Transient integrations

32

## Integration Challenge

- Try to create a database of all `"VLDB program committee members"`


33

## Octopus

- Provides "workbench" of data integration operators to build target database
  - Most operators are not correct/incorrect, but high/low quality (like search)
  - Also, prosaic traditional operators

34

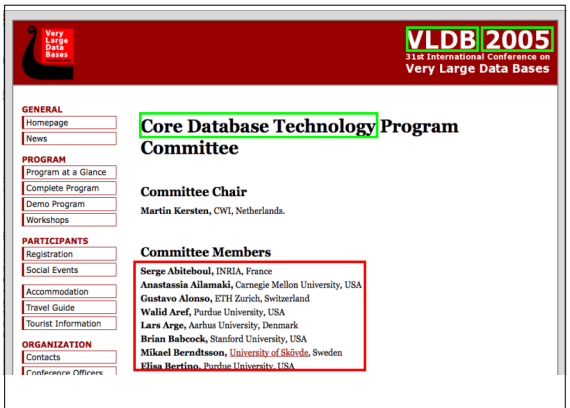
## Walkthrough - Operator #1

- **SEARCH**(`"VLDB program committee members"`)


| | |
|---|---|
| serge abiteboul | inria |
| michael adiba | …grenoble |
| antonio albano | …pisa |
| … | … |


| | |
|---|---|
| serge abiteboul | inria |
| anastassia ail… | carnegie… |
| gustavo alonso | etz zurich |
| … | … |

35




36

## Walkthrough - Operator #2

- Recover relevant data

**CONTEXT()**

| | | |
|---|---|---|
| serge abiteboul | inria | 1996 |
| michael adiba | …grenoble | 1996 |
| antonio albano | …pisa | 1996 |
| … | … | … |

**CONTEXT()**

| | | |
|---|---|---|
| serge abiteboul | inria | 2005 |
| anastassia ail… | carnegie… | 2005 |
| gustavo alonso | etz zurich | 2005 |
| … | … | … |

37

## Walkthrough - Union

- Combine datasets

Union()

| | | |
|---|---|---|
| serge abiteboul | inria | 1996 |
| michael adiba | …grenoble | 1996 |
| antonio albano | …pisa | 1996 |
| serge abiteboul | inria | 2005 |
| anastassia ail… | carnegie… | 2005 |
| gustavo alonso | etz zurich | 2005 |
| … | … | … |

38

## Walkthrough - Operator #3

- Add column to data
- Similar to "join" but join target is a topic

**EXTEND**( "publications", col=0)

"publications"

| | | | | | |
|---|---|---|---|---|---|
| serge abiteboul | inria | serge abiteboul | inria | P2P Dist… | 1996 |
| michael adiba | …grenoble | michael adiba | …grenoble | Exploiting… | 1996 |
| antonio albano | …pisa | antonio albano | …pisa | Another Exam… | 1996 |
| serge abiteboul | inria | serge abiteboul | inria | P2P Dist… | 2005 |
| anastassia ail… | carnegie… | anastassia ail… | carnegie… | Efficient… | 2005 |
| gustavo alonso | etz zurich | gustavo alonso | etz zurich | Web Exhibi… | 2005 |
| … | … | … | … | | … |

- User has integrated data sources with little effort
- No wrappers; data was never intended for reuse

39

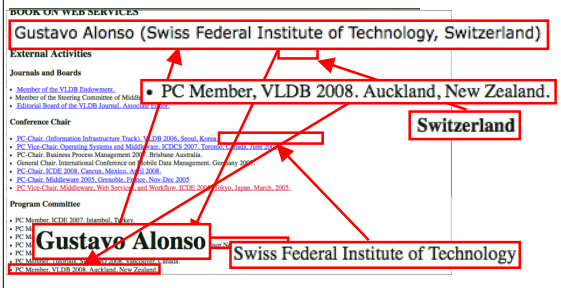
## CONTEXT Algorithms

- Input: table and source page
- Output: data values to add to table
- *SignificantTerms* sorts terms in source page by "importance" (tf-idf)

40

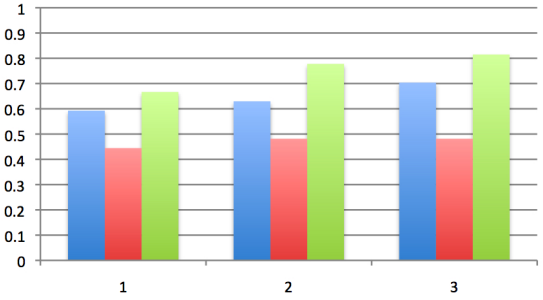
## Related View Partners

- Looks for different "views" of same data




41

## CONTEXT Experiments




42

## Octopus Contributions

- Basic operators that enable Web data integration with very small user burden
- Realistic and useful implementations for all three operators

43

## Web Research Topics

- The Structured Web exists in raw form today, but tools largely ignore it
- Traditional databases and Web search not appopriate
- I've described a few useful techniques
- Lots of other interesting Web research problems

44

## Extra Material

- For the dedicated reader!

45

## Outline

- Introduction
- The Structured Web
  - Tables
  - Multiple Tables
  - Text
- Future Work
- Conclusions

46

## Text

- "Einstein was born in Germany" ⇒
  `(einstein, was-born-in, germany)`

47

## Wrapper Induction

*[IJCAI97, "Wrapper Induction for…", Kushmerick, Weld, Doorenbos]*
*[AutAgent01, "Hierarchical Wrapper Induction…", Muslea, Minton, Knoblock]*


```
<h1 class="page-title">[RESTAURANT NAME]</h1>
<span class="phone">[Phone-Number]</span>
<a ref="…">[Neighborhood]</a>
```

- Site-or-domain-specific training data
- Only for "semistructured" sources, *e.g.*, Amazon

48

## Extraction Rule Learning

*[WebDB98, "Extracting Patterns and Relations…", Brin]*
*[SIGMOD01, "Snowball: A Prototype System…", Agichtein et al]*

**"Servers at Microsoft's headquarters in Redmond…"**
**"The Armonk-based IBM has introduced…"**
**"Intel, Santa Clara, cut prices of its Pentium…"**

| | |
|---|---|
| Microsoft | Redmond |
| IBM | Armonk |
| Intel | Santa Clara |

[ORGANIZATION]'s headquarters in [LOCATION]
[LOCATION]-based [ORGANIZATION]
[ORGANIZATION], [LOCATION]

- Again, requires domain-specific training data

49

## System Comparison

| | Domain independence | Comp. complexity |
|---|---|---|
| Wrapper & Extraction-Rule Learning *[various]* | | |
| WebKB *[AIJ00, "Learning to Construct…", Craven et al]* | | |
| KnowItAll *[AAAI04, "Methods for Domain…", Etzioni et al]* | | |
| Shinyama & Sekine '06 *[HLT-NAACL06, "Preemptive…", Shinyama & Sekine]* | | |
| | | |

50

## System Comparison

| | Domain independence | Comp. complexity |
|---|---|---|
| Wrapper & Extraction-Rule Learning *[various]* | Domain-specific training data | |
| WebKB *[AIJ00, "Learning to Construct…", Craven et al]* | Domain-specific training data | |
| KnowItAll *[AAAI04, "Methods for Domain…", Etzioni et al]* | Relations must be given in advance | |
| Shinyama & Sekine '06 *[HLT-NAACL06, "Preemptive…", Shinyama & Sekine]* | Relations discovered automatically | |
| | | |

51

## System Comparison

| | Domain independence | Comp. complexity |
|---|---|---|
| Wrapper & Extraction-Rule Learning *[various]* | Domain-specific training data | various |
| WebKB *[AIJ00, "Learning to Construct…", Craven et al]* | Domain-specific training data | O(D * R) |
| KnowItAll *[AAAI04, "Methods for Domain…", Etzioni et al]* | Relations must be given in advance | O(D * R) |
| Shinyama & Sekine '06 *[HLT-NAACL06, "Preemptive…", Shinyama & Sekine]* | Relations discovered automatically | $O(D^2)$ |
| | | |

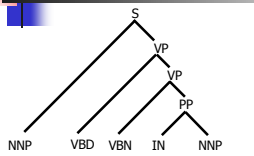
52

## System Comparison

| | Domain independence | Comp. complexity |
|---|---|---|
| Wrapper & Extraction-Rule Learning *[various]* | Domain-specific training data | various |
| WebKB *[AIJ00, "Learning to Construct…", Craven et al]* | Domain-specific training data | O(D * R) |
| KnowItAll *[AAAI04, "Methods for Domain…", Etzioni et al]* | Relations must be given in advance | O(D * R) |
| Shinyama & Sekine '06 *[HLT-NAACL06, "Preemptive…", Shinyama & Sekine]* | Relations discovered automatically | $O(D^2)$ |
| **TextRunner** *[IJCAI07, "Open Information…", Banko, Cafarella, et al]* | Relations discovered automatically | O(D) |

53

## Naïve TextRunner




Einstein was born in Germany.

**Step 1. Parse each sentence**

| | |
|---|---|
| $E_1$ Verb $E_2$ | *X established Y* |
| $E_1$ NP Prep $E_2$ | *X professor of Y* |
| $E_1$ Verb Prep $E_2$ | *X moved to Y* |

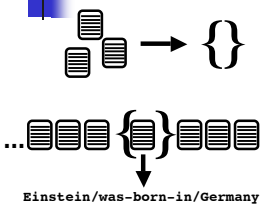
**Einstein | was-born-in | Germany**

**Step 2. Apply extraction patterns**

- Domain-independent
- No training data or extraction rules needed (the corpus itself is the only input)
- Complexity O(D)
- Extremely computationally expensive
- Can be brittle on Web text

54

## TextRunner Architecture



1. With parser-derived training data, build efficient extractor using lightweight features

2. Run efficient extractor over corpus

Einstein/was-born-in/Germany

| Extraction | # |
|---|---|
| Einstein/was-born-in/Germany | 97 |
| Tesla/invented/coil-transformer | 54 |
| demands/of-securing/border | 2 |
| And-McCain/saw/chance | 1 |

3. Redundancy-based assessment
*[IJCAI05, "A Probabilistic Model…", Downey, Etzioni, Soderland]*

55

## Output Quality

- Test corpus of 9M pages
  - 133M sentences
  - 11.3M unique tuples
- On 10 randomly-selected relations…

| | Avg error rate | # correct extractions |
|---|---|---|
| KnowItAll | 18% | 11,631 |
| TextRunner | | |

56

## Output Quality

- Test corpus of 9M pages
  - 133M sentences
  - 11.3M unique tuples
- On 10 randomly-selected relations…

| | Avg error rate | # correct extractions |
|---|---|---|
| KnowItAll | 18% | 11,631 |
| TextRunner | 12% | 11,476 |

57

## Output Quality

- Test corpus of 9M pages
  - 133M sentences
  - 11.3M unique tuples
- On 10 randomly-selected relations…

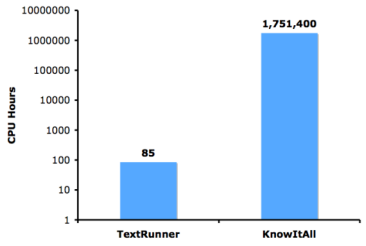
| | Avg error rate | # correct extractions |
|---|---|---|
| KnowItAll | 18% | 11,631 |
| TextRunner | 12% | 11,476 |

- Overall, 79% extractions are "well-formed", 80% accurate

58

## Runtime Performance

- KnowItAll O(DR): 6.3 CPU hours per relation
- TextRunner O(D): 85 CPU hours for corpus
- Corpus has 278k distinct relation strings




59

## TextRunner Contributions

- TextRunner scales textual information extraction to the entire Web
  - Domain independent (in particular, no relations in advance)
  - Reduces runtime to O(D)

60

## Future Work

- WebTables
  - Schema autocomplete & synonyms just few of many possible *semantic services*
    - Input: schema; Output: tuples
      *database autopopulate*
    - Input: tuples; Output: schema
      *schema autogenerate*
- Octopus
  - Index support for interactive speeds

61

## Future Work (2)

- "The Database of Everything"
  *[CIDR09, "Extracting and Querying…", Cafarella]*
  - Is domain-independence enough?

| Text-embedded |
|---|
| be/is |
| ask/call |
| arrive/come/go |
| join/lead |
| born-in |

| Table-embedded |
|---|
| <web access log> |
| <file listing> |
| <forum posts> |
| <album listing> |
| <phone numbers> |

  - Multi-model, multi-extractor approach probable
  - Vast deduplication challenge
  - New tools needed for user/extractor interaction

62