II. Future of Search

Hints of the future in:

- Mulder Question-answering (UW, 2001)
  - Who killed JFK?
  - Elvis!
- Flipdog --- jobs.
- Froogle --- products.

Information Extraction

What is Information Extraction (IE)

- Is the task of populating database slots with corresponding phrases from text

Previous Approaches

- Supervised Learning
  - Hidden Markov models
  - Conditional random fields
  - Performs pretty well, but
  - Requires LOTs
- Challenge: Autonomous!
  - Unsupervised approach
Web Search “Paradigm Shift”

Could information extraction be the basis for a new kind of search engine?

I. Architecture
II. Lessons Learned
III. Evaluation
IV. Improving Recall
V. Future Directions

I. Generate-n-Test Architecture

**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”

Test

Assess candidate extractions using Mutual Information (PMI-IR) (Turney ’01).

\[
PMI(Seattle, City) = \frac{|Hits(Seattle + City)|}{|Hits(Seattle)|}
\]

Many variations are possible…
Assessment

\[
PMI (I, D) = \frac{|Hits (I + D)|}{|Hits (I)|}
\]

- \(PMI\) = frequency of I & D co-occurrence
- 5-50 discriminators \(D_i\)
- Each \(PMI\) for \(D_i\) is a feature \(f_i\)
- Naïve Bayes evidence combination:

\[
P(\phi \mid f_1, f_2, ..., f_n) = \frac{P(\phi) \prod_{i=1}^{n} P(f_i \mid \phi)}{P(\phi) \prod_{i=1}^{n} P(f_i \mid \phi) + P(\neg \phi) \prod_{i=1}^{n} P(f_i \mid \neg \phi)}
\]

Assessment In Action

1. I = “Yakima” (1,340,000)
2. D = <class name>
3. I+D = “Yakima city” (2760)
4. PMI = \((2760 / 1.34M) = 0.02\)
   - I = “Avocado” (1,000,000)
   - I+D = “Avocado city” (10)
     \(PMI = 0.00001 < 0.02\)

Contrast with Related Work

- **Unsupervised** learning.
  - Contrast with HMM, CRF, etc.
- Open-ended, scalable extraction.
  - Contrast with small collections, narrow domain.

II. Questions/Lessons

1. What about ambiguity?
2. Can we scale over time?
3. Can we extract “correct” facts?
4. Can we be comprehensive?

Some Sources of ambiguity

- **Time**: “Clinton is the president” (in 1996).
- **Context**: “common misconceptions..”
- **Opinion**: Elvis…
- **Multiple word senses**: Amazon, Chicago, Chevy Chase, etc.
  - Dominant senses can mask recessive ones!
  - Approach: unmasking. ‘Chicago –City’

Chicago

\[
PMI (I, D, C) = \frac{|Hits (I + D \mid C)|}{|Hits (I \mid C)|}
\]
Chicago Unmasked

Impact of Unmasking on PMI

<table>
<thead>
<tr>
<th>Name</th>
<th>Recessive</th>
<th>Original</th>
<th>Unmask</th>
<th>Boost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Washington</td>
<td>city</td>
<td>0.50</td>
<td>0.99</td>
<td>96%</td>
</tr>
<tr>
<td>Casablanca</td>
<td>city</td>
<td>0.41</td>
<td>0.93</td>
<td>127%</td>
</tr>
<tr>
<td>Chevy Chase</td>
<td>actor</td>
<td>0.09</td>
<td>0.58</td>
<td>512%</td>
</tr>
<tr>
<td>Chicago</td>
<td>movie</td>
<td>0.02</td>
<td>0.21</td>
<td>972%</td>
</tr>
</tbody>
</table>

2. Can we scale over time?

Performance
(New, correct facts per Web page)

III. Metrics for Evaluation

- Precision = |correct results| / |results|
- Recall = |results| / |true answer set|
  - Web recall = |correct results|
- Ranked results → Recall/precision tradeoff
  - Best result → high precision, but low recall
  - All results → low precision, but high recall
IV. Methods for Improving Recall
( AAAI '04)

• RL: learn class-specific patterns.
  “Headquarted in <city>”

• SE: Recursively extract subclasses.
  “Scientists such as physicists, chemists, and biologists…”

• LE: extract lists of items (~ Google Sets).

List Extraction (LE)

1. Query Engine with known items.
2. Learn a wrapper for each result page.
3. Collect large number of lists.
4. Sort items by number of list “votes”.

LE+A = sort list according to Assessor.

Evaluation: Web recall, at precision= 0.9.

Results for City

Found 10,300 cities missing from Tipster Gazetteer.

Results for Film

Results for Scientist
Bindings Engine (BE, WWW ‘05)

- Allow queries containing typed variables
  - “cities such as <ProperNoun>”
- KnowItNow: an omnivore!
- Limited extraction at interactive speeds.
  - KnowItNow

V. Research Directions

- Binary Relations
  - Binary Extraction Rules
  - Hidden Markov Models
- OPINE
  - Can KnowItAll learn to read?
  - Learn 100,000,000 (interesting!) facts
    - separate facts from opinions
    - Focus of attention
  - KnowItAll
  - http://www.cs.washington.edu/research/knowitall

Binary Extraction patterns

\[
R(C_1, C_2) \leftrightarrow C_1 \text{“,” } R \text{“of” } C_2
\]

Instantiated Patterns:
- Ceo(Person, Company) \rightarrow <person> “, CEO of” <company>
- Star(Actor, Film) \rightarrow <actor> “, star of” <film>
- Ceo(Person, Microsoft) \rightarrow <person> “, CEO of Microsoft”

Assessment for Binary Predicates

Extraction: \text{ceo(“Jeff Bezos”, “Amazon”)}

Discriminator: <arg1> CEO of <arg2>

\[
PMI(D, I_1, I_2) = \frac{|hits(D + I_1 + I_2)|}{|hits(I_1, I_2)|}
\]

670 hits for “Jeff Bezos CEO of Amazon”
39,000 hits for “Jeff Bezos”, “Amazon”
PMI = 0.017