Record-Boundary Discovery in Web Documents

D. W. Embley
Y. Jiang
Y. -K. Ng

Brigham Young University

“Ontology-based Extraction and Structuring of Information from Data-rich Unstructured Documents”

Outline

• Introduction
• Record-Boundary Discovery
• Individual Heuristics
• Combined Heuristic
• Experiment
• Extraction and Structuring
• Conclusion
Introduction

- Extract and structure the web data
  - develop an ontological model instance for a domain of interest (Application Ontology)
  - parse this ontology (Ontology Parser)
    - generate the rules for matching constants and keywords
    - generate a database scheme
  - separate a web page into individual record-size chunks (Record Extractor)
  - extract objects and relationships (Recognizer)
  - populate the database instance (Generator)

Framework
Introduction

- Assumptions
  - a web page has multiple records (data-rich)
    - car advertisements, job listing, obituaries
  - a web page contains at least one record-separator tag
    - discover boundaries of records

Sample web page

```html
<html><head><title>Classification</title></head>
<body background="#FFFFF1">
<h1>Funeral Notice - November 1, 1998</h1>
<p>Leonard Adams died on September 30, 1998. Leonard was born on September 5, 1931.
</p>
</body></html>
```
Record-Boundary Discovery

- **Tag tree**
  - represent the nested structure of a web page
  - a node ≡ a region in the web page

- **Locating groups of records**
  - choose the subtree whose root has the highest fan-out
  - count the number of appearances for each tag in the immediate child nodes
    - irrelevant tag (≤ 10%): `<h1>`
    - candidate tag: `<hr>`, `<b>`, `<br>`

- **Ranking the candidate tags**
  - individual heuristics
  - combined heuristics
Individual Heuristics

- **Highest-count Tags: HT**
  - sort by the number of appearances in the highest fan-out subtree
    - b:8 → br:5 → hr:4

- **Identifiable “separator” Tags: IT**
  - ordered list for common use by observation
    - hr tr td a table p br h4 h1 strong b i
    - 75 15 15 14 10 10 10 10 10 10 6 5 (%)

- **Standard Deviation: SD**
  - compute the standard deviation of the interval between each occurrence of a tag
    - hr:0.57 → b:0.89 → br:1.25

- **Repeating-tag Pattern: RP**
  - count the number of occurrences for all pairs of candidate tags without intervening text
  - calculate the differences between counts
    - $x_{xy} = |c_{xy} - c_x|$, $y_{xy} = |c_{xy} - c_y|$,
    - $x \rightarrow y$ if $\min(x_{ai}) < \min(y_{aj})$ for all possible $i, j$
    - hr:1 → br:2 → b:5 [$c_{br,hr}=3$, $c_{hr,b}=3$]

- **Ontology Matching: OM**
  - estimate the number of records by applying the record-identifying fields
    - Funeral:4, Birth Date: 2, Death Date: 3 → 3
    - hr:1 → br:2 → b:5
Combined Heuristic

- Certainty measure
  - Define a confidence measure by using Stanford certainty theory
    - Two evidences E₁ and E₂ come from the same observation B
    - CF(E₁) and CF(E₂) are certainty factors (CF)
      - Compound CF = CF(E₁) + CF(E₂) - CF(E₁) × CF(E₂)
  - Example
    - HT \(\Rightarrow 88\%\), IT \(\Rightarrow 74\%\), SD \(\Rightarrow 66\%\)
      - \(88\% + 74\% + 66\% - 88\% \times 74\% - 74\% \times 66\% - 66\% \times 88\% + 88\% \times 74\% \times 66\% = 98.93\%\)

Combined Heuristic

- CF of individual heuristics [training]

<table>
<thead>
<tr>
<th>Heuristic \ Ranking</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obituaries</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OM</td>
<td>83%</td>
<td>17%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>RP</td>
<td>83%</td>
<td>7%</td>
<td>10%</td>
<td>0%</td>
</tr>
<tr>
<td>SD</td>
<td>59%</td>
<td>27%</td>
<td>14%</td>
<td>0%</td>
</tr>
<tr>
<td>IT</td>
<td>92%</td>
<td>8%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>HT</td>
<td>58%</td>
<td>23%</td>
<td>17%</td>
<td>2%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Heuristic \ Ranking</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Car advertisements</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OM</td>
<td>86%</td>
<td>8%</td>
<td>4%</td>
<td>2%</td>
</tr>
<tr>
<td>RP</td>
<td>72%</td>
<td>18%</td>
<td>8%</td>
<td>2%</td>
</tr>
<tr>
<td>SD</td>
<td>72%</td>
<td>18%</td>
<td>10%</td>
<td>0%</td>
</tr>
<tr>
<td>IT</td>
<td>100%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>HT</td>
<td>40%</td>
<td>42%</td>
<td>16%</td>
<td>2%</td>
</tr>
</tbody>
</table>

Selected CF (average)
### Combined Heuristic

#### Compound heuristic
- apply individual heuristics to ranking
  - ranks of `<hr>`: HT ⇒ 3, IT ⇒ 1, SD ⇒ 1
  - $2\% + 96\% + 65.5\% - \ldots + 2\% \times 96\% \times 65.5\%$

#### Algorithm
- construct a tag tree
- locate the highest fan-out subtree
- extract the candidate tags
- apply individual heuristics
- apply Stanford certainty theory
- choose the tag with the highest compound CF

---

### Combined Heuristic

#### Combinations of five heuristics [training]
- 26 choices
- success rate

<table>
<thead>
<tr>
<th>Compound Heuristic</th>
<th>Success Rate</th>
<th>Compound Heuristic</th>
<th>Success Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>OR</td>
<td>85.83%</td>
<td>OR</td>
<td>95.00%</td>
</tr>
<tr>
<td>OS</td>
<td>88.00%</td>
<td>OSH</td>
<td>87.50%</td>
</tr>
<tr>
<td>OI</td>
<td>95.00%</td>
<td>OIH</td>
<td>95.00%</td>
</tr>
<tr>
<td>OH</td>
<td>79.00%</td>
<td>RSI</td>
<td>95.00%</td>
</tr>
<tr>
<td>RS</td>
<td>79.50%</td>
<td>RSH</td>
<td>85.50%</td>
</tr>
<tr>
<td>RI</td>
<td>95.00%</td>
<td>RIH</td>
<td>95.00%</td>
</tr>
<tr>
<td>RH</td>
<td>76.33%</td>
<td>SIH</td>
<td>95.00%</td>
</tr>
<tr>
<td>SI</td>
<td>95.00%</td>
<td>ORSI</td>
<td>100.00%</td>
</tr>
<tr>
<td>SH</td>
<td>69.50%</td>
<td>ORSH</td>
<td>82.50%</td>
</tr>
<tr>
<td>IH</td>
<td>95.00%</td>
<td>ORIH</td>
<td>100.00%</td>
</tr>
<tr>
<td>ORS</td>
<td>81.50%</td>
<td>ORS</td>
<td>95.00%</td>
</tr>
<tr>
<td>ORI</td>
<td>93.33%</td>
<td>RSIH</td>
<td>100.00%</td>
</tr>
<tr>
<td>ORH</td>
<td>84.83%</td>
<td>ORSIIH</td>
<td>100.00%</td>
</tr>
</tbody>
</table>
**Combined Heuristic**

- **Example**
  - individual heuristics
    - HT: [(b,1), (br,2), (hr,3)]
    - IT: [(hr,1), (br,2), (b,3)]
    - SD: [(hr,1), (b,2), (br,3)]
    - RP: [(hr,1), (br,2), (b,3)]
    - OM: [(hr,1), (br,2), (b,3)]
  - compound heuristic
    - ORSIH: [(hr,99.96%), (br,64.75%), (b,56.34%)]

**Experiment**

- **Test results**
  - 4 test sets
  - 20 web pages

![Table with rank numbers of correct tag]
Experiment

- Success rates

<table>
<thead>
<tr>
<th>Heuristic</th>
<th>Success Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>OM</td>
<td>80%</td>
</tr>
<tr>
<td>RP</td>
<td>75%</td>
</tr>
<tr>
<td>SD</td>
<td>65%</td>
</tr>
<tr>
<td>IT</td>
<td>95%</td>
</tr>
<tr>
<td>HT</td>
<td>45%</td>
</tr>
<tr>
<td>ORSIH</td>
<td>100%</td>
</tr>
</tbody>
</table>

Extraction and Structuring

- Main processes
  - ontology parser
    - constant/keyword matching rules
      - a list of regular expressions for each object-set
  - SQL create-table statements
    - object-set name $\Rightarrow$ attribute name
  - constant/keyword recognizer
    - name/string/position table
  - database-instance generator
    - tuples with a sequence of (attribute,value) pairs
Extraction and Structuring

- Ontology for car advertisements
  - graphical
  - textual

```
Car [0:1] has Year [1:*]
```

```
Year (regexp(2): \d(2):\b\d(2)\b )
```

Extraction and Structuring

```
Car [0:1] has Year [1:*]
Car has Make [1:*]
Year has Mileage [0:1]
Year has Price [0:1]
Model has Feature [0:1]
Make has PhoneNr [0:1]
Model has Mileage [0:1]
Year has Mileage [0:1]
Price has Mileage [0:1]
Feature has Mileage [0:1]
Feature has Year [1:*]
```

participation constraint

object-set
Extraction and Structuring

- **Heuristics for selection of constants**
  - keyword proximity: one-to-many
    - (Mileage, 15000)
  - functional relationship: one-to-one
    - (Model, Carlo), (PhoneNr, 298-8090)
  - nonfunctional relationship: many-to-many
    - (Feature, Red)
  - first occurrence without constraint violation
    - (Year, 96), (Make, CHEV), (Price, 14990)

Extraction and Structuring

- **Database instances**
  - tuples
    - Car (1001, “96”, “CHEV”, “Monte Carlo”, “15000”, “14990”, “298-8090”)
    - Car-Feature(“1001”, ”Red”)
  - table

<table>
<thead>
<tr>
<th>Year</th>
<th>Make</th>
<th>Model</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>94</td>
<td>DODGE</td>
<td>Intrepid</td>
<td>10,000</td>
</tr>
<tr>
<td>94</td>
<td>DODGE</td>
<td>Taurus</td>
<td>3,500</td>
</tr>
<tr>
<td>91</td>
<td>FORD</td>
<td>Probe</td>
<td>4,995</td>
</tr>
<tr>
<td>90</td>
<td>FORD</td>
<td>Escort</td>
<td>1000</td>
</tr>
</tbody>
</table>
Extraction and Structuring

- **Experiments**
  - $N$: number of facts in the source
  - $C$: number of facts declared correctly
  - $I$: number of facts declared incorrectly

<table>
<thead>
<tr>
<th></th>
<th>$N$</th>
<th>$C$</th>
<th>$I$</th>
<th>$\frac{C}{N}$</th>
<th>$\frac{C}{C+I}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Car</td>
<td>116</td>
<td>116</td>
<td>0</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>Year</td>
<td>116</td>
<td>116</td>
<td>0</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>Make</td>
<td>114</td>
<td>93</td>
<td>0</td>
<td>0.82</td>
<td>1.00</td>
</tr>
<tr>
<td>Model</td>
<td>114</td>
<td>95</td>
<td>0</td>
<td>0.82</td>
<td>1.00</td>
</tr>
<tr>
<td>Mileage</td>
<td>31</td>
<td>28</td>
<td>0</td>
<td>0.90</td>
<td>1.00</td>
</tr>
<tr>
<td>Price</td>
<td>103</td>
<td>103</td>
<td>0</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>PhoneNo</td>
<td>116</td>
<td>109</td>
<td>0</td>
<td>0.94</td>
<td>1.00</td>
</tr>
<tr>
<td>Extension</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0.50</td>
<td>1.00</td>
</tr>
<tr>
<td>Feature</td>
<td>289</td>
<td>264</td>
<td>1</td>
<td>0.91</td>
<td>0.996</td>
</tr>
<tr>
<td>All Attributes</td>
<td>1003</td>
<td>943</td>
<td>1</td>
<td>0.94</td>
<td>0.9989</td>
</tr>
</tbody>
</table>

Conclusion

- **Contributions**
  - an ontology-based framework for extracting and structuring information in web pages
  - a heuristic approach to discovering record boundaries in web pages
  - a heuristic approach to recognizing facts contained in web pages

- **Difficulties**
  - misidentification of attributes: I-15566-2441
  - variations in patterns: Wind95 ⇒ Win95
  - typographical mistakes: Chrystler ⇒ Chrysler
Conclusion

- Issues
  - make the ontological descriptions richer
    - generalization/specialization hierarchies, aggregation, n-ary relationship sets, …
  - improve schema generation and database population
    - better data type: 55000 vs. 55k