Unsupervised Wrapper Induction using Linked Data

Anna Lisa Gentile  Z. Zhang  I. Augenstein  F. Ciravegna

Department of Computer Science, The University of Sheffield, UK

KCAP 2015, 8th October 2015
Outline

1. Wrapper Induction: Task definition
2. Proposed Methodology
3. Dataset
4. Experiment
5. Conclusions
Wrapper Induction: definition of the task

- Automatically learning wrappers using a collection of manually annotated Web pages as training data
  [Kushmerick, 1997, Muslea et al., 2003, Dalvi et al., 2009, Dalvi et al., 2011, Wong and Lam, 2010]

- Data is generally extracted from “detail” Web pages
  [Carlson and Schafer, 2008]
    - pages corresponding to a single data record (or entity) of a certain type or *concept* (also called *vertical* in the literature)
    - render various attributes of each record in a human-readable form
Wrapper Induction: Task definition

Proposed Methodology

Dataset

Experiment

Conclusions

Wrapper Induction: example

Extracting book attributes on e-commerce websites
Web Scale Wrapper Induction

- **Traditional wrapper induction task**
  - schema
  - set of pages output from a single script
  - training data are given as input, and a wrapper is inferred that recovers data from the pages according to the schema.

- **Web-scale wrapper induction task**
  - large number of sites
  - each site comprising the output of an unknown number of scripts, along with a schema
  - per-site training examples can no longer be given

---

[Gulhane et al., 2011]
Learning Extraction rules: Characteristics

- **Languages**
  - Grammars
  - Xpath
  - OXpath
  - Xstring [Grigalis, 2013]

- **Techniques**
  - contextual rules
    (boundaries detection)
  - html-aware
  - visual features
  - hybrid approaches
    [Zhai and Liu, 2005,
    Zhao et al., 2005,
    Grigalis, 2013]

- **Approaches**
  - supervised
  - unsupervised

- **Extraction dimensions**
  - attribute-value pairs from tables
  - record level extractor (lists)
    [Álvarez et al., 2008,
    Zhai and Liu, 2005,
    Zhao et al., 2005]
  - detail page extractor
Outline

1. Wrapper Induction: Task definition
2. Proposed Methodology
3. Dataset
4. Experiment
5. Conclusions
Proposed solution

- usage of *Linked Data* as background Knowledge
- flexible with respect to different domains
- no training data needed
Task definition

- $C$ - set of concepts of interest $C = \{c_1, \ldots, c_i\}$
- their attributes $\{a_{i,1}, \ldots, a_{i,k}\}$
- a website containing Web pages that describe entities of each concept $W_{c_i}$
- **TASK**: retrieve attributes values for each entity on the Web pages
Methodology

1. **Dictionary Generation**
   - for each attribute $a_{i,k}$ of each concept $c_i$, generate a dictionary $d_{i,k}$ for $a_{i,k}$ by exploiting *Linked Data*

2. **Page annotation**
   - $W_{j,i}$, Web pages from a website $j$ containing entities of $c_i$
   - annotate pages in $W_{j,i}$ by matching every entry in $d_{i,k}$ against the text content in the leaf nodes
   - for each match, create the pair $<xpath, value_{i,k}>$ for $W_{j,i}$

3. **Xpath identification**
   - for each attribute, gather all xpaths of matching annotations and their matched values
   - rate each path based on the number of different values it extracts
   - apply $wp_{j,i,k}$ best scoring xpath to re-annotate the website $j$ for attribute $a_{i,k}$. 
Dictionary Generation

- **User Information Need formalisation**
  - translate the concept and attributes of interest to the vocabularies used within the *Linked Data*
  - given a SPARQL endpoint, query the exposed *Linked Data* to identify the relevant concepts
  - select the most appropriate class and properties that describe the attributes of interest
  - using the SPARQL endpoint, query the *Linked Data* to retrieve instances of the properties of interest
Dictionary Generation example

Find all concepts matching the keyword “university”

```
SELECT DISTINCT ?uni WHERE {
  FILTER regex(?lab,"university","i") }
```

Identify all properties defined with this concept

```
SELECT DISTINCT ?prop WHERE {
  ?uni a <http://dbpedia.org/ontology/University> ; ?prop ?o . }
```

Extract all available values of this attribute

```
SELECT DISTINCT ?name WHERE{
  ?uni a <http://dbpedia.org/ontology/University> ;
  <http://dbpedia.org/property/name> ?name .
  FILTER (langMatches(lang(?name), 'EN')). }
```
Website Annotation

AbeBooks.com - Fiction for books

Search Results:

Search Term: Breaking Dawn

Books:

- Breaking Dawn
- Breaking Dawn
- Breaking Dawn
- Breaking Dawn
- Breaking Dawn

Related Subjects:

- Children's Fiction
- Stephenie Meyer

More by Stephenie Meyer:

- Twilight Saga Collection
- Midnight Sun

Price: $15.99

[Image of the AbeBooks.com website with search results for "Breaking Dawn"]
XPath identification

INTUITION
useful patterns will be likely to match a larger variety of dictionary entries

```xml
...
...
```
Outline

1. Wrapper Induction: Task definition
2. Proposed Methodology
3. Dataset
4. Experiment
5. Conclusions
Dataset

- 124K pages collected from 80 websites
- 8 verticals
  - Autos, Books, Cameras, Jobs, Movies, NBA Players, Restaurants, and Universities
- 10 different websites (200 to 2,000 pages per website)
- set of 3 to 5 common attributes to extract

Ground truth

- for each attribute-website pair, a file listing all possible attribute values found on the website is generated
- using a few handcrafted regular expressions over each website
- not all attributes are present on all websites (5 such cases in the dataset)

Hao et al. [Hao et al., 2011]
Dataset

<table>
<thead>
<tr>
<th>Vertical</th>
<th>Web Sites</th>
<th>Web Pages</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auto</td>
<td>10</td>
<td>17923</td>
<td>model (m), price (p), engine (e), fuel economy (f)</td>
</tr>
<tr>
<td>Book</td>
<td>10</td>
<td>20000</td>
<td>title (t), author (a), ISBN-13 (i), publisher (p), publish-date (pd)</td>
</tr>
<tr>
<td>Camera</td>
<td>10</td>
<td>5258</td>
<td>model (md), price (p), manufacturer (m)</td>
</tr>
<tr>
<td>Job</td>
<td>10</td>
<td>20000</td>
<td>title (t), company (c), location (l), date (d)</td>
</tr>
<tr>
<td>Movie</td>
<td>10</td>
<td>20000</td>
<td>title (t), director (d), genre (g), rating (r)</td>
</tr>
<tr>
<td>NBA player</td>
<td>10</td>
<td>4405</td>
<td>name (n), team (t), height (h), weight (w)</td>
</tr>
<tr>
<td>Restaurant</td>
<td>10</td>
<td>20000</td>
<td>name (n), address (a), phone (p), cuisine (c)</td>
</tr>
<tr>
<td>University</td>
<td>10</td>
<td>16705</td>
<td>name (n), phone (p), website (w), type (t)</td>
</tr>
</tbody>
</table>

Hao et al. [Hao et al., 2011]
Outline

1. Wrapper Induction: Task definition
2. Proposed Methodology
3. Dataset
4. Experiment
5. Conclusions
Experiments

- **Topline experiment**
  - artificially created dictionaries specifically tailored to the data
  - minimum level of noise
  - sets a higher limit of the performance of the method

- **Linked Data based WI experiment**
  - dictionaries generated from *Linked Data*
  - generated independently from the data
  - likely to contain noise
Experiments dictionaries

- **Topline** dictionaries
  - for each attribute of a vertical, collect all answers in the ground truth
  - each dictionary contains all (but not only) the true answers

- **Linked Data** dictionaries
  - manually explore *Linked Data* and create queries
  - query Sindice SPARQL endpoint
    - not all verticals/attributes are covered by the *Linked Data*
  - results comparison only for covered attributes

---

\(^{1}\)http://sparql.sindice.com/
## Dictionaries statistics

<table>
<thead>
<tr>
<th>Vertical</th>
<th>Attribute</th>
<th>Topline</th>
<th>LD</th>
</tr>
</thead>
<tbody>
<tr>
<td>University</td>
<td>phone</td>
<td>16973</td>
<td>283</td>
</tr>
<tr>
<td></td>
<td>website</td>
<td>7968</td>
<td>12930</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>9224</td>
<td>13144</td>
</tr>
<tr>
<td></td>
<td>type</td>
<td>68</td>
<td></td>
</tr>
<tr>
<td>Camera</td>
<td>model</td>
<td>5428</td>
<td></td>
</tr>
<tr>
<td></td>
<td>price</td>
<td>1524</td>
<td></td>
</tr>
<tr>
<td></td>
<td>manufacturer</td>
<td>253</td>
<td></td>
</tr>
<tr>
<td>Book</td>
<td>isbn_13</td>
<td>19302</td>
<td>39112</td>
</tr>
<tr>
<td></td>
<td>author</td>
<td>14228</td>
<td>13060</td>
</tr>
<tr>
<td></td>
<td>title</td>
<td>17402</td>
<td>37485</td>
</tr>
<tr>
<td></td>
<td>publication date</td>
<td>6645</td>
<td>3048</td>
</tr>
<tr>
<td></td>
<td>publisher</td>
<td>6175</td>
<td>520</td>
</tr>
<tr>
<td>Movie</td>
<td>genre</td>
<td>1398</td>
<td>114</td>
</tr>
<tr>
<td></td>
<td>title</td>
<td>17146</td>
<td>57292</td>
</tr>
<tr>
<td></td>
<td>mpaa rating</td>
<td>3255</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>director</td>
<td>7398</td>
<td>16079</td>
</tr>
<tr>
<td>Job</td>
<td>title</td>
<td>17712</td>
<td>2381</td>
</tr>
<tr>
<td></td>
<td>date posted</td>
<td>2381</td>
<td></td>
</tr>
<tr>
<td></td>
<td>location</td>
<td>5634</td>
<td></td>
</tr>
<tr>
<td></td>
<td>company</td>
<td>5655</td>
<td></td>
</tr>
<tr>
<td>Auto</td>
<td>model</td>
<td>9916</td>
<td></td>
</tr>
<tr>
<td></td>
<td>price</td>
<td>10792</td>
<td></td>
</tr>
<tr>
<td></td>
<td>engine</td>
<td>2469</td>
<td></td>
</tr>
<tr>
<td></td>
<td>fuel economy</td>
<td>2051</td>
<td></td>
</tr>
<tr>
<td>Restaurant</td>
<td>phone</td>
<td>19510</td>
<td></td>
</tr>
<tr>
<td></td>
<td>cuisine</td>
<td>2378</td>
<td>72</td>
</tr>
<tr>
<td></td>
<td>address</td>
<td>29687</td>
<td>37</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>16631</td>
<td>312</td>
</tr>
<tr>
<td>NBA player</td>
<td>weight</td>
<td>507</td>
<td></td>
</tr>
<tr>
<td></td>
<td>height</td>
<td>121</td>
<td></td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>1457</td>
<td></td>
</tr>
<tr>
<td></td>
<td>team</td>
<td>60</td>
<td></td>
</tr>
</tbody>
</table>
Results

- majority of cases the induced wrappers achieved very high accuracy
- number of cases where they failed
  - incorrect wrapper induced
  - failures often related to the nature of specific websites
  - proposed method is not suitable for all situations
Topline results
Linked Data results

![Graphs showing data results for University, Book, Movie, and Restaurant categories. Each category has a scatter plot with labeled axes and data points indicating some sort of linked data comparison or analysis.]
## Overall results

<table>
<thead>
<tr>
<th>Concept</th>
<th>Hao</th>
<th>Topline</th>
<th>LD</th>
</tr>
</thead>
<tbody>
<tr>
<td>auto</td>
<td>0.71</td>
<td>0.94</td>
<td></td>
</tr>
<tr>
<td>book</td>
<td>0.87</td>
<td>0.85</td>
<td>0.78</td>
</tr>
<tr>
<td>camera</td>
<td>0.91</td>
<td>0.76</td>
<td></td>
</tr>
<tr>
<td>job</td>
<td>0.85</td>
<td>0.82</td>
<td></td>
</tr>
<tr>
<td>movie</td>
<td>0.79</td>
<td>0.86</td>
<td>0.76</td>
</tr>
<tr>
<td>nbaplayer</td>
<td>0.82</td>
<td>0.9</td>
<td>0.87</td>
</tr>
<tr>
<td>restaurant</td>
<td>0.96</td>
<td>0.89</td>
<td>0.69</td>
</tr>
<tr>
<td>university</td>
<td>0.83</td>
<td>0.96</td>
<td>0.91</td>
</tr>
</tbody>
</table>
Outline

1. Wrapper Induction: Task definition
2. Proposed Methodology
3. Dataset
4. Experiment
5. Conclusions
Recap

- is Linked Data suitable Knowledge source for Web Scale Information Extraction?
  - investigation on Wrapper Induction task

- Contributions
  - study on the suitability of *Linked Data* to build dictionaries
  - good results overall for Wrapper Induction
  - some failure cases
Recap

- **Simple idea**
  - generate knowledge resources from Linked Data, in the form of dictionaries
  - use the dictionaries to annotate websites
  - look for recurrent patterns

- **Advantages**
  - no training material required
  - dictionaries are reusable across all websites of a pertinent domain
  - adaption across domains and websites with little human effort

- **Limitations**
  - not all concepts are covered by Linked Data
  - not all concepts are easy to locate in the Linked Data
  - lack of robustness in the learnt wrappers
    - irregular structure of the website
    - quality of the dictionary
Interesting future directions

- investigation on the quality of dictionaries
  - is the dictionary sufficiently large for a task?
  - distributional features of the dictionary
  - compatibility between dictionary and the set of answers
Further reading I

Finding and Extracting Data Records from Web Pages.

Bootstrapping information extraction from semi-structured web pages.
*e European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases.*

Robust web extraction: an approach based on a probabilistic tree-edit model.
*Proceedings of the 35th SIGMOD international conference on Management of data.*

Automatic wrappers for large scale web extraction.

Towards web-scale structured web data extraction.
*Proceedings of the sixth ACM international conference on Web search and data mining - WSDM '13*, page 753.

Web-scale information extraction with vertex.
Further reading II

From One Tree to a Forest: a Unified Solution for Structured Web Data Extraction.
In SIGIR 2011, pages 775–784.

Wrapper Induction for information Extraction.
In IJCAI97, pages 729–735.

Active Learning with Strong and Weak Views: A Case Study on Wrapper Induction.
IJCAI’03 8th international joint conference on Artificial intelligence, pages 415–420.

Learning to adapt web information extraction knowledge and discovering new attributes via a Bayesian approach.
Knowledge and Data Engineering, IEEE, 22(4):523–536.

Web data extraction based on partial tree alignment.
. . .the 14th international conference on World Wide Web, pages 76–85.

Fully automatic wrapper generation for search engines.