Mining the Heterogeneous Transformations for Record Linkage

Matthew Michelson & Craig A. Knoblock
Fetch Technologies & USC Information Sciences Institute

ICAI 2009
### Record Linkage

<table>
<thead>
<tr>
<th>Source 1</th>
<th>Source 2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Manager</strong></td>
<td><strong>Restaurant</strong></td>
</tr>
<tr>
<td>Bobby Jones</td>
<td>California Pizza Kitchen</td>
</tr>
<tr>
<td>William Smith</td>
<td>Arroyo Chop House</td>
</tr>
<tr>
<td>Bobby Smith</td>
<td>Panini Cafe</td>
</tr>
</tbody>
</table>
Heterogeneous Transformations

- Not characterized by a single function (vs. edit distances …)
  - Synonyms/Nicknames
    - Robert $\rightarrow$ Bobby
  - Acronyms
    - California Pizza Kitchen $\rightarrow$ CPK
  - Representations
    - 4$^{th}$ $\rightarrow$ Fourth
  - Specificity
    - Los Angeles $\rightarrow$ Pasadena
  - Combinations
    - Sport Utility 4D $\rightarrow$ 4 Dr SUV
Heterogeneous Transformations

- Applications
  - Record linkage
    - Disambiguating records: Robert = Bobby
  - Information retrieval
    - Search: “4dr SUV” Return: “4 door Sport Util…”
  - Text understanding
    - Acronyms, Synonyms, Specificities
  - Information extraction
    - Expand extraction types
Heterogeneous Transformations

- **Before**: Manually created a priori

- **Now**: Mined from datasets,
  - minimal human effort
Algorithm overview (3 steps)

Step 1
Select record pairs whose TF-IDF score > $T_{\text{cos}}$

Step 2
Mine transformations from these possible matches

Step 3
Prune errant transformations (optional)
Step 1: Selecting record pairs

- Select record pairs that are “close”
  - High token-level similarity
  - Loosens requirement on training data
  - “Close” is not exact
    - Share some similarity
    - Mine transformations from differences

<table>
<thead>
<tr>
<th>Bobby Jones</th>
<th>California Pizza Kitchen</th>
<th>Robert Jones</th>
<th>CPK</th>
</tr>
</thead>
<tbody>
<tr>
<td>William Smith</td>
<td>Arroyo Chop House</td>
<td>Bill Smyth</td>
<td>Arroyo Steak Place</td>
</tr>
</tbody>
</table>
Step 2: Mining Transformations

1. Get co-occurring token sets (not exact matches)

   Bobby Jones  California Pizza Kitchen
   William Smith  Arroyo Chop House

   (Bobby, Robert)
   (William Smith, Bill Smyth)

2. Select token sets with mutual information \( > T_{MI} \)
Mutual Information

\[ MI(s, t) = p(s, t) \times \log \left( \frac{p(s, t)}{p(s)p(t)} \right) \]

- high mutual information
  - occur together with a high likelihood
  - carry information about the transformation occurring in that field for possible matches
Results: Example Mined Transformations

<table>
<thead>
<tr>
<th>Cars Domain</th>
<th>Field</th>
<th>Kelly Blue Book Value</th>
<th>Edmunds Trans.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trim</td>
<td>Coupe 2D</td>
<td></td>
<td>2 Dr Hatchback</td>
</tr>
<tr>
<td>Trim</td>
<td>Sport Utility 4D</td>
<td>4 Dr 4WD SUV or 4 Dr STD 4WD SUV or 4 Dr SUV</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>BiddingForTravel domain</th>
<th>Field</th>
<th>Text Value</th>
<th>Hotel Trans.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local area</td>
<td>DT</td>
<td></td>
<td>Downtown</td>
</tr>
<tr>
<td>Hotel name</td>
<td>Hol</td>
<td>Holiday</td>
<td></td>
</tr>
<tr>
<td>Local area</td>
<td>Pittsburgh</td>
<td>PIT (airport code!)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Restaurants domain</th>
<th>Field</th>
<th>Fodors Value</th>
<th>Zagats Trans.</th>
</tr>
</thead>
<tbody>
<tr>
<td>City</td>
<td>Los Angeles</td>
<td>Pasadena or Studio City or W. Hollywood</td>
<td></td>
</tr>
<tr>
<td>Cuisine</td>
<td>Asian</td>
<td>Chinese or Japanese or Thai or Indian or Seafood</td>
<td></td>
</tr>
<tr>
<td>Address</td>
<td>4th</td>
<td>Fourth</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>and</td>
<td>&amp;</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>delicatessen</td>
<td>delis or deli</td>
<td></td>
</tr>
</tbody>
</table>
Results: Threshold Behavior

- More sensitive to $T_{MI}$ than $T_{cos}$
  - $T_{MI}$ picks transformations, $T_{cos}$ picks candidate matches
- Lower $T_{MI}$ yields more transformations
  - Fewer transformations are common ones
    - bad discriminators for record linkage (e.g. 2dr = 2 Door)
- Setting $T_{cos}$ too high limits what can be mined

Strategy

- Set $T_{cos}$ low enough so it’s not too restrictive
- Set $T_{MI}$ low enough so that you mine a fair number of transformations
  - Yields noise, but does not affect record linkage
## Results: Record Linkage Improvement

RL experiments use $T_{\text{cos}} = 0.65$ and $T_{\text{MI}} = 0.025$, for threshold sensitivity results, see paper

<table>
<thead>
<tr>
<th>Domain</th>
<th>Recall</th>
<th>Prec.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cars domain</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>No trans.</td>
<td>66.75</td>
<td>84.74</td>
</tr>
<tr>
<td>Full trans.</td>
<td><strong>75.12</strong></td>
<td>83.73</td>
</tr>
<tr>
<td>Pruned trans.</td>
<td>75.12</td>
<td>83.73</td>
</tr>
<tr>
<td><strong>BFT domain</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>No trans.</td>
<td>79.17</td>
<td>93.82</td>
</tr>
<tr>
<td>Full trans.</td>
<td><strong>82.89</strong></td>
<td>92.56</td>
</tr>
<tr>
<td>Pruned trans.</td>
<td>82.47</td>
<td>92.87</td>
</tr>
<tr>
<td><strong>Restaurants domain</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>No trans.</td>
<td>91.00</td>
<td>97.05</td>
</tr>
<tr>
<td>Full trans.</td>
<td>91.01</td>
<td>97.79</td>
</tr>
<tr>
<td>Pruned trans.</td>
<td>90.83</td>
<td>97.79</td>
</tr>
</tbody>
</table>

In all domains, not stat. sig. between pruned set & full set $\rightarrow$ pruning optional

Trans. mostly in “cuisine” but decision tree ignores this field
Conclusions and Future Work

- **Conclusions:**
  - Mine transformations without labeling data
  - Pruning errant transformations is optional

- **Future Work**
  - Some fields are ignored, so waste time mining
    - Predictable?
  - Better candidate generation
    - Different methods?
  - Explore technique with other applications
Related Work

- Similar to association rules (Agrawal, et. al. 1993)
  - Even mined using mutual information (Sy 2003)
  - Assoc. rules defined over set of transactions
    - “users who buy cereal also buy milk”
  - Our transformations defined between sources

- Phrase co-occurrence in NLP
  - IR results to find synonyms (Turney 2001)
  - Identify paraphrases & generate grammatical sentences (Pang, Knight & Marcu 2003)
  - We are not limited word based transformations: “4d” is “4 Dr”
    - No syntax is needed
Thank you!