Database Entity Resolution using Program Synthesis

Rohit Singh, Vamsi Meduri, Samuel Madden, Paolo Papotti, Jorge-Arnulfo Quiané-Ruiz, Armando Solar-Lezama, Nan Tang

MIT (Cambridge, MA), QCRI (Doha, Qatar), ASU (Tempe, AZ)

ExCAPE Review Meeting May 10, 2016
Problem Overview

Input:

<table>
<thead>
<tr>
<th>Name</th>
<th>Position</th>
<th>Institution</th>
<th>Gender</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wei Wang</td>
<td>Associate Prof</td>
<td>UNSW</td>
<td>M</td>
</tr>
<tr>
<td>Wei Wang</td>
<td>Prof</td>
<td>UCLA</td>
<td>F</td>
</tr>
<tr>
<td>Sam Madden</td>
<td>Prof</td>
<td>MIT</td>
<td>M</td>
</tr>
<tr>
<td>Patrick Valduriez</td>
<td>Director</td>
<td>INRIA</td>
<td>M</td>
</tr>
</tbody>
</table>

Output:

\[ r \in R \quad s \in S \]

if sim(Name, Person) and sim(Institution, University) and = (Gender, Sex) then r and s match
Problem Overview

Input:

<table>
<thead>
<tr>
<th>Name</th>
<th>Position</th>
<th>Institution</th>
<th>Gender</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wei Wang</td>
<td>Associate</td>
<td>UNSW</td>
<td>M</td>
</tr>
<tr>
<td>Wei Wang</td>
<td>Prof</td>
<td>UCLA</td>
<td>F</td>
</tr>
<tr>
<td>Sam Madden</td>
<td>Prof</td>
<td>MIT</td>
<td>M</td>
</tr>
<tr>
<td>Patrick Valduriez</td>
<td>Director</td>
<td>INRIA</td>
<td>M</td>
</tr>
</tbody>
</table>

Output:

if \( \text{sim}(\text{Name, Person}) \) and \( \text{sim}(\text{Institution, University}) \) and \( =(\text{Gender, Sex}) \) then \( r \) and \( s \) match

Similarity Predicates: \( \text{sim}_\text{function}(\text{Name, Person}) \geq \text{threshold} \)
Why does it matter?

Linking Census Data over the years

Duplicate Contacts Detection

Better Web Search (Knowledge Graph)

Comparison Shopping

Spam Detection
Machine Reading
Public Health
Counter-terrorism... [1]

Challenges

Schema Matching

Reasoning about Similarity Functions

Structure of the Matching Rules

Noisy data, Slow convergence
Approach

**Schema Matching**

- **Schema R**
  - Purchase Order
  - Product
  - Billing Name
  - Billing Address
  - Shipping Name
  - Shipping Address

- **Schema S**
  - POrder
  - Payee Info
  - Article

**Leverage Prior Work**

**Reasoning about Similarity Functions**

**Structure of the Matching Rules**

1. if $\text{sim}(A, A')$ and $\text{sim}(B, B')$ and $\text{sim}(C, C')$ then $r$ and $s$ match
2. if $\text{sim}(A, A')$ and $\text{sim}(B, B')$ then $r$ and $s$ match
3. if $\text{sim}(C, C')$ and $\text{sim}(D, D')$ then $r$ and $s$ match
4. if $\text{sim}(A, A')$ and $\text{sim}(B, B')$ and $\text{sim}(C, C')$ and $\text{sim}(D, D')$ then $r$ and $s$ match

**SyGuS**

- Use SyGuS model functions with tables

**CEGIS + RANSAC**

- Noisy data, Slow convergence
Motivation

• User provided Example pairs
  • Some of these are wrong (✗)

• Challenges
  • Too many examples
  • Don’t know which ones are wrong

• Hypotheses
  • Matching Rules are easy to learn
  • Small fraction of wrong examples
How it works?

- **Counter-Example Guided Inductive Synthesis**
  1. Pick an example pair randomly
  2. Synthesize a matching function
  3. Include an incorrectly matched example pair (✗)
  4. Go to 2 (repeat)

Matched correctly (□) or incorrectly (✗)
How it works?

• Counter-Example Guided Inductive Synthesis

Matched correctly (□) or incorrectly (○)
How it works?

• Counter-Example Guided Inductive Synthesis

Matched correctly (□) or incorrectly (○)
How it works?

- Counter-Example Guided Inductive Synthesis

Matched correctly (□) or incorrectly (○)
How it works?

- Counter-Example Guided Inductive Synthesis

Matched correctly (□) or incorrectly (○)
How it works?

- Counter-Example Guided Inductive Synthesis
- Record and Restart
- Best over K runs (RANSAC)

Matched correctly (☐) or incorrectly (○)
Improvements

• Non-uniform sampling across restarts
  • Penalize examples resulting into UNSAT
  • More penalty for examples picked at a later CEGIS iteration

• Resilient synthesizer
  • Maximize number of matches
  • Threshold on number of allowed mismatches

Less likely to be picked again
Experiments: Effectiveness Metric

- **Precision** = \( \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \)
- **Recall** = \( \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \)
- **Accuracy** = \( \frac{\text{True Positive} + \text{True Negative}}{\text{Total Population}} \)
- **F - Measure** = \( 2 \left( \frac{1}{\text{Precision}} + \frac{1}{\text{Recall}} \right)^{-1} \)
Experiments: Comparison with SIFI [2]

Expert provided fixed rule structure used for both
5000 example pairs

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Best SIFI</th>
<th>Best Synth</th>
</tr>
</thead>
<tbody>
<tr>
<td>Restaurant</td>
<td>0.86</td>
<td>0.85</td>
</tr>
<tr>
<td>Cora</td>
<td>0.72</td>
<td>0.75</td>
</tr>
<tr>
<td>Dbgen</td>
<td>0.957</td>
<td>0.964</td>
</tr>
</tbody>
</table>

10-fold 10% training examples (Avg F-measure)

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Best SIFI</th>
<th>Best Synth</th>
</tr>
</thead>
<tbody>
<tr>
<td>Restaurant</td>
<td>0.63</td>
<td>0.76</td>
</tr>
<tr>
<td>Cora</td>
<td>0.37</td>
<td>0.52</td>
</tr>
<tr>
<td>Dbgen</td>
<td>0.87</td>
<td>0.93</td>
</tr>
</tbody>
</table>

100-fold 1% training examples (Avg F-measure)

Experiments: Comparison with SIFI [2]

Expert provided structure for SIFI, bounded DNF grammar for Synthesis
5000 example pairs

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Best SIFI</th>
<th>Best Synth</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cora</td>
<td>0.72</td>
<td>0.87</td>
</tr>
<tr>
<td>Dbgen</td>
<td>0.957</td>
<td>0.98</td>
</tr>
</tbody>
</table>

10-fold 10% training examples (Avg F-measure)

Summary

• Industry need for better Entity Resolution methods
• Synthesis enabled solution can
  • Outperform or match state of the art with “small” sized training sets
  • Potentially save effort and $$ by reducing expert involvement
• Work in Progress:
  • Evaluating on industry datasets