State of the Art Ontology Mapping

By Justin Martineau

Motivation

Ontology Mapping Benefits:

• Business
  • Better communication with subcontractors
• Artificial Intelligence Researchers
  • Source of Training Data
  • Way to share learning results
• Programmers
  • Tool to make better applications
• Laymen
  • Indirectly through Tools
  Ex: Tool for comparison of Similar Products

Theory

Applicable Techniques:

• Machine Learning
• Natural Language Processing
• Heuristics
• Database Schema Merging
• Formal Concept Analysis (Produce a Concept Lattice)
  • Cluster into Objects with same subset of properties & Properties belonging to object clusters
Practice

- FCA-Merge - uses Formal Concept Analysis, & NLP
- IF-Map - uses thy of info flow (Barwise & Seligman 97)
- SMART - uses linguistic similarity and heuristics
- PROMPT - uses linguistic similarity and heuristics
- GLUE - uses ML, Meta-Learning, Naïve Bayes, Relaxation Labeling …
- CAIMAN - uses ML, text classification and probability
- ITTalks - uses text classification, and Bayesian reasoning
- ONION - uses Heuristics, user checks input, ML of user choices
- ConceptTool - uses Description Logic, linguistics, heuristics

Prompt for Protégé

- Many Different Tools
  - iPrompt - Ontology-mergering tool
  - AnchorPrompt - Ontology-alignment tool
  - PromptDiff - Ontology-versioning tool
  - PromptFactor - Determine semantic sub-ontologies

Prompt Architecture

iPrompt Ontology-Merging Flowchart

1. Make initial suggestions
2. Select the next operation
3. Perform automatic updates
   - Find inconsistencies and potential problems
4. Make suggestions
**iPrompt UI**

- Suggestions ordered based on last operation to maintain user's focus
- Can prefer one Ontology over another, so conflicts are resolved in its favor
- Suggestions are Explained
- Logs operations, Log can be open and applied.

**Ex: Merging 2 Ontologies**

Before Starting the Merger

<table>
<thead>
<tr>
<th>CMU Hierarchy</th>
<th>UMD Hierarchy</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Publications</td>
<td>- Architecture</td>
</tr>
<tr>
<td>- Academic, Missis</td>
<td>- Activity</td>
</tr>
<tr>
<td>- Office</td>
<td>- Address</td>
</tr>
<tr>
<td>- Organization, org</td>
<td>- Agent</td>
</tr>
<tr>
<td>- Person</td>
<td>- Person</td>
</tr>
<tr>
<td>- Organization</td>
<td>- GradStudent</td>
</tr>
<tr>
<td>- Project</td>
<td>- Undergraduate</td>
</tr>
<tr>
<td>- Research, Deliverables</td>
<td>- Research</td>
</tr>
<tr>
<td>- Employee</td>
<td>- Student</td>
</tr>
<tr>
<td></td>
<td>- Graduate</td>
</tr>
<tr>
<td></td>
<td>- Staff</td>
</tr>
</tbody>
</table>

An Empty Ontology with a list of suggestions

Person Class added, Receives slots from both Ontologies

<table>
<thead>
<tr>
<th>Name</th>
<th>Documentation</th>
<th>Card</th>
</tr>
</thead>
<tbody>
<tr>
<td>Person</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Role</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Email</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Template slots

- Name: [text]
- Type: [text]
- Continuity: [text]
- Other: [text]
**iPrompt Conflicts**

- Name Conflict
- Dangling Pointers (Suggests Importing)
- Class Hierarchy Redundancy
- Slot Value Restriction Violations

**PromptDiff - Ontology Version Tracking**

- Unix Diff doesn't work well with Ontologies
- Heuristic Algorithm
- Produces Structured Diff Representation

---

**Ex: Wine Ontology**

---

**Questions**

---

Figures and Images from:
The PROMPT Suite: Interactive Tools For Ontology Merging And Mapping