Ontology Matching: via a meta ontology or a machine learning approach?

Valentina Cordì

Università degli Studi di Genova – DISI

23 Maggio 2006
An ontology typically provides a vocabulary that describes a domain of interest and a specification of the meaning of terms used in the vocabulary. In open or evolving systems, such as the Semantic Web, different parties would, in general, adopt different ontologies.

Thus, just using ontologies, like just using XML, does not reduce heterogeneity: it raises heterogeneity problems at a higher level.
An ontology typically provides a vocabulary that describes a domain of interest and a specification of the meaning of terms used in the vocabulary. In open or evolving systems, such as the Semantic Web, different parties would, in general, adopt different ontologies. Thus, just using ontologies, like just using XML, does not reduce heterogeneity: it raises heterogeneity problems at a higher level.

A single ontology is no longer enough to support the tasks envisaged by a distributed environment like the Semantic Web.
Ontology Matching

The process of discovering correspondences between semantically related entities of the given ontologies.
Introduction

Ontology Matching

The process of discovering correspondences between semantically related entities of the given ontologies.

Ontology matching is seen as a solution provider in today’s landscape of ontology research.

As the number of ontologies that are made publicly available and accessible on the Web increases steadily, so does the need for applications to use them.

Matching could provide a common layer from which several ontologies could be accessed and hence could exchange information in semantically sound manners.
Types of Mismatches

- Language-level mismatches
  Difference in expressiveness or semantics of ontology language

- Ontology-level mismatches
  Difference in the structure of semantics of the ontology
Language-level Mismatches

- Syntax

- Expressiveness
  e.g., presence of disjoints, negations, expressions, unions, intersections, metaclasses, etc. in the language

- Semantics of primitives
  e.g., union vs intersection semantics for multiple domain and range declarations
Ontology-level Mismatches

- The same terms describing different concepts
- Different terms describing the same concept
- Different modeling paradigms
  e.g., intervals or points to describe temporal aspects
- Different modeling conventions
- Different levels of granularity
- Different coverage
- Different points of view
- ...
Developing such matching has been the focus of a variety of works originating from diverse communities over a number of years.
Developing such matching has been the focus of a variety of works originating from diverse communities over a number of years.

In this talk we review and present two of these works:

1. shared ontology approach
2. machine learning approach (GLUE)
Using a Shared Ontology

- “Upper” ontologies designed to support information integration
  - designed in principled way
  - provide common reference terminology
  - SUMO, DOLCE

- Domain-specific interlingua
  - PSL
Ontology Matching

SUMO - Suggested Upper Ontology

It is being created as part of the IEEE Standard Upper Ontology Working Group

- data interoperability
- information search and retrieval
- automated inferencing
- natural language processing

Merging publicly available ontological content into a single, comprehensive, and cohesive structure.
SUMO - Suggested Upper Merged Ontology

- Extract nouns and verbs from a source text
- Find classes in SUMO for the nouns and verbs
- Record a mapping as being either equal, subsuming or instance.
- Create a subclass of SUMO if it’s a subsuming mapping
- Add properties to the subclass
  - reusing SUMO properties
  - extending SUMO properties by creating a subrelation of an existing property
- Add English definition to the class
  - define constraints that express how the subclass is more specific than the superclass
- Express the classes and properties in KIF and begin creating axioms, based on the English definitions created previously
SUMO - Suggested Upper Ontology

- entity
  - physical
    - object
  - process
    - dual object process
    - intentional process
      - intentional psychological process
      - recreation or exercise
      - organizational process
    - guiding
    - keeping
      - maintaining
    - repairing
    - poking
  - content development
  - making
    - constructing
  - manufacture
    - publication
    - cooking
  - searching
  - social interaction
    - maneuver
  - motion
  - internal change
    - shape change
  - abstract
WordNet

is divided by part of speech into nouns, verbs, adjectives, and adverbs. The nouns are organized as a hierarchy of nodes, where each node is a word meaning or, as it is termed in WordNet, a synset. A synset is simply a set of English words that express the same meaning in at least one context.

00047131 04 n 02 accession 0 addition 0 001 @ 09536731 n 0000 |
something added to what you have already; the librarian shelved the new accessions; he was a new addition to the staff
Mapping Methodology

- synonymy
- hypernymmy
- instantiation
Mapping Methodology

- synonymy
- hypernymmy
- instantiation

A living organism lacking the power of locomotion
Mapping Methodology

- synonymy
- hypernymmy
- instantiation

00008864 03 n 03 plant 0 flora 0 plant_life 0 027 @ . . . | a living organism lacking the power of locomotion

00008864 03 n 03 plant 0 flora 0 plant_life 0 027 @ . . . | a living organism lacking the power of locomotion &%Plant=
Mapping Methodology

04719796 09 n 01 Christian Science 0 001 @ 04718274 n 0000
| religious system based on teachings of Mary Baker Eddy
| emphasizing spiritual healing
Mapping Methodology

04719796 09 n 01 Christian_Science 0 001 @ 04718274 n 0000
| religious system based on teachings of Mary Baker Eddy
emphasizing spiritual healing

04719796 09 n 01 Christian_Science 0 001 @ 04718274 n 0000
| religious system based on teachings of Mary Baker Eddy
emphasizing spiritual healing &%ReligiousOrganization+
DOLCE

A formal foundational ontology developed as a top-level ontology in the WonderWeb project.

- a starting point for building new ontologies
- a reference point for easy and rigorous comparisons among different ontological approaches
- a foundational framework for analyzing, harmonizing and integrating existing ontologies and metadata standards (by manually mapping existing categories into the categories assumed by some module(s) in the library).
Facilitate correct and complete exchange of process information among manufacturing systems such as scheduling, process modeling, and process planning.

The PSL Ontology is a set of first-order theories organized into PSL-Core and a partially ordered set of extensions.

Grüninger and Kopena [1] developed an integration architecture with the PSL ontology at the center and mappings between ontologies for specific manufacturing processes and the PSL ontology.
PSL - Process Specification Language

PSL is intended to be used as a mediating ontology that is independent of the applications’ ontologies and that is used as a neutral interchange ontology.

The mappings are defined semi-automatically by presenting ontology developers with a set of questions (in natural language) helping them to map terms in their process-specific ontology to the terms in PSL.

**Note**

The generation of these mappings is defined formally and is not based on heuristics. These mappings can be composed to provide mappings between any task-specific ontologies.
A scenario in which two software agents, Alice and Bob, need to exchange process information.

Alice’s designer specifies the semantic mapping between Alice’s ontology and the PSL ontology, and Bob’s designer specifies the semantic mapping between Bob’s ontology and the PSL ontology.

When Alice and Bob first interact, they use these previously specified mappings to automatically generate the semantic mappings between each other’s ontologies.
Using a Machine Learning Approach

GLUE

a system that employs machine learning techniques to find such mappings

Given two ontologies, for each concept in one ontology finds the most similar concept in the other ontology.

- joint probability distribution
- multiple learning strategies
- relaxation labeling
Ontology matching

- An ontology
  - concepts organized into a taxonomy tree
  - each concept has:
    - a set of attributes
    - a set of instances
  - relations among concepts

- Matching
  - concepts
  - attributes
  - relations
Ontology matching

CS Dept. US
- Entity
  - Undergrad Courses
  - Grad Courses
  - People
    - Faculty
      - Assistant Professor
      - Associate Professor
      - Professor
  - Staff

CS Dept. Australia
- Entity
  - Courses
  - Staff
    - Academic Staff
    - Technical Staff
  - Lecturer
  - Senior Lecturer
  - Professor
Sim(Concept A, Concept S) = \frac{P(A \cap S)}{P(A \cup S)} = \frac{P(A, S)}{P(A, -S) + P(A, S) + P(-A, S)}
Similarity Measures
The GLUE Architecture

- Relaxation Labeling
- Similarity Matrix
- Similarity Estimator
- Meta Learner
- Base Learner

Input:
- Taxonomy $O_1$ (tree structure + data instances)
- Taxonomy $O_2$ (tree structure + data instances)

Process:
- Common Knowledge & Domain Constraints
- Similarity Function
- Joint Probability Distribution $P(A, B), P(A', B)$

Output:
- Matches for $O_1$, Matches for $O_2$
Multi-Strategy Learning

Many different type of information that a learner can glean from the training instances.

- *frequencies* of words
- instance *names*
- the value *formats*
- the *characteristics* of value distributions

Instead of training a single learner $L$, we train a set of learners $L_1, \ldots, L_n$ called *base learners*.

To classify an instance we apply the *base learners* to the instance and combine their predictions using a *meta-learner*.
Multi-Strategy Learning

The Content Learner: Exploits the frequencies of words in the textual content of an instance to make predictions. GLUE does not handle attributes directly; rather, it treats them and their values as the textual content of the instance.

The Name Learner: Similar to the Content Learner, but works using the full name of the input instance. The full name of an instance is the concatenation of concept names leading from the root of the taxonomy to that instance.

The Meta-Learner: Assigns to each base learner a learner weight that indicates how much it trusts that learner’s predictions. It combines the base learners’ predictions via a weighted sum.
Constraints in Taxonomy Matching

- Domain-dependent convey the general knowledge about the interaction between related nodes.
- Domain-independent convey the knowledge about the interaction between specific nodes in the taxonomies.

<table>
<thead>
<tr>
<th>Constraint Types</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domain-Dependent</td>
<td></td>
</tr>
<tr>
<td>Neighborhood</td>
<td>Two nodes match if their children also match. Two nodes match if their parents match and at least x% of their children also match. Two nodes match if their parents match and some of their descendants also match.</td>
</tr>
<tr>
<td>Union</td>
<td>If all children of node X match node Y, then X also matches Y.</td>
</tr>
<tr>
<td>Subsumption</td>
<td>If node Y is a descendant of node X, and Y matches PROFESSOR, then it is unlikely that X matches ASST PROFESSOR. If node Y is NOT a descendant of node X, and Y matches PROFESSOR, then it is unlikely that X matches FACULTY.</td>
</tr>
<tr>
<td>Domain-Dependent</td>
<td></td>
</tr>
<tr>
<td>Frequency</td>
<td>There can be at most one node that matches DEPARTMENT CHAIR.</td>
</tr>
<tr>
<td>Nearby</td>
<td>If a node in the neighborhood of node X matches ASSOC PROFESSOR, then the chance that X matches PROFESSOR is increased.</td>
</tr>
</tbody>
</table>
Relaxation Labeling

Efficient technique to solve the problem of assigning labels to nodes of a graph, given a set of constraints.

The label of a node is typically influenced by the features of the node’s neighborhood in the graph.

The influence of a node’s neighborhood on its label is quantified using a formula for the probability of each label as a function of the neighborhood features.

It regards nodes (concepts) in $O_2$ as labels, and recast the problem as finding the best label assignment to nodes (concepts) in $O_1$, given all knowledge we have about the domain and the two taxonomies.
GLUE - Conclusion

- based on well-founded notions of semantic similarity
- used of machine learning, and in particular, of multi-strategy learning, for computing concept similarities.
- used of relaxation labeling to the ontology-matching context.
Credits

Parts of this talk are based on:


Credits

N. F. Noy.

I. Niles, and A. Pease.
Towards a standard upper ontology.