Ontology Alignment, Matching and Translation

In the old days

- People have been building knowledge based systems for ~40 years
- There was not much interest in integrating them before the mid 80s
- Cyc argued (~1985) for the utility of having a shared KB, but just one that all would refer to
- Agent oriented approaches in the 90s imagined having multiple share ontologies
  - KIF was proposed as an interlingua for importing and exporting knowledge

Ontology matching

- Matching or aligning knowledge encoded in different KR languages can be very hard
- Differences in the KR languages can be major or subtle and both can cause problems
  - E.g., FOL, vs. bayesian vs defaults vs stereotypes vs …
- Trying to deal with this problem usually means that you need to adopt a very abstract and flexible interlingua
- It’s much easier if we can limit ourselves to translation between different schemas in the same KR languages
  - e.g., like the problem of schema mapping in RDBMs

The Semantic Web Vision

- Everyone uses the same Knowledge Representation language – OWL
- There is no assumption of having ONE ontology for any topic
  - Assume many will be used and invest in techniques for translation
  - Analogy for how the UN manages translations
- OWL also has primitives that can describe some mappings
  - foaf:Person owl:sameClassAs wn:Human
  - wn:Human rdfs:subClass spire:homoSapien
But…

- Mappings can be complex
  - \( o1:Boy = \text{intersection}(o2:Human, o2:Male, \text{complement}(o2:Adult)) \)
  - Here’s where DL can help and do so efficiently
- Not all useful mappings can be expressed in FOL
- \( o1:\text{Mammal} \sim o2:\text{FurryAnimal} \)
  - Dolphins are mammals but are not furry
  - We would benefit from conditional probabilities, e.g., \( p(o1:\text{Mammal}|o2:\text{FurryAnimal}) \) and \( p(o2:\text{FurryAnimal}|o1:\text{Mammal}) \)
- Peng and others are exploring this idea
  - Probabilities can come from human judgments or shared data
  - Need to respect the FOL constraints inherent in OWL

Discovering Mappings

- Automatically discovering the mappings at a schema level
  - Hard problem without common instance data
- Semi-automatically discovering the mappings at a schema level
  - Can use OWL’s constraints, e.g., if \( a:C1 < a:C2 \) and \( b:C3 < b:C4 \), then \( b:C4 < a:C1 \) implies \( b:C3 < a:C1 \) and \( b:C3 < a:C2 \)
- Using instance data to suggest or rule out alignments
  - If we’re lucky, the ontologies might share some instances
  - We might also note patterns (e.g., “138-35-9866”) in literal data
- We can also get the mappings manually or collect them using Swoogle

Using Mappings

- Once we have the mappings, how do we use them?
- One model for translation: merge the ontology and instance data from the source data and the ontology from the target ontology
- Add bridging axioms for source and target ontologies
  - \( o1:Boy = \text{intersection}(o2:Human, o2:Male, \text{complement}(o2:Adult)) \)
  - \( o3:Journal < o4:Serial \)
- Draw all possible interferences over the instance data
- Write out the instance data expressed in the target ontologies

Using Mappings

- Such systems have been built
  - Dejing Dou, Drew McDermott, and Peishen Qi
- And the approach may be used in many ad hoc, one-off translation systems
- But no widely used tools are available, to my knowledge
Let’s do this as a project?