Semantic Web - KEN3140

Lab 8: 60min

Constructing advanced OWL axioms in Protege, performing advanced reasoning tasks and ontology debugging

Kody Moodley & Michel Dumontier, 28 September 2020

Purpose of this document

This document outlines your tasks for Lab 8 which follow on from Lab 7. You will load a variation of the family relations ontology from Lab 7 into Protégé and extend the ontology to describe more conceptual knowledge about family relations, which special focus on the constructs and reasoning assumptions covered in Lecture 8.

Learning objectives:

1. How to identify which kinds of knowledge cannot be captured in ALC and require additional constructs from SROIQ (OWL 2 DL)
2. How to identify the specific DL features required to capture a logical statement about the domain
3. To be able to anticipate the effects on reasoning when enforcing constraints such as transitivity and symmetry to roles (for smaller ontologies with less than 15 individuals)
4. To be able to explain the intuitions behind Open World Assumption and Unique Name Assumption and to anticipate the effects they have on entailments for smaller ontologies
5. How to use the OWL reasoner and explanation facilities in Protege to diagnose logical incoherences in an ontology
6. How to use explanations (justifications) for logical incoherences to pinpoint problematic axioms and formulate strategies to rectify the errors

Tasks

Task 1a [15min]

Load the family relations ontology (“KEN3140_Lab8_myfamilyrelations.ttl”). If you have not done so from Lab 7, create new equivalence axioms in the ontology to define the following concepts:

- Aunt
- Uncle
- Grandmother

- Grandfather

After creating your axioms, run the reasoner to check which members of the family become "realized" under these concepts. Please also use the DL Query Tab to query for instances of these new concepts (**hint:** although it is not mandatory, you can test your definitions out using the DL query tab before adding the equivalence axioms to the ontology).
If you are stuck, have a look at Lecture 7's slides for some examples.

Task 1b [15min]

Once you have added the equivalence axioms requested, your tasks are to:

- introduce new relations into the object property hierarchy for **father, mother, grandfather, grandmother, relative, son and daughter**. **NB:** you have to reuse existing vocabulary - do not create your own custom relations. Possible sources to look for vocabulary are: schema.org, wikidata.org, linked open vocabularies, foaf and dbpedia.
- make a choice about where to place these new properties in the property hierarchy in Protege.
- Pick 3 - 5 individuals from the ABox who are persons that have children, parents and siblings. Remove all role assertions concerning parents, children and siblings for these individuals. Replace these role assertions with new ones using **father, mother, son and daughter.**
- Run the reasoner (choose HermiT or Pellet since they are OWL 2 DL reasoners). Observe the inferences made and whether your changes to the ontology have changed the inferences that the reasoner makes. Remember the inferences (implicit axioms) are highlighted with a yellow background in Protege.

Task 2a [15min]

Once you have added the relations from Task 1b in the correct locations in the hierarchy:

- Select which role constraints (transitivity, symmetry, reflexivity etc.) to enforce on these. You can toggle the checkboxes for these constraints in the object properties tab of Protege. Verify the effect that these constraints have on the inferences made by the reasoner, by observing the inferences made before toggling the constraints, and then after.
- Create two unique **role composition axioms** to capture different ways that someone can be a **relative** to another person in the ontology. To give an example of one or inspiration: we know that if a is a **parent** of b and b is a **sibling** of c then a is a relative of c. Create a further two unique **role composition axioms** to capture different ways that someone can be a **grandparent** to another person. Verify the effect that these constraints have on the inferences made by the reasoner, by observing the inferences made before toggling the constraints, and then after.

Task 2b [15min]

Your first task is to:

- Identify all members of the family in the ontology who have at least 2 sons. **Hint:** you will use the DL Query tab to express a class expression query which you can then execute to return your results (and the answer). Check the Lecture 8 slides to see what DL feature you need to express this class. If you execute your query without modifying the ontology, you may notice that it will not return any results. **Hint:** you need to tell the OWL reasoner that all the individual names cannot refer to the same objects in the domain (we want to simulate the unique name assumption - UNA) - see Figure 1.

Figure 1: UI component potentially useful for Task 2b.

  After modifying the ontology to make sure the UNA is being made, execute the query again in the DL query tab to see if there is a difference in the results.
- Examine the definitions for “Nerd” and “SportsFanatic” in the ontology. Modify the definition of “Nerd” to mean that: only those people having **all** their hobbies as either “Cosplay”, “PCgame” or “Comic” (or any combination thereof) **with no other hobbies,** are considered as a “Nerd”. Run the reasoner before and after to see who is classified under “Nerd” and compare the results.
- Modify the definition of “SportsFanatic” to mean that only those people that have at least 3 unique hobbies that concern sports are considered a “SportsFanatic”. **Hint:** this task may be trickier than the previous one. You will need a different approach. Think about using *disjoint classes* to get the answer. Run the reasoner before and after to see who is classified under “SportsFanatic” and compare the results.