Knowledge Engineering and Semantic Web

Exercise Sheet: 5

Will be discussed on: June 29,2021

TUTORS:
Yaser Jaradeh, Salomon Kabenamualu, Vitalis Wiens

LECTURE SLIDES: The lecture slides can be accessed through the following link:

https://slidewiki.org/playlist/237

QUESTIONS: Please don't hesitate to ask any questions. Questions help you and your peers.
PRINT: Please consider the environment before printing the exercise.

Required Slides `https://slidewiki.org/deck/90759/05-rdf-and-rdfs-semantics`

1 Inference

1. Hierarchy of properties: Select the correct inferences among the following ones.

a)
```
:a :p1 :b .
:a :p2 :c .
->
:b rdfs:subPropertyOf :c .
```

b)
```
:a :p1 :b .
:b rdf:type :C .
->
:p1 rdfs:range :C .
```

c)
```
:a :p1 :b .
:p2 rdfs:subPropertyOf :p1 .
->
:a :p2 :b .
```

d)
```
:p1 rdfs:subPropertyOf :p2 .
:a :p1 :b .
->
:a :p2 :b .
```

2. Hierarchy of Classes: Select the correct inferences among the following ones.

a)
```
:A rdfs:subClassOf :B .
:c rdf:type :A .
->
:c rdf:type :B .
```

b)
```
:a :p1 :b .
:a :p2 :c .
:b rdf:type :B .
:c rdf:type :C .
:B rdfs:subClassOf :C .
->
:p1 rdfs:subPropertyOf :p2 .
```

c)
```
:p1 rdfs:domain :A .
:p1 rdfs:range :C .
:p2 rdfs:domain :B .
:p2 rdfs:range :D .
:p1 rdfs:subPropertyOf :p2 .
->
:A rdfs:subClassOf :B .
:C rdfs:subClassOf :D .
```

d)
```
:a :p1 :b .
:p2 rdfs:domain :C .
:p1 rdfs:subPropertyOf :p2.
->
:a rdf:type :C .
```

3. Equivalence of Classes: Select the correct inferences among the following ones.

a)
```
:A rdfs:subClassOf :B .
:B rdfs:subClassOf :C .
:C rdfs:subClassOf :D .
:D rdfs:subClassOf :A .
->
:A , :B , :C , :D
are equivalent classes.
```

b)
```
:A rdfs:subClassOf :B .
:B rdfs:subClassOf :C .
:c rdf:type :A .
->
:c rdf:type :C .
```

c)
```
:A rdfs:subClassOf :B .
:B rdfs:subClassOf :A .
:c rdf:type :A .
:d rdf:type :A .
->
:c and :d are equivalent.
```

d)
```
:p1 rdfs:subPropertyOf :p2 .
:p2 rdfs:subPropertyOf :p1 .
:p1 rdfs:range  :B ;
    rdfs:domain :A .
:p2 rdfs:range  :D .
:p2 rdfs:domain :C .
->
:A is equal to :C and :B is equal to :D.
```

2 Consider the following statements:

a) Represent them in RDF Turtle serialization.

b) Select the correct ones.

1. $< rdfs : subClassOf^I, rdfs : Resource^I > \in I_{EXT}(rdfs : domain^I)$.
2. $< rdf : List^I, rdf : rest^I > \in I_{EXT}(rdfs : domain^I)$.
3. $I_{CEXT}(rdfs : Class^I) \subseteq I_{CEXT}(rdfs : Resource^I)$.
4. $< rdfs : domain^I, rdf : Property^I > \in I_{EXT}(rdf : type^I)$.
5. If $< x, y > \in I_{EXT}(rdfs : domain^I)$ and $< u, v > \in I_{EXT}(x) \rightarrow u \in I_{CEXT}(x)$.

3 For the following knowledge base, indicate which statement can be entailed. Prove the true answers with proof-theoretic semantics.

```
@prefix ex:   <http://example.org> .
@prefix rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd:  <http://www.w3.org/2001/XMLSchema#> .



ex:dog      rdfs:subClassOf   ex:animal .
ex:horse    rdfs:subClassOf   ex:creature .
ex:person   rdfs:subClassOf   ex:creature .


ex:isEnemyOf     rdfs:subPropertyOf   ex:knows .
ex:isEnemyOf     rdfs:domain          ex:person;
                 rdfs:range           ex:person .
ex:isFriendOf    rdfs:subPropertyOf   ex:knows .




ex:LuckLuke      a   ex:person .
ex:JollyJumper   a   ex:horse .
ex:Rantanplan    a   ex:dog .


ex:LuckyLuke     ex:isFriendOf   ex:JollyJumper .
ex:JollyJumper   ex:isFriendOf   ex:Rantanplan .
ex:LuckyLuke     ex:isEnemyOf   exJoeDalton .
```

Statements:

1. `ex:Rantanplan   a   ex:creature.`
2. `ex:Rantanplan   ex:isFriendOf   ex:JollyJumper.`
3. `ex:LuckyLuke   ex:isFriendOf   ex:RantanPlan.`

4. `ex:LuckyLuke ex:knows ex:JoeDalton`.
5. `ex:JoeDalton ex:isEnemyOf ex:LuckyLuke`.
6. `ex:JoeDalton a ex:creature`.

Required Slides `https://slidewiki.org/deck/90751/06-owl-syntax-and-intuition`

4 Review Questions

1. Among the following statements about OWL, which ones are correct?
   (a) `owl:Nothing` is the subclass of all OWL classes.
   (b) All object properties are functional.
   (c) `:A owl:disjointWith :B` means the classes `:A` and `:B` don't have any instance in common.
   (d) `owl:sameAs` denotes the similarity of two classes in OWL.
2. Of the statements below, which one is correct about `owl:NegativePropertyAssertion`?
   (a) It's supported by all versions of OWL.
   (b) It can take `Classes` and `Individuals` as its source.
   (c) `owl:targetIndividual` is an `owl:ObjectProperty`.
   (d) It's used to express negative facts in an OWL ontology.
3. Which ones of the statements below are correct?
   (a)
   ```
   :p   a   owl:ObjectProperty ;
        rdfs:range   xsd:string .
   ```
   (b)
   ```
   :a   :p   :c .
   :b   :p   :c .
   :p   a    owl:FunctionalPropery .
   →
   :a   owl:sameAs   :b.
   ```
   (c)
   ```
   :p   a   owl:DatatypeProperty .
   :b   :p   :c .
   →
   :c   rdfs:subClassOf   owl:Class .
   ```
   (d)
   ```
   :a   :p   :c .
   :b   :p   :c .
   :p   a   owl:inverseFunctionalProperty .
   →
   :a   owl:sameAs   :b .
   ```

5 Given are the following OWL expressions in RDF/XML syntax.

a.
```
<owl:Restriction>
  <owl:onProperty rdf:resource="#hasParent" />
  <owl:someValuesFrom rdf:resource="#Physician" />
</owl:Restriction>
```

b.
```
<owl:Class>
  <owl:intersectionOf rdf:parseType="Collection">
    <owl:Class>
      <owl:oneOf rdf:parseType="Collection">
        <owl:Thing rdf:about="#Tosca" />
        <owl:Thing rdf:about="#Salome" />
      </owl:oneOf>
    </owl:Class>
    <owl:Class>
      <owl:oneOf rdf:parseType="Collection">
        <owl:Thing rdf:about="#Turandot" />
        <owl:Thing rdf:about="#Tosca" />
      </owl:oneOf>
    </owl:Class>
  </owl:intersectionOf>
</owl:Class>
```

c.
```
<owl:Class rdf:about="#MusicDrama">
  <owl:equivalentClass>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Opera"/>
        <owl:Class rdf:about="#Operetta"/>
        <owl:Class rdf:about="#Musical"/>
      </owl:unionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>

<owl:Class rdf:about="#Opera">
  <rdfs:subClassOf rdf:resource="#MusicDrama"/>
</owl:Class>

<owl:Class rdf:about="#Operetta">
  <rdfs:subClassOf rdf:resource="#MusicDrama"/>
  <owl:disjointWith rdf:resource="#Opera"/>
</owl:Class>

<owl:Class rdf:about="#Musical">
  <rdfs:subClassOf rdf:resource="#MusicDrama"/>
  <owl:disjointWith rdf:resource="#Opera"/>
  <owl:disjointWith rdf:resource="#Operetta"/>
</owl:Class>
```

1. Explain the meaning of each expression in your own words.
2. Represent each expression in the Turtle or Manchester-syntax serialization.

6 Indicate which of the following statements are logical consequences of the knowledge base below.

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>.
@prefix xsd: <http://www.w3.org/2001/XMLSchema#>.
@prefix owl: <http://www.w3.org/2002/07/owl#>.
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix ex: <http://example.org#>.
ex:LuckyLuke   a   ex:Person;
               ex:isFriendOf   ex:JollyJumper, ex:Shrief .
ex:JollyJumper   a   ex:Horse .
ex:Rantanplan   a   ex:Dog ;
                ex:isFriendOf   ex:JollyJumper .
ex:Dog   rdfs:subClassOf   ex:Animal .
ex:Horse   rdfs:subClassOf   ex:Animal .
ex:LukePet   rdfs:subClassOf   [
  rdf:type   owl:Class ;
  owl:intersectionOf  ( ex:Animal   [
                                    owl:equivalentClass [
                                    rdf:type   owl:Restriction;
                                    owl:onProperty   ex:isFriendOf;
                                    owl:hasValue   ex:LuckyLuke]
                                        ])] .
ex:Creature   rdfs:subClassOf   [
                                 a   owl:Class;
                                 owl:unionOf   (ex:Animal   ex:Person)].
ex:LuckyLuke   ex:isEnemyOf   ex:JoeDalton .
ex:isEnemyOf   a   owl:SymmetricProperty;
               rdfs:subPropertyOf   ex:knows ;
               rdfs:domain   ex:Person .
```

Statements:

a. `ex:Rantanplan   a   ex:creature .`

b. `ex:Rantanplan   ex:isFriendOf   ex:Shrief .`

c. `ex:Shrief   a   ex:LukePet .`

d. `ex:Rantanplan   a   ex:LukePet .`

e. `ex:JoeDalton   ex:knows   ex:LuckyLuke .`

f. `ex:JoeDalton   a   ex:creature .`

g. Talk about the statement d, in the case we add these triples to our knowledge base.

```
ex:isFriendOf   a   owl:SymmetricProperty, owl:transitiveProperty.
```

7 Modeling in OWL:

Given are some facts about the SDA research group and the “Semantic Data Web” lecture. Model them in an appropriate way as an OWL ontology.

1. ”SDW” and ”Semantic Data Web” are two names for the same lecture.
2. SDW *lecture* is different from SDW *seminar*.
3. If a mentor is the supervisor of a student, then that student is supervised by that mentor.
4. SDA has some PhD or Master students. (In your model, take it into consideration that one student cannot be PhD and Master student in the same time.)

5. All tutors of SDW are students and enrolled in Uni Bonn.
6. A student eligible to register their master thesis should have achieved at least 1 and at most 2 seminars.
7. Professor Jens Lehmann is not a lecturer of SDW.
8. The SDA group offers two *different* lectures: SDW and KGA.
9. Tutors of SDW are Mehrdad, Vitalis, Rohan and Hantong.
10. Students who failed SDW are students who have enrolled in SDW and haven't passed it.