COMP318

Ontologies and Semantic Web

RDF - Part 10

Dr Valentina Tamma

V.Tamma@liverpool.ac.uk

Where were we

- RDFS schema language
  - RDFS as an extension of RDF

Semantics

- RDFS vocabulary has built-in “meaning”
- RDFS Semantics
  - Makes meaning explicit
  - Defines what follows from an RDF graph
- Semantic notions
  - Subgraph
  - Entailment

Logical consequence

- What can we derive from the triples below?

```
mus: Shoegazing  rdfs:subClassOf mus: Alternative_rock
mus: Alternative_rock rdfs: subClassOf mus: Music_genre
```

Logical consequence

- What can we derive from the triples below?

mus: Shoegazing rdfs:subClassOf mus: Alternative_rock
mus: Alternative_rock rdfs: subClassOf mus: Music_genre
mus: Shoegazing rdfs: subClassOf mus: Music_genre

Example of logical consequence

- In deriving the triple

```
mus: Shoegazing  rdfs: subClassOf mus: Music_genre
```

we used the transitive property holding for **`subclassOf`**

- we used properties of the RDFS vocabulary

RDFS entailment

- An RDFS graph entails implicit triples
  - triples not explicitly contained in the graph, but that can be derived from an RDFS graph
    - using the special semantics of the vocabulary of the graph
      - vocabulary of the graph: set of names which occurs as the subject, predicate, object
  - Interpretations assign special meaning to the symbols in a particular vocabulary

Subgraph

- G2 is a subgraph of G1 if and only if the triples in G2 are a subset of the triples in G1

```
ex1:johnURI    ex1:hasName   ex1:johnfullname
ex1:johnfullname ex1:firstName  ”John”
ex1:johnfullname ex1:surname   ”John”
```

- Each of the following set of triples is a subgraph:

```
ex1:johnURI    ex1:hasName   ex1:johnfullname
```

```
ex1:johnURI    ex1:hasName   ex1:johnfullname
ex1:johnfullname ex1:firstName  ”John”
```

Entailment regimes

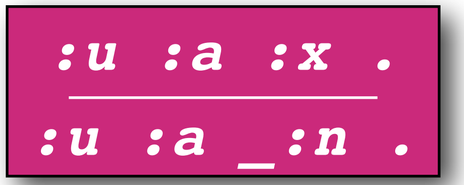
- Three entailment regimes
  - **simple entailment**: no particular extra conditions are posed on a vocabulary, including the RDF vocabulary itself;
    - it involves only graph transformations.
  - RDF entailment;
  - RDFS entailment: some extra conditions are posed by in the form of axiomatic triples and semantic conditions

Let’s state the formalism

- `:a, :b,`
  - refer to any arbitrary URI,
    - (i.e. anything that can appear in the **predicate** of a triple)
- `:u, :v,`
  - refer to any arbitrary URI or blank node ID
    - (i.e. anything that can appear in the **subject** of a triple)
- `:x, :y,`
  - refer to an arbitrary URI, blank node ID or literal
    - (i.e. anything that can appear in the **object** of a triple)
- `_:n,`
  - refer to the ID of a blank node
    - (i.e. appearing as a **subject** or **object**)
- `:l,`
  - refers to a literal
    - (i.e. a string that is sometimes found in the **object**)

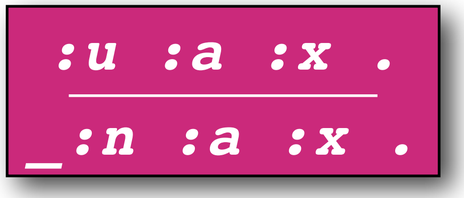
Simple entailment deduction rules

- If the triple `:u, :a, :x` is valid, then we can entail that the triple `:u, :a, _:n` is valid



se1

- If the triple :u, :a, :x is valid, then we can entail that the triple _:n, :a, :x is valid



se2

Formalism

:a, :b, refer to any arbitrary URI

:u, :v, refer to any arbitrary URI or blank node ID

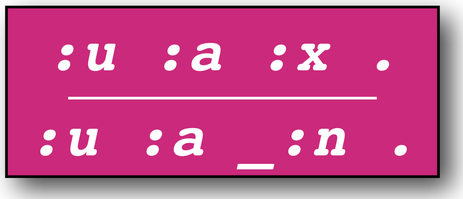
:x, :y, refer to any arbitrary URI, blank node ID or literal

_:n, refer to the ID of a blank node

:l, refers to a literal

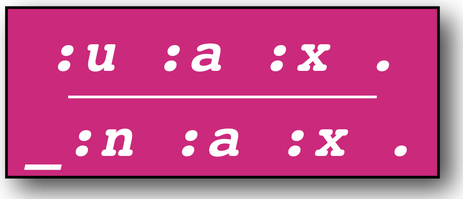
Simple entailment deduction rules

- URIs are all treated equally
  - we can decide whether a graph entails by applying the following rules se1 and se2 and adding the resulting triples to the original graph



se1

*I.e. if you have an **object**, you can derive a **blank node for that object**, as long as the **subject and predicate** stay the same.*



se2

*I.e. if you have a **subject**, you can derive a **blank node for that subject**, as long as the **object and predicate** stay the same.*

Simple entailment deduction rules

- A graph G1 simply entails a graph G2, if G1 can be extended to a graph G’1 by virtue of the rules se1 and se2 such that G2 is contained in G’1
  - G2 ⊆ G’1

COMP318

Ontologies and Semantic Web

End of RDF - Part 10

Dr Valentina Tamma

`V.Tamma@liverpool.ac.uk`