Overview of OWL Constraints for RDFS

Peter F. Patel-Schneider

Nuance Communications

pfpschneider@gmail.com

31 October 2014

Inputs

G an RDF graph

C a set of constraints and recognition axioms in the form of OWL axioms

- The constraints only use vocabulary from *G*.
- The recognition axioms define vocabulary not in *G*.
- Recognition axioms that are not recursive can act as macros in constraints.
- Recursive recognition axioms probably should not have recursively defined classes and properties appearing in negated positions.

Example

G

B rdfs:subclassOf C .
a rdf:type B .   a D `"hi"` .
a R b .   a R a .

C

$C \sqsubseteq \geq 2\, R \qquad C \sqsubseteq \leq 2\, R$
$C \sqsubseteq \forall D.\text{xsd:string}$

$\geq 1\, R \sqsubseteq \forall D.\text{xsd:string}$

$a \in \exists R.\text{xsd:string}$
$a \in D : \texttt{"hi"} \qquad a \in \geq 2\, R$

$E \equiv C \sqcap \leq 3\, R$

$F \equiv \geq 1\, R.F$
$G \equiv \leq 1\, R.G$

Output

Basic Operation

- whether a constraint was violated - no constraint above is violated
- instances of the new vocabulary - E above contains a
- recursive recogntion can be complex - F above contains a

Advanced Operation

- the reduced set of constraint violations
  - needs a mechanism for naming constraints and specifying what to report
  - the OWL annotation mechanism can be used for this purpose

Definition

A constraint in C is violated if the OWL axiom is not satisfied in the minimal RDFS model of G.

- The minimal RDFS model of G is essentially the Herbrand model of the RDFS closure of G.
- Because RDFS models are infinite, some care has to be taken to ensure that an infinite set of constraint violations does not have to be returned. This can be done by lumping all the ”similar”, unused domain elements together and reporting only a single violation.

Implementation

- Constraints can be turned into SPARQL queries that are satisfied if the query is empty.
- Non-recursive recognitions can be turned into SPARQL queries.
- Recursive recognitions can be more complex to implement.

Glitches and Extensions

Glitches Blank nodes as values for data properties work in a rather strange way.

Extensions A useful extension would be to allow datatype functions and relations from SWRL, e.g., to say that the value of one property is the sum of the values of two other properties.