CS5560 Knowledge Discovery and Management

Problem Set 7 & 8

Submission Deadline: July 28, 2017

ttps://goo.gl/forms/aTXnl4oRHMdS8j1L2

Name: Revanth Chakilam
Class ID: 02

References

I. Logical knowledge representation

First Order Logic Reference: http://pages.cs.wisc.edu/~dyer/cs540/notes/fopc.html

1) Let us define the statements as follows:
   - G(x): "x is a giraffe"
   - F(x): "x is 15 feet or higher,"
   - Z(x): "x is animal in this zoo"
   - M(x): "x belongs to me"

   Express each of the following statements in First-Order Logic using G(x), F(x), Z(x), and M(x).

   a) Nothing, except giraffes, can be 15 feet or higher;
   b) There is no animal in this zoo that does not belong to me;
   c) I have no animals less than 15 feet high.
   d) All animals in this zoo are giraffes.

2) Which of the following are semantically and syntactically correct translations of "No dog bites a child of its owner"? Justify your answer
   a) $\forall$ x Dog(x) $\Rightarrow$ $\neg$Bites(x, Child(Owner(x)))
   b) $\neg\exists$ x, y Dog(x) $\wedge$ Child(y, Owner(x)) $\wedge$ Bites(x, y)
   c) $\forall$ x Dog(x) $\Rightarrow$ ($\forall$ y Child(y, Owner(x)) $\Rightarrow$ $\neg$Bites(x, y))
   d) $\neg\exists$ x Dog(x) $\Rightarrow$ ($\exists$ y Child(y, Owner(x)) $\wedge$ Bites(x, y))

3) For each of the following queries, describe each using Description Logic
Reference: http://www.inf.ed.ac.uk/teaching/courses/kmm/PDF/L3-L4-DL.pdf

   a) Define a person is Vegan
   b) Define a person is Vegetarian
   c) Define a person is Omnivore

II. SPARQL

Reference: https://www.w3.org/2009/Talks/0615-qbe/

Design a SPARQL query for following queries and show an expected output.

Query #1: Multiple triple patterns: property retrieval
Find me all the people in Tim Berners-Lee's FOAF file that have names and email addresses. Return each person's URI, name, and email address.

Query #2: Multiple triple patterns: traversing a graph
Find me the homepage of anyone known by Tim Berners-Lee.

Query #3: Basic SPARQL filters
Find me all landlocked countries with a population greater than 15 million.

Query #4: Finding artists' info
Find all Jamendo artists along with their image, home page, and the location they're near, if any.

Query #5. Design your own query

III. SWRL

References:
https://www.w3.org/Submission/SWRL/
https://dior.ics.muni.cz/~makub/owl/

Design SWRL rules for the following cases

Rule #1: design hasUncle property using hasParent and hasBrother properties
Rule #2: an individual X from the Person class, which has parents Y and Z such that Y has spouse Z, belongs to a new class ChildOfMarriedParents.
Rue #3: persons who have age higher than 18 are adults.
Rue #4: Compute the person's born in year
Rule #5: Compute the person's age in years
Rule #6: Design your own rule

I. Logical Knowledge Representation:

Logical knowledge representation is the field of artificial intelligence (AI) dedicated to represent information about the world in a form that a computer system can utilize to solve complex tasks such as diagnosing a medical condition or having a dialog in natural language. Knowledge representation incorporates findings from psychology about how humans solve problems.

First Order Logic (FOL or FOPC) syntax:

→ Constant symbols (i.e; individuals in this world")
→ Function symbols (mapping individuals to individuals)
→ Predicate symbols (mapping from individuals to truth values)

FOL supplies Ex: x, y, these primitives

1) Variable symbols Ex: x, y
2) Connectives not (¬) and (∧) or (∨), implies ⇒ if and only if (⇔)
3) Quantifiers : Universal (∀) and Existential (∃)

(1) Possible translation for the given statements are

$\forall x (\neg G_1(x) \rightarrow \neg F(x))$ or $\forall x (F(x) \rightarrow G(x))$

$\neg \exists x (z(x) \wedge \rightarrow M(x))$ or $\forall x(z(x) \rightarrow m(x))$

$\forall x \ (M(x) \rightarrow F(x))$

$\forall x \ (Z(x) \rightarrow G(x))$

(2) Syntactic Analysis :

The goal of syntactic analysis is to determine whether the sent string on input is a sentence in the given natural language.

Semantic Analysis :-

Semantic and pragmatic analysis make up the most complex phase of language processing as they build on upon results of all mentioned disciplines.

a) $\forall x \; Dog(x) \Rightarrow \neg \; Bites(x, child(owner(x)))$

No dog bites dogs and owner of children

b) $\neg \; \exists x, y \; Dog(x) \wedge child(y, owner(x)) \wedge Bites(x, y)$

No dog Bites owners children.

c) $\forall x \; Dog(x) \Rightarrow (\forall y \; child(y, owner(x)) \Rightarrow \neg \; Bites(x, y)$

All dog do not bite their children of owner.

d) $\neg \exists x \; Dog(x) \Rightarrow (\exists y \; child(y, owner(x)) \wedge Bites(x, y)$

Dog Bite the children of owners.

Therefore, the correct translations are (b) and (c)

(3) Description Logic : Description logic allows formal concep definitions that can be reasoned about to be Expressed. It is an important element of the semantic web.

a) Define a person is vegan

People who dosnot eat or use animal products

∀ eats ¬ Animal products.

b) Define a person is vegetarian

People who dosnot eat animal products

∀ eats ¬ Animal.

c) Define a person is omnivore

Animal / person eats food of both plant and animal

∃ eats Animal.

II SPARQL :

SPARQL is the query language of the semantic web. It lets us

1) pull values from structured and semi-structured data
2) Explore data by querying unknown relationships
3) Perform Complex joins of disparate databases in a single simple query.
4) Transform RDF data from one vocabulary to another.

Query #1: Multiple triple patterns : property retrieval

```
PREFIX FOAF : <http://xmlns.com/foaf/0.1>

Select *
where { ? person foaf : name ? name
        ? person foaf : mbox ? email
```

Expected o/p: person name email

<http://www.w3.org/people/Berners-Lee/ "Amy Vander Hie/" <mailto: amy@w3.org>
card#amy>

<http://www.w3.org/people/Berners-lee/card#ds> "Dean Jackson" <mailto: dean@w3.org>

<http://www.w3.org/people/Berners-lee/card#ens> "Edd Dumbill" <mailto: edd@usefulinc.com>

Query #2: Multiple triple patterns: traversing a graph

```
PREFIX foaf: <http>://xmlns.com/foaf/0.1>
PREFIX card: <http://www.w3.org/people/Berners-lee/card#>
   Select ?homepage.
FROM <http://www.w3.org/people/Berners-Lee/card>
Where {
        card:i foaf:knows ?known
         ?known foaf:homepage ?homepage
```

Expected o/p:

<http://purl.org/net/eric/>

<http://www.mellon.org/about-foundation/staff/program-area-staff/irafuman>

<http://www.johnseelybrown.com/>

<http://heddley.com/edd>

#Query-3: Basic SPARQL filters:

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX type: <http://dbpedia.org/class/yago/>
PREFIX prop: <http://dbpedia.org/property/>
Select ? country_name ? population
where {
    ∀ eats ⊤ Animal products
```

b) Define a person is vegetarian

people who dosnot eat animal

∀ eats ⊤ Animal.

c) Define a person is omnivore

Animal / person eats food of both plant and Animal

∃ eats Animal

#Query-4: Finding artists info.

```
PREFIX mo: <http://purl.org/ontology/mo/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?name ? img ? hp ? loc
where {
    ?a a mo:music Artist;
    foaf:name ?name;
```

Expected Output:

name img

" cicada " xsd: string http://img.jamendo.com/artists/h/hattrimany.jpg

" Hace Soul " xsd: string http://img.jamendo.com/artists/hace-soul.jpg

http://www.hacesoul.com http://sws.geoname.org/2510769

" Vincent " xsd: string http://img.jamendo.com/artists/v/vincent.jpg

http://v-joudrier.free.fr/site v http://sws.geonames.org/3020781

Query-5: Design your own query

Is the amazon river longer than the Nile River?

PREFIX prop: <http://dbpedia.org/property/>

<http://dbpedia.org/resource/Nile> prop: length ?nile

FILTER(?amazon > ?nile)

}

Expected Output: <?xml version = "1.0"?>

<sparql xmlns: "https://www.w3.org/2005/sparql-results#

xmlns: xsi = "http://www.w3.org/2001/sw/DataAccess/xsd" >

<head> </head>

<boolean>true </boolean>

</sparql>

11. SWRL:

Semantic Web Rule Language

Combining OWL and Rule ML

Rule #1:- design has uncle property using has parent & has Brother properties.

has parent (?x1, ?x2) ∧ has Brother (?x2, ?x3) ⇒ has uncle (?x, ?x3)

Rule #2: an Individual x from the person class, which has parents y and z such that has spouse z, belongs to a new class child of Maried Parents

person (?x), has parent (?x, ?y), has parent (?x, ?z), has Spouse (?x, ?z) child of maried parents (?x)

Rule #3: persons who have age higher than 18 are adults

person (?p), has Age (?p, ?age), swrlb: greater than (?age, 18) → Adult (?p)

Rule #4: Compute the person born in year

person (?p), born on Date (?p, ?date), xsd: date (?date), swrlb: date (?date, ?year, ?month, ?date, ? timezone) → born In year (?p, ?year)

Rule #5: Compute the person's age in years.

person (?p), born In year (?p, ?year), my: this Year (?new year) swrlb: subtract (?age, ?new year, ?year) → has Age (?p, ?age)

Rule #6: Design your own rule

→ design has son property using has child and man properties

has child (?x, ?y) ∧ man (?y) ⇒ has Son (?x, ?y)