

Examen session 2 du cours Recherche d'information / web sémantique

ENSEEIH 3A Info – année 2018-2019

Partie « Web sémantique »

Documents autorisés - durée indicative : 50 mn – barème 10 pts

1. Questions de cours (3 points). Réponses COURTES attendues

1.1. Donner 3 manières dont un moteur de recherche comme Google peut utiliser un graphe de connaissances générales pour fournir de meilleures réponses à des requêtes.

- . trouver des mots synonymes à ceux de la requête pour la reformuler ou l'élargir à ces mots
- . désambigüiser les mots de la requête pour éviter de fournir des réponses non pertinentes
- . reconnaître des entités de la base de connaissances dans la question ou dans la réponse, et fournir des informations complémentaires disponibles dans la BC et concernant ces entités

1.2. Définir ce qu'est base de connaissances du web sémantique ; donner des exemples de bases de connaissances disponibles sur le web des données

Une base de connaissance est une ressource dans lesquelles des connaissances sont représentées selon un format permettant leur manipulation informatique, et surtout la production de raisonnement ou d'inférences. Dans le contexte du web sémantique, les bases de connaissances sont représentées sous forme de graphes de concepts reliés par des relations sémantiques. Ces graphes sont représentés en RDF.

Exemples de BC du LOD : DBpedia, Yago, BabelNet, WikiData

2. Exercice (3 points)

2.1. Le texte ci-dessous définit une propriété de type owl:DataProperty. Dans cette définition :
corriger les 3 erreurs de syntaxe présents (vérifier la « ponctuation », les préfixes et les propriétés ou relations utilisés)

.ajouter les informations manquantes signalées par ??? (pour répondre, indiquer le numéro de la ligne et pour chacune, ce qui doit remplacer ???)

```
1 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
  @prefix dbo: <http://dbpedia.org/ontology/> .
  @prefix owl: <http://www.w3.org/2002/07/owl#> .
  @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
5 @prefix wikidata: <http://www.wikidata.org/entity/> .
  @prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

  dbo:inseeCode    rdf:type    rdf:Property .
  dbo:inseeCode    rdf:type    owl:DataProperty .
10 dbo:inseeCode    rdf:    owl:subPropertyOf    dbo:codeSettlement .
    dbo:inseeCode    owl:equivalentProperty    wikidata:P374 ; ,
```

```

    rdfs:label      "INSEE code"@en ,
                  "INSEE-code"@nl ,
    rdfs:domain    dbo:Settlement ;
15  rdfs:range      xsd:string ;
    rdfs:comment    "numerical indexing code used by the French
National Institute for Statistics and Economic Studies (INSEE) to
identify various entities"@en .

```

```

1 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
  @prefix dbo:   <http://dbpedia.org/ontology/> .
  @prefix owl:   <http://www.w3.org/2002/07/owl#> .
  @prefix rdfs:   <http://www.w3.org/2000/01/rdf-schema#> .
5 @prefix wikidata: <http://www.wikidata.org/entity/> .
  @prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

```

```

    dbo:inseeCode  rdf:type      rdf:Property .
    dbo:inseeCode  rdfs:type    owl:DataProperty .
10 dbo:inseeCode  owl:subPropertyOf  dbo:codeSettlement . erreur 1
    dbo:inseeCode  owl:equivalentProperty  wikidata:P374 ;
                  rdfs:label      "INSEE code"@en ,
                  "INSEE-code"@nl ;
                  rdfs:domain    dbo:Settlement ;
15                  rdfs:range    xsd:String ; erreur 2
                  rdfs:comment    "numerical indexing code used by the French National Institute for
Statistics and Economic Studies (INSEE) to identify various entities"@en .

```

2.2. En utilisant la même syntaxe,

- Définir que le range de la propriété `dbo:inseeCode` est une chaîne de caractères
- `dbo:inseeCode rdfs:range xsd:String`
- définir que cette propriété est de type `owl:FunctionalProperty`.
- `dbo:inseeCode rdf:type owl:FunctionalProperty`

2.3. Que signifie que cette propriété soit « fonctionnel » ? après avoir lancé un raisonneur, quel nouveau type sera associé à « Toulouse » ?

Si la propriété est « fonctionnel » alors elle doit être obligatoirement définie pour toute entité de type ville. Le raisonneur va conclure que Toulouse a `dbo:area`

3. Exercice (4 points)

La ressource WIKIDATA (www.wikidata.org) est une base de connaissances construite à partir de pages Wikipedia. Les espaces de noms suivants lui sont associés :

- wdt pour les propriétés ; **wdt:P31 est la propriété « instance-de », wdt:P279 « sous-classe de » ; wdt:P17 : « a pour origine » ; wdt:P583 « a pour date » : wdt:P571 « date début »**
- wd pour les entités ; **wd:Q11424 est la classe FILM ; wd:Q30 est la classe USA**

Soit la requête SPARQL suivante :

PREFIX xsd: <<http://www.w3.org/2001/XMLSchema#>>

```
SELECT ?film ?date ?prop ?valProp WHERE {
  ?film (wdt:P31/wdt:P279*) wd:Q11424.
  ?film ?prop ?valProp.
  OPTIONAL { ?film wdt:P583 ?date. }
  OPTIONAL { ?film wdt:P571 ?date. }
  FILTER((BOUND(?date)) && ((DATATYPE(?date)) = xsd:dateTime )
}
LIMIT 20
```

3.1 Expliquer précisément les données que retourne cette requête et comment elles seront présentées. Pourquoi OPTIONAL et BOUND sont-ils utilisés ?

Elle retourne un tableau avec 4 colonnes

Chaque ligne est de type film (instance ou sous-classe de la classe FILM). Pour chaque film, elle affiche son identifiant, la date de sa sortie ou sa date (une des 2 prop) si elle est connue (Optional) et à condition que la date soit donnée avec le type XSD xsd:dateTime. Elle affiche aussi toutes les propriétés associées et leurs valeurs. > une ligne par film et prop/valeur.

3.2 Modifiez-là pour qu'elle retourne le nom (rdfs:label) des films à la place de leur identifiant, et les labels des valeurs liées aux propriétés (?valProp) demandées à la place de leur identifiant.

PREFIX xsd: <<http://www.w3.org/2001/XMLSchema#>>

```
SELECT ?filmLabel ?date ?prop ?valPropLabel WHERE {
  ?film (wdt:P31/wdt:P279*) wd:Q11424.
  ?film ?prop ?valProp.
  ?valProp rdfs:label ?valPropLabel.
  ?valProp rdfs:label ?valPropLabel.
  OPTIONAL { ?film wdt:P583 ?date. } # date
  OPTIONAL { ?film wdt:P571 ?date. } #date début
  ?film rdfs:label ?filmLabel.
  FILTER((BOUND(?Date)) && ((DATATYPE(?Date)) = xsd:dateTime)) )
```

3.3 Modifier la requête du 3.3 pour n'afficher que des films dont le label et les labels des propriétés sont en anglais et dont le pays d'origine est les USA (utiliser les propriétés et les classes fournies dans le début de l'énoncé en gras).

PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

```

SELECT ?filmLabel ?date ?prop ?valPropLabel WHERE {
  ?film (wdt:P31/wdt:P279*) wd:Q11424.
  ?film ?prop ?valProp.
  ?valProp rdfs:label ?valPropLabel.
  ?film wdt:P17 wd:Q30.
  ?valProp rdfs:label ?valPropLabel.
  OPTIONAL { ?film wdt:P583 ?date. } # date
  OPTIONAL { ?film wdt:P571 ?date. } #date début
  ?film rdfs:label ?filmLabel.
  FILTER((LANG(?filmLabel)) = "en")
  FILTER((LANG(?valPropLabel)) = "en")
  FILTER((BOUND(?Date)) && ((DATATYPE(?Date)) = xsd:dateTime)) )

```