RECUPERACIÓN DE INFORMACIÓN

Sistema de Recuperación de Información Semántico

Curso 2019-2020

Víctor Miguel Peñasco Estívalez - 741294
Rubén Rodríguez Esteban - 737215

Índice de contenidos

Introducción **2**

Descripción del sistema **2**

Cambios realizados al RDFS original 2

Consultas definidas 3

Evaluación del sistema **4**

Apéndice **5**

Introducción

En este trabajo se ha procedido a realizar un sistema de recuperación de información semántico. Para poder construir dicho sistema se han empleado los siguientes recursos:

- La colección de datos del repositorio Zaguán en formato XML facilitada por el profesorado de la asignatura.
- Las necesidades de información elegidas entre las propuestas por todos los equipos.
- El modelo RDFS en Turtle con el modelo conceptual que va a seguir el sistema de recuperación semántico a diseñar.

Usando este modelo RDFS se ha creado un modelo enriquecido en OWL, que cuenta con elementos que han podido mejorar la calidad del sistema de recuperación semántico con respecto del tradicional.

Con el fin de sacar partido a la búsqueda semántica se ha creado un modelo de terminológico, que sigue el esquema SKOS, que define los términos usados para describir los recursos. Este modelo terminológico permite proporcionar una jerarquía de los términos usados en las consultas, sinónimos, y otros términos también usados en los metadatos de la colección.

Finalmente se han creado las consultas SPARQL adecuadas para cada una de las necesidades de información seleccionadas anteriormente. Estas consultas se han definido de forma manual de forma que sean las más adecuadas al esquema OWL diseñado.

Descripción del sistema

Cambios realizados al RDFS original

Primeramente se hizo una pequeña modificación al modelo RDFS, ya que en la propiedad *nombre* se habían establecido múltiples dominios: *persona* y *organización*. Este hecho supone un AND, pero la intención realmente era establecer una relación OR. Para esto se optó por crear dos propiedades de nombre: *nombrePersona* y *nombreOrganización*. Otra solución podría haber sido crear una superclase, de las que fueran hijos *Persona* y *Organización*, y que la propiedad nombre tuviera como dominio únicamente esa superclase.

Los grandes cambios realizados al modelo RDFS se corresponden con una traducción a OWL. Para ello se han realizado las siguientes modificaciones:

- Cada una de las clases definida como una *rdfs:Class* se ha convertido en una subclase de *owl:Thing*.

- Las propiedades rdf:Property cuyo rango es una clase se han convertido en owl*:ObjectProperty.*
- Las propiedades rdf:Property cuyo rango es un tipo de dato se han convertido en *owl:DatatypeProperty*.
- Se ha especificado en la propiedad del objeto *tema* un *owl:propertyChainAxiom* entre *tema* y *skos:broader,* para así hacer inferencias entre los términos *skos:Concept* del tesauro definido relacionados con *skos:broader.* Gracias a esto, si en un documento existe el tema Zaragoza, también se incluye el tema Aragón y el tema España.

Consultas definidas

Para las necesidades de información especificadas por el profesorado de la asignatura se ha procedido a elaborar las correspondientes consultas en lenguaje SPARQL escritas en un fichero a razón de una por línea. El código de las consultas puede observarse en la sección Apéndice del documento.

Estas consultas han sido escritas empleando índices textuales para poder proporcionar ránking a los resultados obtenidos. Se han creado índices sobre los literales que son rango de las propiedades título y descripción.

Aunque las consultas tienen sus particularidades, todas siguen un esquema similar. En todas ellas se tiene como requisito obligatorio que el documento se corresponda con una serie de temas. En la necesidad de información 01-2 (Figura A1), también se incluye como restricción obligatoria que la fecha esté en un rango concreto de años. En la necesidad de información 15-4 (Figura A5), también es una restricción que en el campo autor del documento se encuentre la palabra Javier, lo cual se ha especificado a través de un FILTER con expresión regular.

Además de las restricciones obligatorias de las consultas que se acaban de describir, se han incluido una restricciones opcionales para hacer uso de los índices textuales creados sobre el título y la descripción. La consulta que se hace sobre el índice utilizando *text:query* es de una sola palabra, que se corresponde con la que se ha visto empíricamente que ofrece mejores resultados de cara al score. Este score, que por defecto se le da el valor 0, es devuelto por la query tanto para el índice sobre el título como para el índice sobre la descripción. En cada una de las consultas se ha especificado que estos dos scores se sumen, y los resultados sean mostrados según el orden decreciente del valor del score total. Es así como se consigue que los resultados devueltos más relevantes sean los primeros de la lista.

Evaluación del sistema

A continuación se ofrece una comparativa del rendimiento obtenido por ambos sistemas de recuperación diseñados, mostrándose la evaluación del sistema tradicional a la izquierda, y la del semántico a la derecha.

```
TOTAL
precision 0.289
recall 0.338
F1 0.312
prec@10 0.440
MAP 0.526
interpolated_recall_precision
0.000 0.788
0.100 0.573
0.200 0.541
0.300 0.339
0.400 0.092
0.500 0.079
0.600 0.000
0.700 0.000
0.800 0.000
0.900 0.000
1.000 0.000
```

```
TOTAL
precision 0.530
recall 0.164
F1 0.251
prec@10 0.520
MAP 0.840
interpolated_recall_precision
0.000 0.950
0.100 0.639
0.200 0.338
0.300 0.000
0.400 0.000
0.500 0.000
0.600 0.000
0.700 0.000
0.800 0.000
0.900 0.000
1.000 0.000
```

En vista de los resultados anteriores se puede observar que el sistema de recuperación semántico se comporta de manera muy diferente al tradicional.

El recall del sistema semántico es bastante menor al del tradicional debido a que en las consultas se ha buscado obtener resultados lo más precisos posibles a través de fuertes restricciones obligatorias.

Por otra parte, la precisión, tal y como estaba previsto, ha aumentado significativamente, incrementando así el valor de otras medidas de evaluación como la prec@10 o el MAP.

Se podría considerar este cambio en los resultados como positivo, ya que en muchos contextos, la precisión, sobre todo de los primeros resultados, es mucho más importante que obtener una gran colección de resultados, de entre los cuales muy pocos son relevantes.

Aunque un sistema de recuperación de información semántico suele ofrecer mejores resultados que uno tradicional, hay que tener en cuenta los costes de crear cada uno de ellos, ya que para un sistema semántico se requiere un esfuerzo extra que en algunos contextos puede no merecer la pena.

Apéndice

```
PREFIX  xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX  text: <http://jena.apache.org/text#>
PREFIX  rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX  m:    <http://github.com/vpec/Recoreon/>
PREFIX  m_skos: <http://github.com/vpec/Recoreon/skos#>

SELECT  ?uriDoc
WHERE
  { ?uriDoc  m:tema   m_skos:enfermedad ;
             m:tema   m_skos:ocular ;
             m:fecha  ?fecha
    FILTER ( ( "2010"^^xsd:gYear <= ?fecha ) && ( ?fecha <= "2015"^^xsd:gYear ) )
    OPTIONAL
      { ( ?uriDoc ?score2 )
              text:query  ( m:titulo "ocular" )
      }
    OPTIONAL
      { ( ?uriDoc ?score1 )
              text:query  ( m:descripcion "ocular" )
      }
    BIND(( coalesce(?score1, 0) + coalesce(?score2, 0) ) AS ?scoretot)
  }
ORDER BY DESC(?scoretot)
```

Figura A1
Consulta de la necesidad de información 01-2

```
PREFIX  text: <http://jena.apache.org/text#>
PREFIX  rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX  m:    <http://github.com/vpec/Recoreon/>
PREFIX  m_skos: <http://github.com/vpec/Recoreon/skos#>

SELECT  ?uriDoc
WHERE
  { ?uriDoc  m:tema  m_skos:cine ;
             m:tema  m_skos:ideología
    OPTIONAL
      { ( ?uriDoc ?score2 )
              text:query  ( m:titulo "cine" )
      }
    OPTIONAL
      { ( ?uriDoc ?score1 )
              text:query  ( m:descripcion "cine" )
      }
    BIND(( coalesce(?score1, 0) + coalesce(?score2, 0) ) AS ?scoretot)
  }
ORDER BY DESC(?scoretot)
```

Figura A2
Consulta de la necesidad de información 04-4

```
PREFIX  xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX  text: <http://jena.apache.org/text#>
PREFIX  m:    <http://github.com/vpec/Recoreon/>
PREFIX  m_skos: <http://github.com/vpec/Recoreon/skos#>

SELECT  ?uriDoc
WHERE
  { ?uriDoc  m:tema  m_skos:inteligencia
      { ?uriDoc  m:tema  m_skos:videojuego }
    UNION
      { ?uriDoc  m:tema  m_skos:personaje }
    ?uriDoc  m:fecha  ?fecha
    FILTER ( ( "2012"^^xsd:gYear <= ?fecha ) && ( ?fecha <= "2020"^^xsd:gYear ) )
    OPTIONAL
      { ( ?uriDoc ?score2 )
              text:query            ( m:titulo "inteligencia" )
      }
    OPTIONAL
      { ( ?uriDoc ?score1 )
              text:query            ( m:descripcion "inteligencia" )
      }
    BIND(( coalesce(?score1, 0) + coalesce(?score2, 0) ) AS ?scoretot)
  }
ORDER BY DESC(?scoretot)
```

Figura A3
Consulta de la necesidad de información 06-1

```
PREFIX  text: <http://jena.apache.org/text#>
PREFIX  rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX  m:    <http://github.com/vpec/Recoreon/>
PREFIX  m_skos: <http://github.com/vpec/Recoreon/skos#>

SELECT  ?uriDoc
WHERE
  { ?uriDoc  rdf:type  m:MasterThesis ;
             m:tema    m_skos:contaminación ;
             m:tema    m_skos:España
    OPTIONAL
      { ( ?uriDoc ?score2 )
              text:query  ( m:titulo "contaminación" )
      }
    OPTIONAL
      { ( ?uriDoc ?score1 )
              text:query  ( m:descripcion "contaminación" )
      }
    BIND(( coalesce(?score1, 0) + coalesce(?score2, 0) ) AS ?scoretot)
  }
ORDER BY DESC(?scoretot)
```

Figura A4
Consulta de la necesidad de información 11-4

```
PREFIX  text: <http://jena.apache.org/text#>
PREFIX  rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX  m:    <http://github.com/vpec/Recoreon/>
PREFIX  m_skos: <http://github.com/vpec/Recoreon/skos#>

SELECT  ?uriDoc ?score2
WHERE
  { ?uriDoc   rdf:type         m:BachelorThesis ;
              m:tema           m_skos:informática ;
              m:creador        ?persona .
    ?persona  m:nombrePersona  ?nombre
    FILTER regex(?nombre, "Javier")
    OPTIONAL
      { ( ?uriDoc ?score2 )
              text:query  ( m:titulo "informática" )
      }
    OPTIONAL
      { ( ?uriDoc ?score1 )
              text:query  ( m:descripcion "informática" )
      }
    BIND(( coalesce(?score1, 0) + coalesce(?score2, 0) ) AS ?scoretot)
  }
ORDER BY DESC(?scoretot)
```

Figura A5
Consulta de la necesidad de información 15-4