9ª práctica de laboratorio de SIW
“Información semántica II”

Motivación

En la práctica anterior se trabajó con microdatos y JSON-LD, un formato de serialización RDF. En la sesión de hoy ahondaremos en el ecosistema RDF mediante la creación de tripletas de forma manual, así como explorando distintas tecnologías que aspiran a generar datos estructurados a partir de texto libre. En esta segunda práctica el objetivo, además, es ir más allá de aportar metadatos y utilizar tecnologías semánticas para modelar conocimiento.

Textos de prueba

Para las distintas fases del ejercicio se **usarán dos** de los siguientes textos[1]:

- *Miles Davis was an American jazz musician.*
- *President Barack Obama and European Union leaders huddled in Washington amid growing fears over the future of the euro, which closed greater than 1.3 dollars.*
- *The New York Times reported that John McCarthy died. He invented the programming language LISP.*

Así como **otro de la siguiente lista**[2] que se asignará de manera aleatoria a cada estudiante:

1. *The longest river in Africa is the Nile, which is about 6,853 kilometers long.*
2. *The surface area of Mars is 144,798,500 square kilometers.*
3. *The Vatican is the smallest country in the world, measuring just 0.44 square kilometers.*
4. *The average lifespan of a housefly is 21 days.*
5. *Koalas sleep up to 22 hours a day.*
6. *The largest desert on Earth is the Sahara, which covers an area of 9.4 million square kilometers.*
7. *The color orange is named after the fruit, not the other way around.*
8. *Helen Keller was an American author, political activist, and lecturer.*
9. *The population of Japan is 127 million.*

[1] Ejemplos obtenidos de http://wit.istc.cnr.it/stlab-tools/fred/demo/

[2] Estos se generaron usando GPT-3 y empleando los anteriores como prompt.

10. *The Wright brothers, Orville and Wilbur, were two American brothers, inventors, and aviation pioneers who are generally credited with inventing and building the world's first successful airplane.*
11. *The Eiffel Tower was built in 1889.*
12. *The human brain is about 75% water.*
13. *There are more than 60,000 miles of blood vessels in the human body.*
14. *The first successful powered flight was made by the Wright brothers on December 17, 1903.*
15. *The highest mountain in the solar system is Olympus Mons on Mars, which is about 22 kilometers high.*
16. *The first product to have a barcode was Wrigley's gum.*

Descripción del ejercicio

Primera fase: creación manual de información estructurada

1. Obtener toda la información estructurada posible de cada texto usando al menos tipos de `schema.org` y describirla en lenguaje natural. **¡Atención!** No usar conocimiento ajeno al texto (p.ej. No incluir la fecha de defunción de Miles Davis o John McCarthy).

2. Modelar en RDF la información obtenida en el paso anterior usando el formato Turtle (mucho más sencillo para humanos que JSON-LD). Se recomienda el uso de Visual Studio Code con la extensión *Stardog RDF Grammars* que ofrece, entre otras cosas, sintaxis coloreada para para Turtle y SPARQL.

3. Validar la sintaxis de los datos RDF y luego verificar que una máquina puede extraer la semántica deseada (aplicar *Schema Markup Validator* sobre sobre la conversión automática de Turtle a JSON-LD).

Elegid uno de los ejemplos para que lo usemos en clase.

Segunda fase: obtención automática información estructurada

Existen diversos servicios web que permiten obtener información estructurada (p.ej. en formato RDF) a partir de texto en lenguaje natural. En esta sesión probaremos los siguientes:

- Open Calais.
- DBpedia Spotlight.
- FRED.

¡Atención! Aunque todos los servicios pueden consumirse programáticamente para esta sesión podrán consumirse mediante sus respectivos formularios.

- Para cada servicio y cada texto es necesario obtener un documento RDF (resulta indiferente su formato) con la información extraída.
- Dicha información será traducida (mediante RDF Shape) a Turtle para su examen manual y a JSON-LD para su validación por medio de *Schema Markup Validator*.

Haremos una demostración práctica con el mismo ejemplo que hayáis elegido.

A la vista de las pruebas realizadas, responde a las siguientes preguntas de forma razonada y con algunos ejemplos:

1. ¿Qué ontologías usa cada servicio para "tipar" las instancias detectadas en el texto?
2. ¿Existe algún tipo de dichas ontologías que pudiera considerarse equivalente a otro tipo en Schema.org? Señala todos los casos que te hayas encontrado con cada servicio, indicando también (si fuera posible) qué propiedades de un tipo mapearían sobre propiedades del otro.
3. Reflexiona acerca de los motivos que pueden llevar a cada equipo de desarrolladores a producir una ontología propia. Investiga (someramente) sobre el problema de alineación de ontologías.
4. Utiliza el servicio sameAs.org para localizar equivalencias[3] para los tipos fundamentales de Open Calais, DBpedia Spotlight y FRED, así como los homólogos de Schema.org que detectaste. ¿Existe algún servicio para el que no se disponga de información en *sameAs*?

[3] La URI debe ser absoluta, no es lo mismo `schema.org/City` que `http://schema.org/City`…