

La Lógica, Los Agentes y la Web Semántica

Jacinto Dávila

Universidad de Los Andes, Venezuela

La Web es un enorme “repositorio” de información, de contribución abierta y acceso libre. La Web Semántica pretende ser un enorme repositorio de conocimiento. La distancia entre una y otra puede ser cubierta por un agente capaz de entender esa información, o inclusive datos crudos, y convertirla o convertirlos en conocimiento. Por ahora, esos agentes son seres humanos. Pero si dispusiéramos de una representación del conocimiento para la Web y en la Web que pudiera ser procesada por agentes artificiales, estaríamos a un paso de relevar o, al menos, de asistir a los humanos en el uso de la Web para acceder al conocimiento.

Siendo el medio de representación de conocimiento más conocido y más aplicado en la historia tecnológica de Occidente, no es de extrañar que la lógica, en algunas de sus formas computacionales, sea convocada para proveer esa representación del conocimiento para la Web.

Por razones cercanas, no debe extrañar tampoco que quisiéramos aprovechar el Libro del Prof. Kowalski, dedicado a elaborar un modelo de agente artificial basado e inspirado sobre la lógica, para completar la conexión con la Web Semántica.

La Web Semántica se convirtió en proyecto tecnológico por iniciativa del mismo creador de la Web: Sir Tim Berners-Lee. El objetivo de ese proyecto es definir el lenguaje y las estrategias para dotar a la Web tradicional de representaciones del conocimiento procesables por máquinas.

El primer paso en esa dirección fue la definición de una estrategia de **anotación** de textos (**marcado** o **etiquetado**, también se le dice) similar al lenguaje base de la Web: HTML, pero que sirviera para crear nuevos lenguajes con los cuáles se pudieran etiquetar (o anotar) los textos con referencias **significativas** en términos del uso humano del lenguaje. Esa estrategia, casi un lenguaje, es **XML**.

XML, sin embargo, *es solamente una notación*. A diferencia de HTML, un

lenguaje que se le parece mucho, las etiquetas en XML no han sido pre-definidas para causar cierto efecto operativo en la máquina. De hecho, no han sido pre-definidas para nada. El propio usuario, creador de una “página XML”, agrega sus propias etiquetas (tal como haría con una página Web, si estuviera usando HTML). Más aún, se pueden combinar ambas notaciones.

Así por ejemplo, en lugar de la tradicional estructura y etiquetas HTML de una *hiper*-página web:

```
<html>
<body>
Este documento describe a la Web Semántica.
</body>
</html>
```

una página en la Web Semántica podría incluir otras etiquetas “más significativas” para las máquinas:

```
<html>
<body>

<resumen>
Este documento describe a la <tema>Web
Semántica</tema>.
</resumen>

</body>
</html>
```

con etiquetas como <tema> que pueden ser aprovechadas por un agente que “entienda” su sentido para separar fácilmente, a partir de ese texto, la respuesta a la pregunta “¿de qué trata este texto?” (u otras consultas como “Hazme un resumen de esta página!”).

Pero XML no es un lenguaje. Es solamente una notación (parte de la sintaxis de un lenguaje). Para crear un lenguaje a partir de XML tenemos que disponer de una forma de proveer a esa notación de semántica.

El primer paso en esa dirección es la definición de *las cosas de las que puede hablar ese lenguaje*. Esto es, prestando el término de los filósofos, una

ontología. Una especificación de cuáles cosas (u otros “conceptos”) se podrían aplicar allá afuera en el mundo y del cómo se les puede nombrar o referir en el lenguaje.

Como quiera que los lenguajes los crea una comunidad (y los re-crea constantemente), los ingenieros de la Web entendieron, mas bien temprano, que su trabajo antes de ser definir una ontología, era proveer a sus usuarios de los medios para definir sus propias ontologías y lenguajes, quizás a partir de una ontología básica y algunos medios para extenderlas. Esa ontología básica y esos medios para extenderlas constituyen, en si mismas, los llamados **lenguajes ontológicos** (un nombre más bien desafortunado, porque parece sugerir que hay lenguajes que no tienen ontología asociada, una confusión muy peligrosa como podrá dar fe cualquier filósofo de la lógica).

Así surge el proyecto por los lenguajes ontológicos que ha creado una explosión de interés y esfuerzos de investigación sobre la Web Semántica. En este texto vamos a mencionar solamente a 2 de ellos, RDF y OWL. Este último muy cercano a una forma de lógica un tanto distinta a la que se estudia en el libro del Prof. Kowalski.

RDF es un lenguaje, basado en XML, dotado con una semántica “para hablar de la web”. En RDF se supone que “hay un mundo” en el que existen **recursos** y **relaciones entre recursos**, y el lenguaje se dispone para que un agente **declare hechos** que involucren esas relaciones y esos recursos.

Por ejemplo, la declaración del hecho:

Jacinto Dávila tiene una página web en
<http://webdelprofesor.ula.ve/ingenieria/jacinto>

sería plasmado en así en RDF (recuerde que esta es una notación para la máquina, difícil de leer para un humano)¹.

```
<?xml version="1.0" encoding="UTF-16"?>
<rdf:RDF
```

¹ Siguiendo los ejemplos de “A semantic Web Primer” de Grigoris Antoniou y Frank van Harmelen. MIT Press. ISBN 0-262-012110-3. El espacio de nombres en midominio es hipotético. NO lo usen.

```
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-  
syntax-ns#"   
xmlns:midominio="http://www.midominio.org/mi-  
-rdf-ns">  
  
<rdf:Description  
  rdf:about="http://webdelprofesor.ula.ve/inge-  
niera/jacinto">  
  
  <midominio:posee-web>  
    Jacinto Dávila  
  </midominio:posee-web>  
</rdf:Description>  
  
</rdf:RDF>
```

El lector, con un poco de tiempo, podrá identificar cuáles secciones corresponde a terminología predefinida y cuáles no.

RDF, un lenguaje completo, sin embargo está atado a hablar de la web (recursos – relaciones - objetos o **tripletras** RDF como también se les conoce).

Quienes quieren ir más allá de esta semántica (útil para que las máquinas “entiendan” como ubicar y gestionar recursos Web) tendrían que probar con un lenguaje de propósito más general.

OWL es uno de esos lenguajes que, además, está “impregnado” de una forma particular de lógica llamada lógica descriptiva². Las lógicas descriptivas son un desarrollo más bien reciente dirigido a lenguajes que se prestasen para el tratamiento computacional, incluso si eso implicaba restringirlos para que ciertas declaraciones (problemáticas por su complejidad computacional) no se pudieran “decir” en ellos.

Existen varios de esos lenguajes. Incluso se ha adoptado un mecanismo de clasificación de esos lenguajes que indica su nivel de expresividad (vs tractabilidad).

2 Hay varios otros desarrollos intermedios entre RDF y OWL, como RDF-Scheme y DAML-OIL. No se trata de

Las lógicas descriptivas construyen sus discursos (o teorías) sobre una pocas relaciones binarias. Particularmente sobre la relación “es un” (o “es una”, normalmente la misma relación puesto que el Inglés no distingue).

Así que un documento OWL es, básicamente, un ARBOL taxonómico que explica que objetos son de qué tipo y con cuáles atributos o propiedades. Una jerarquía de clases equivalente a las que se emplean en la Orientación por Objetos, OO pero, en muchos casos, incorporando otros elementos en la formalización de la jerarquía (como restricciones de integridad, al estilo de las Bases de Datos).

No queremos, en estas notas complementarias del curso de lógica y agentes, contrariar (más) el esfuerzo del Prof. Kowalski, en el texto guía, de explicar un modelo de agentes basado en lógica, sin emplear ninguna lógica formal para ello. Así que no mostraremos más detalles de la sintaxis de los lenguajes ontológicos.

Lo que si queremos es mostrar 2 ejemplos de modelos de agentes que implementan otras dos nociones que se han vuelto fundamentales en la Web: Los **metadatos** y el **razonamiento no monótono**.

Una de las primeras aplicaciones naturales de la tecnología de la Web Semántica es la búsqueda de documentos. Con recursos como repositorios RDF y OWL en la Web, uno puede imaginar un agente que nos ayude a ubicar documentos a partir del tipo de conocimiento que contienen (y que buscamos). Por ejemplo, para responder preguntas como ¿Quienes han escrito sobre el legado epistolar del Libertador de Venezuela?, uno puede pensar³ en un agente con una META como esta:

si me preguntan ¿Quien Opina sobre Tema? y
Quien es una variante gramatical de “Quienes” y
Opina es una variante semántica de “opinar” y
el análisis del Tema arroja estos Descriptores **entonces**
Rastreo la Web buscando todos los Autores de documentos
con esos Descriptores.

3 Agradecemos a Icaro Alzuru por motivar este ejemplo con su trabajo de maestría en [“Alejandría Inteligente: Un experimento en la Web Semántica con un sistema de gestión de documentos”](#).

Esta meta requerirá, desde luego, mecanismos de soporte (que podrían convertirse en creencias del agente) para realizar el análisis del Tema que produce los Descriptores y el Rastreo de la Web.

En la tarea de análisis del tema, sería muy útil contar con un mecanismo que le permita al agente, en el caso particular de nuestra pregunta, traducir “legado epistolar” en todos sus sinónimos e hipónimos. Así el agente sabría que quien quiera que declare haber escrito sobre las “cartas” del Libertador es un autor a revisar. Algo similar habría que hacer con “Libertador de Venezuela” para que la máquina entienda que se trata de Simón Bolívar. Esto último requeriría, desde luego, un mecanismo adecuado para el caso venezolano-colombiano-ecuatoriano-peruano-boliviano.

Esta clase de relaciones de sinonimia, tanto universales como locales, son posibles con sistemas como WordNET, una especie de diccionario y tesoro terminológico automático. WordNET contiene una ontología que le permite relacionar los términos. Desafortunadamente, esa ontología no está escrita en ninguno de los lenguajes ontológicos de la Web Semántica, pero aún así puede ser útil.

La segunda tarea es también muy interesante. Requiere que nuestro agente disponga de medios para identificar entre la enorme cantidad de documentos contenidos en la Web, los autores y los descriptores de esos documentos. Esa información que refiere el contenido y otros atributos de los datos es conocida como **METADATA** y es fundamental para que las máquinas puedan contribuir a la gestión de los repositorios de conocimiento.

Hay muchos esfuerzos en la dirección de proveernos de un ***standard de metadatos*** en la forma de una ontología que explique que son y que contienen los documentos formales. El *Dublin Core* es quizás el más avanzado de esos esfuerzos.

Podemos ver, en ese primer ejemplo de agente para la Web Semántica, como se incorporaría el manejo de significado en un agente que atienda consultas en la Web.

Hay, sin embargo, un siguiente nivel de complejidad en el manejo de significados para atender consultas. Algo que es bien conocido en Bases de

Datos desde hace años: El manejo de datos temporales y el **razonamiento no monótono**.

Se habla de razonamiento no monótono a aquel que “cambia de opinión”. Típicamente, un agente, con cierto cúmulo de creencias, alcanza ciertas conclusiones, pero si esas creencias cambian (debido quizás, a un cambio en el mundo), las conclusiones también podrían cambiar.

Los lectores del libro del Profesor podrán asociar esa forma de razonar con esos lenguajes lógicos que estudiaron en el capítulo sobre el cambiante mundo (lógicas modales, el cálculo de situaciones y el cálculo de eventos).

Permítannos, sin embargo, un último ejemplo con un agente que implementa una forma de razonamiento no monótono indispensable para la aplicación a su cargo:

Imaginen un agente que maneja un repositorio de información catastral⁴ elemental. Entre muchos otros tipos de consulta, este agente debe poder explicar quienes son los propietarios de los terrenos en un área dada.

La meta de mantenimiento y creencia correspondiente para este agente es simple:

si me solicitan los Propietarios de terrenos en un Área **entonces** reportar Propietarios del Área.

Para reportar Propietarios del Área **haga**
transforme la descripción del Área en un Área Geográfica y
Los Propietarios **son** los sujetos para
quienes se cumple (ahora) que poseen propiedades
contenidas en el Área Geográfica.

La última sub-meta (“Los Propietarios..”) es una consulta que se podría responder con una versión extendida del cálculo de situaciones o el de

⁴ Agradecemos a Nelcy Patricia Piña por motivar este ejemplo con su trabajo de maestría en [“Una Ontología para Manejo de Información Catastral”](#).

eventos adecuada para lidiar con propiedades y áreas geográficas. Lo importante en este caso es que si a este agente se le hace la consulta en un momento dado, su respuesta bien puede variar respecto a otro momento si han ocurrido eventos que cambien las relaciones propiedad-propietario en ese área geográfica.

Desde luego, este agente tendría que operar con conocimiento que le permita asociar cualquier descripción de un área (por ejemplo, en términos políticos: parroquia, municipio, país, etc) con un conjunto de coordenadas *standard*, por ejemplo. Más importante aún, el agente requeriría de un registro sistemático de los eventos de compra y venta de propiedades que incluya los detalles de cada propietario y propiedad (incluyendo la ubicación de cada una, desde luego). Todo este conocimiento bien puede estar almacenado en un repositorio asociado con una ontología de registro catastral (por ejemplo, un archivo, con marcaciones en OWL específicas para el problema).

Esos dos ejemplos, informales y muy simples, pueden servir para ilustrar la extraordinaria riqueza de posibilidades en este encuentro entre la Web, la lógica y los agentes.

Copyright © Jacinto Dávila. El Autor se reserva el derecho de proclamarse autor de este texto y asume la responsabilidad por las consecuencias. El texto puede distribuirse sin ninguna otra restricción implícita o explícita. Noten que los autores citados o referidos mantienen íntegramente sus derechos.

Fin del documento. Versión 2.0.