La Web Semántica como plataforma para sistemas de recomendación

Guido Zuccarelli

October 9, 2014

Abstract

1

1 Introducción

1.1 Evaluaciones entrecruzadas

Definiremos a una reseña (de aquí en adelante review) como la forma que tiene un usuario de transmitir mediante el uso de la computadora, su grado de satisfacción con respecto a un ítem.
Estará conformado mediante una serie de atributos m'inimos necesarios y otros que ser'an que si bien no son indispensables, enriquecen no sólo su valor intrínseco, sino también su utilidad para el contexto que será planteado.
Llamaremos ítem, a un elemento que es aprovechado por las personas, dicho ítem puede ser, un objeto, un servicio, una idea, un programa, etc. Por ejemplo, una pelicula.
Este último debería poder ser identificable, para ello, será necesario como m'inimo un atributo (o un conjunto) que cumpla esa función, si tomamos como ejemplo un libro, sólo disponer del atributo ISBN, será suficiente apra identificarlo.
El review es una expresión de la persona que utilizó el ítem que refleja su grado de conformidad con el mismo, con el objetivo de informarlo a otras personas. Por lo que entonces, el atributo mínimo será aquel que refleje este sentimiento del usuario hacia el ítem, el mismo puede ser un texto explicativo (por ejemplo Ün pel'icula hermosa, pero me parecieron flojos lso actores¨) o un valor num'erico dentro de un rango determinado (7 en escala de 1 a 10).
A partir del atributo base, existen muchos otros atributos que, bien construidos, aportan mucha riqueza al aprovechamiento del review. Por ejemplo, el autor, que será un atributo que identifica a la persona creadora del review. Otro caso es el de la fecha.

1.2 La web sem'antica

En sus comienzos en los 90, la web podía verse como un conjunto de sitios web que ofrecían una colección de documentos web, con el objetivo de comunicar información a los usuarios.
Con el correr de los años, múltiples tecnologías se fueron implementando y permitieron el desarrollo de una web mucho más grande y aprovechable

1.3 RDF y OWL

1.4 Reviews en la web sem'antica

1.5 Linked data cloud

Con el fin de crear una aplicación que satisfaga los requerimientos mensionados anteriormente, se debe encontrar un procedimiento que incluya desde obtener los datos relevantes de la web hasta llevarlos a un estado que permita una explotación satisfactoria.
El procedimiento debería incluir los siguientes pasos
Selección de vocabularios
Tanto la recolección como la publicación de datos en la web semántica involucra la elección del o de los vocabulario/s que mejor modelen el dominio de problema, con la excepción que para su publicación existe la posibilidad de desarrollar uno propio que se ajuste correctamente en caso de que no se encuentre uno existente. Seleccionar un vocabulario implica analizar varios aspectos del mismo, no sólo su definición e implementación, sino también el uso práctico dado por sus usuarios.
En primer lugar se debe comprobar que los nombres de las propiedades que posee sean correctamente autoexplicativas. Supongamos por ejemplo que existe un ítem con un rating agregado modelado por una ontología que posee las siguientes propiedades:
=minRating
=ratingValue
=countRating
Las dos últimas propiedades resultan facilmente identificables, ratingValue se trata del promedio de puntaje, y ratingCount la cantidad de puntajes que le fueron otorgados, pero la propiedad minRating podría generar distintas formas de interpretación, alguien podría suponer que se trata del valor mínimo que fue adquirido por un usuario, o el valor mínimo que un usuario puede otorgar. Y muchas veces la documentación de la ontología (si es que existe) no es suficiente. Luego se deberá analizar si existen las propiedades para cubrir las necesidades mínimas de los casos de uso.
Y por último se debería intentar buscar ejemplos reales que muestren el uso que le dieron los usuarios a la ontología, para determinar qué propiedades están incorrectamente interpretadas o también para los casos donde las propiedades que se encuentren en desuso.
Con estas precauciones en mente se puede emprender la búsqueda, que podría tener como comienzo búsquedas en search engines. Existen dos buscadores específicos para esta tarea.

En la actualidad existen cuatro vocabularios que cumplen los requerimientos mínimos para modelar el dominio de problema planteado

RDF Review Vocabulary
También conocido como Review Ontology, es una de las ontologías más antiguas de review, que fue pensada para uso del lenguaje RDF y está definido bajo el namespace http://purl.org/stuff/rev .
Fue utilizado para la construcción Revyu , y sirvió como guía para otras ontologías.
Consta de tres clases y trece propiedades.
Clases
Comment: Un comentario sobre el review.
Feedback: Expresa la utilidad del review.
Review: El review mismo.

Propiedades
commenter: Especifica el usuario que realizó el comentario del review. Tiene dominio Feedback o Comment y rango foaf:Agent.
Actualmente se encuentra en desuso.

hasReview: Enlaza el ítem evaluado con el review. Su dominio es rdfs:Resource (siendo ésta la clase del ítem evaluado) y su rango es Review. Es una de las propiedades principales.

hasComment: Idem anterior pero con el comentario en lugar del review.
Se encuentra actualmente en desuso.

hasFeedback: Idem anterior pero con feedback en lugar de comentario.
Se encuentra actualmente en desuso.

maxRating: Establece el puntaje máximo que es posible otorgar por un usuario a travez de la propiedad rating. Tiene como dominio Review y como rango Literal siendo este último un número positivo. Su ausencia en un review asume su valor por defecto (5). Por lo que si bien no es indispensable que esté, se deberá respetar la convención ala hora de generar el rating.

minRating: De la misma forma que el anterior, sólo que establece el puntaje mínimo y su valor por defecto es (1).

positiveVotes: Se refiere a la cantidad de votos positivos que tuvo el review, otorgados por usuarios que lo leyeron y lo encontraron útil. SU domino es Review y su rango Literal siendo este último un número positivo.
Se encuentra actualmente en desuso.

rating: Una de las propiedades principales, indica el valor numérico otorgado por el creador del review sobre el ítem evaluado. Su domino es Review y su rango es Literal siendo este último un número entre los valores de minRating y maxRating.

reviewer: Especifica el usuario que realizó el review. Tiene dominio Review y rango foaf:Person.

text: Otra de las propiedades principales, define el texto que describe el sentimento del usuario hacia el ítem. Tiene como domino Review y como rango Literal.

totalVotes: Exactamente igual a positiveVotes.

title: El título del review . Tiene dominio Review y rango Literal. Subclase de dc:title
No tiene demasiada utilidad para el caso de estudio y además se encuentra en desuso.

type: Enuncia el tipo de ítem que clasifica taxonómicamente al ítem evaluado. Su domino es rdfs:Resource (siendo ésta la clase del ítem evaluado) y su

rango no se encuentra especificado.
Es una propiedad muy útil pero actualmente se encuentra en desuso.

date: Si bien no está definida dentro del vocabulario, es correcto utilizar http://purl.org/dc/terms/date, implica la fecha en la que se realizó el review.

hReview

Como se mensionó anteriormente, está establecido por convención, que los microformatos son una forma de publicar información en la web semántica, pero al no disponer de namespaces, no pueden ser representados por ninguna ontología o ningún otro lenguaje de la misma.
Al no poder utilizar ontologías que modelen reviews surgió la necesidad de crear un estándar específico para embeberlos dentro de HTML utilizando microformatos.
Dicho estándar se encuentra actualmente en la versión 0.4 y propone el uso de 10 propiedades, que se supone, deberían ser suficientes para cubrir todas las necesidades a la hora de generar un review.

summary (opcional): Puede ser el título o nombre del review, o es posible también hacer una pequeña sinopsis del mismo. Se encuentra en desuso.

type (opcional): Representa el tipo de ítem evaluado, pero se encuentra acotado a alguno de estos product — business — event — person — place — website — url .
También está en desuso.

item: Esta propiedad enuncia toda la información que se crea necesaria para identificar al ítem, mínimamente los atributos nombre, url y foto.
Para lograr bajo una propiedad cubrir todos los atributos, el rango de la misma debería ser un hCard, que luego contendrá las propiedades mínimas necesarias: fn, url, photo y cualquier otra que se quiera adicional.

reviewer (opcional): Al igual que ítem, indica todo lo necesario para identificar a la persona autora del review, para lo cual también deberá representarse con un hCard.

dtreviewed (opcional): Se refiere a la fecha en la que fue creado el review.

rating: El valor numérico con el cual el usuario expresa su satisfacción con el ítem, y está formado por un entero con un solo decimal de precisión, que se encuentra dentro del rango 1.0 a 5.0. Dicho rango puede ser alterado con la presencia de las propiedades worst y best que restringen el valor mínimo y máximo respectivamente.

descripción (opcional): Establece el valor textual con el cual el usuario expresa su satisfacción con el ítem, creando una sinopsis detallada del mismo.

tags (opcional): Una etiqueta intenta establecer una idea acerque de qué se trata el contenido en una sola palabra para una rápida identificación o para

mejorar las búsquedas.

permalink (opcional): Genera una URI que identificará al review creándole una especie de ID, que será útil para los casos donde se podría repetir la publicación del mismo.

license (opcional): Expresa la licencia del review.

El indicador "(opcional)" se refiere a si es indispensable para conformar un hReview o no, de manera tal que si no se encuentra una propiedad que no está marcada como opcional no podrá ser considerado un hReview.
Cabe destacar que muchas de las propiedades opcionales, podrían ser necesarias para el caso de estudio.

El problema con este vocabulario surge a la hora de trabajar con la información obtenida, que no puede ser representada por ningún otro lenguaje de la web semántica, por lo que se vió la necesidad de mapear las propiedades de hReview, a otro vocabulario que sí pueda.
Esto llevó a que en la web donde definen hReview lo consideren compatible con RDF Review Vocabulary, por lo que en Noviembre de 2007 se creó una herramienta que transforma de uno al otro hreview2rdfxml.xsl, pero existe un problema con el rango de algunas propiedades, por ejemplo reviewer (propiedad homónima en ambos vocabularios, pero una con rango hCard y otra con rango foaf:Person).

En general si bien este vocabulario bien utilizado puede ser efectivo, resulta poco flexible y complciado de manejar por parte de quien quiera explotarlos, el motivo por el cual se ha vuelto muy popular es su facilidad para generarlos, dado que microformatos es un lenguaje muy sensillo y cualquier persona con un relativamente mínimo ocnocimiento de HTML puede generar sin problemas un hReview, teniendo además herramientas online como opción, que generan el código a travez de un formulario. Como es el caso de http://microformats.org/code/hreview/creator.

RDF Data Vocabulary
En Mayo de 2009 Google anuncia la introdución de los llamados "Google Rich Snippets", estos fragmentos enriquecidos son una convención de etiquetas (con soporte para RDFa-lite y microdata) que permitían agregar información útil a los SERP del buscador de Google. De manera tal que los datos que contenían estos fragmentos, recibían un tratamento especial.
Sin embargo en el anuncio, Google revela que el soporte se limitó al uso de las clases y propiedades del vocabulario definido en una página notoriamente improvisada llamada http://rdf.datavocabulary.org/ .
En ella se establecían modelos de clases para varios tipos de ítems, tales como Persona, Organización o Producto y también para los Reviews.
La clase Review, quedó definida bajo el namespace http://data-vocabulary.org/Review e incluía las siguientes propiedades:
itemreviewed: Enlace al ítem que está siendo evaluado.

rating: El valor numérico con el cual el usuario expresa su satisfacción con el ítem, tiene como rango un valor numérico bajo la clase xsd:string o Rating,

y los valores posibles se encuentran en escala de 1 a 5, pudiendo la misma ser alterada con la presencia de las propiedades worst y best que restringen el valor mínimo y máximo respectivamente.

reviewer: El autor del review, su rango es dvocab:Person o xsd:string.

dtreviewed: La fecha en la que se realizó el review, no contiene un rango específico pero aclara que debe respetar el formato ISO para las fechas.

description: El cuerpo del review que representa el valor textual de satisfacción del usuario con el ítem.

summary: Un resumen corto del review.

Se puede notar la excesiva similitud de este vocabulario con hReview, queda claro que no hubo una intención de innovar algo, sino de representar el hReview en microdatos, probablemente por el apuro en la que data vocabulary fue creado.
Vale aclarar que limitar las clases y propiedades posibles en RDFa es básicamente hacerle perder el sentido al lenguaje (la decentralización de los vocabularios sobre los términos) haciendo que el lenguaje se utilice como si fuese microformatos, pero perdiendo su valor más improtante (la simplicidad) de manera tal que tomó la inflexibilidad de microformatos y la complejidad de RDFa.
Más adelante los Google Rich Snippets incluyeron también soporte para microformatos (lo que incluía hReview).

Schema.org
Recolección y extracción de los datos
Como se mencionó antes, la web contiene grandes cantidades de documentos publicados con información semántica. Pero la tarea de encontrarlos, con el agregado de que sólo una pequeña porción de ellos será relevante para los requerimientos no es trivial en lo absoluto debido a la inmensidad del universo en el que se encuentran. La forma de llevar a cabo este objetivo está atada al hardware disponible, tanto para almacenar los datos, como para el tiempo que va a emplear la ejecución de esta tarea.
Dado que las bases de datos semánticas sólo almacenan información en forma de tripletas o cuadrupletas, los documentos encontrados deberán someterse a un proceso de extracción que seleccione las sentencias HTML y las convierta a alguno de los lenguajes que soportan tripletas o cuadrupletas. Para esto existen múltiples herramientas.
Una vez transformados los documentos HTML a documentos semánticos puede construirse la base de datos semántica con la información recolectada.

La forma de llevar a cabo este objetivo no es única, y dependerá de varios aspectos:
Recursos de hardware disponibles
Cantidad y calidad de la información requerida
El grado de atemporalidad mínima tolerable en los datos
El primer paso para realizar la recolección es intentar responder la siguiente pregunta: ¿Dónde encuentro la informanción?
Una vez seleccionados los vocabularios, se necesitará obtener fuentes de datos

que contengan sus datos publicados en esos vocabularios.
Para lograrlo se podrá utilizar como punto de partida:
Sitios indexadores: Son algunos sitios que disponen de un dataset muy grande procesado con documentos indexados, que ofrecen consultar dicho dataset mediante servicios web. Generalmente proveen una API donde se pueden consutlar los datos mediante distintos grados de flexibilidad.
Sindice, LOD cloud cache y UriBurner son algunos ejemplos de estos sitios. Se puede utilizar entocnes, los servicios web a fin de obtener una lista de URL que se cree que tendrán la información necesaria.
Sitios autoritativos: Son sitios conocidos que generan información relevante y la publican en las ontologías y vocabularios seleccionados. Ejemplos aplicados al caso de estudio podrían ser IMDB (que publica reviews de películas bajo el vocabulario schema), o Rottentomatoes (que hace lo mismo pero no solo con películas). Se podrán utilizar entonces estos sitios como punto base para un posible crawling.
Catálogos de enpoints: Son catálogos provistos por algunos sitios que mantienen una lista actualizada de SPARQL endpoints junto con el estado de disponibilidad en el que se encuentran. Por ejemplo los sitios http://labs.mondeca.com/sparqlendpointsstatus/ y http://www.w3.org/wiki/SparqlEndpoints que este último además provee detalles sobre la información de los mismos.
Consultar estos sitios entonces generará una lista de endpoints SPARQL que se podrá utilizar para consultar la información necesaria.
Volcados de datos: Son datasets muy grandes que fueron el resultado de un web crawling, los cuales están disponibles para su descarga y se pueden utilizar para procesarlos y obtener los datos requeridos. El más importante es http://challenge.semanticweb.org/2014/ .

Como sabemos los reviews son creados por usuarios que en su mayoría no están familiarizados con el desarrollo de una aplicación, esto causa que una parte muy significativa del contenido publicado no esté de la forma adecuada para ser procesado. La falta de calidad en el contenido publicado puede deberse tanto a errores por parte del usuario como por parte del publicador.
El publicador deberá asegurarse que los datos respeten rigurosamente las ontologías en las cuales se publican.
El sitio web en el cual el usuario se encuentre realizando el review deberá guiarlo en todo lo posible para lograr que la evaluación quede en un formato adecuado.
Aunque también existen muchos problemas que no dependen del sitio web, generalmente errores semánticos de calidad, donde lo redactado esta hecho de forma inconsistente o insuficiente.
Este paso entonces tiene por objetivo encontrar todos los problemas de calidad que pueda haber en el dataset que acaba de ser descargado y extraído y que generen inconvenientes para una posterior integración/explotación.
Curado de los Datos
En el paso anterior se describieron los problemas en la calidad del dataset que tendrán un impacto en la aplicación resultante, o que, podrían limitar o imposibilitar la realización de la misma. Dichos problemas pueden ser muchos, y algunos muy difíciles de solucionar, será parte entonces del proceso, reconocer aquellos que su resolución sea viable y además encontrar la forma de implementarla.
Cabe destacar que el proceso de evaluar el dataset en búsqueda de problemas y

implementar soluciones es iterativo, debido a que una mejora puede conllevar a nuevos problemas.

Integración de los datos

Los datos recolectados fueron generados por distintos usuarios, en múltiples sitios y bajo distintas ontologías y estándares. Estos aún curados, necesitan un último paso para poder realizar una explotación, y es la integración.

Este gran y dificultoso proceso abarca cualquier operación que intente dar una visión más unificada de la información. Este proceso puede tener distintos niveles y aspectos a integrar, como podría ser por ejemplo, unificar las ontologías de los reviews, o bien en una nueva ontología de review, o bien en una ya existente, para lograr tener información semánticamente más parecida. Ya que este proceso puede ser muy dificultoso, habrá que ver en base a los requerimientos, qué aspectos de los datos se integrarán.

Una vez realizada la integración mínima necesaria par also requerimientos, el dataset ya estará listo para su explotación.

Recolección de los datos

Objetivo

Como se indicó anteriormente la información semántica que va a ser necesaria para construir se encuentra en la web en forma de documentos HTML que tienen la particularidad de ser muy efímeros, de manera tal que un documento que poseía datos relevantes a la fecha, puede al día siguiente, o dejar de estar disponible on-line, o haber cambiado de forma tal que la información de éste ya no es relevante, o ya no la posee.

En [] sección 4.2 Challenges for the selection of data sources se generó una estadística de este caso, donde se estableció que en promedio 62% de los documentos entoncontrados, continuaban on-line luego de un año, y de estos, sólo un 56% aún poseían datos relevantes.

Si bien armar un dataset con sólo información extraída de los documentos sin descargar estos últimos es posible, la situación anterior genera la necesidad de mantenerlos en una copia local para evitar una posible pérdida de los mismos.

El objetivo entonces será armar un repositorio local con los documentos on-line descargados que se cree que tienen la información necesaria.

Estrategia