

Una herramienta visual para la búsqueda semántica RDF

Joanna Alvarado-Uribe¹, Miguel González-Mendoza¹, Neil Hernández-Gress¹,
Carlos Eli Escobar-Ruiz² y Marcos Uriel Hernández-Camacho²

¹Tecnológico de Monterrey, Campus Estado de México,
México

²Universidad Politécnica de Chiapas, Chiapas,
México

joanna.1890@gmail.com;{mgonza,ngress}@itesm.mx;carlosecobar@
portaltuxtla.com;uriel.hdzc@gmail.com
<http://www.itesm.mx>
<http://www.upchiapas.edu.mx>

Resumen. La cantidad de información que uno o más usuarios de Internet generan para la Web Semántica está incrementando diariamente. Por esto, es necesario desarrollar herramientas que nos permitan mostrar esta información de una manera rápida, simple y fácil de entender. De acuerdo con esta premisa, hemos desarrollado una herramienta de visualización de datos semánticos, denominada DBPedia Search, capaz de: 1) consultar cualquier base de datos de tripletas que cuente con un *endpoint* de SPARQL y; 2) generar gráficos, mapas de calor y mapas de geolocalización de manera automática, con base en la información obtenida de la búsqueda realizada por el usuario. El objetivo principal es realizar una búsqueda y un análisis simplificados de los datos semánticos y presentarlos gráficamente.

Palabras clave: DBPedia search, visualización, *Endpoint* de SPARQL, tripletas.

1. Introducción

La Web Semántica es percibida como un área de investigación multidisciplinaria que combina campos científicos como la Inteligencia Artificial, Ciencias de la Información, Teoría de Algoritmo y de la Complejidad, Teoría de Base de datos, Redes de Computadoras, entre otros [1].

La Web Semántica se basa en la idea de agregar más semántica legible por la computadora a la información web a través de anotaciones escritas en *Resource Description Framework* (RDF) [2]. El modelo RDF se introdujo en 1999 como una recomendación del *World Wide Web Consortium* (W3C). Debido a esto,

la propuesta de la Web Semántica es la construcción de una infraestructura de semántica legible por la computadora para los datos en la Web [2].

Con base en la evolución del RDF, se están implementando en la red iniciativas mundiales tales como el *Open Directory Project*, *Dublin Core*, *Friend Of a Friend* (FOAF), *Simple Knowledge Organization System* (SKOS), *vCard Ontology*, y *Really Simple Syndication* (RSS) [2]. Este hecho es crucial para el desarrollo de la Web Semántica, porque RDF sigue los principios de diseño del W3C y algunas de las características principales de la Web Semántica como la interoperabilidad, extensibilidad, evolución y descentralización. Uno de los objetivos principales por el que el modelo RDF fue diseñado, es permitir que cualquier persona pueda hacer declaraciones sobre cualquier recurso. De esta manera, para la construcción de un modelo RDF únicamente es necesario disponer de un conjunto de recursos, básicamente cualquier cosa que tenga un *Universal Resource Identifier* (URI) [2]. Algunos ejemplos de recursos son: páginas web, imágenes, videos, computadoras, impresoras, etc. [3].

El lenguaje para representar los recursos está constituido por un conjunto de propiedades. Las descripciones de estas propiedades son enunciados estructurados en forma de tripletas sujeto-predicado-objeto o sujeto-propiedad-valor [2][4]. Mientras que el predicado y el objeto son recursos o cadenas, el sujeto y el objeto pueden ser objetos anónimos - también conocidos como *blank nodes* - [2]. Otra forma de explicar los componentes de las tripletas es: el sujeto es el recurso, el predicado es la característica que se describe y el objeto es el valor para esa característica [4]. Un aspecto interesante del modelo RDF es que el sujeto u objeto de una sentencia RDF puede ser otra declaración, esta característica es conocida como *reification* [2].

Gráficamente, el modelo RDF puede ser representado como un grafo de datos, Figura 1. La Figura 2 presenta parte de su codificación en RDF/XML [5].

Para trabajar con esta herramienta es necesario disponer de un *endpoint* de SPARQL de la base de datos de tripletas que se desea consultar. Un *endpoint* de SPARQL permite el procesamiento de consultas remotas [6].

En nuestra primera implementación, elegimos el *endpoint* de la versión en inglés de la ontología DBpedia [7]. La versión en inglés de la ontología DBpedia es parte del proyecto de DBpedia; este proyecto ha estado extrayendo información estructurada de Wikipedia en varios idiomas, como el inglés, el español, el japonés, entre otros; con la finalidad de generar información semántica disponible en la Web [7][8].

Este artículo está dividido en seis secciones. En la primera sección denominada Introducción, presentamos brevemente los temas que vamos a abordar en este documento. En la segunda sección llamada Trabajo relacionado, presentaremos algunas herramientas que trabajan con información semántica y/o *endpoints*

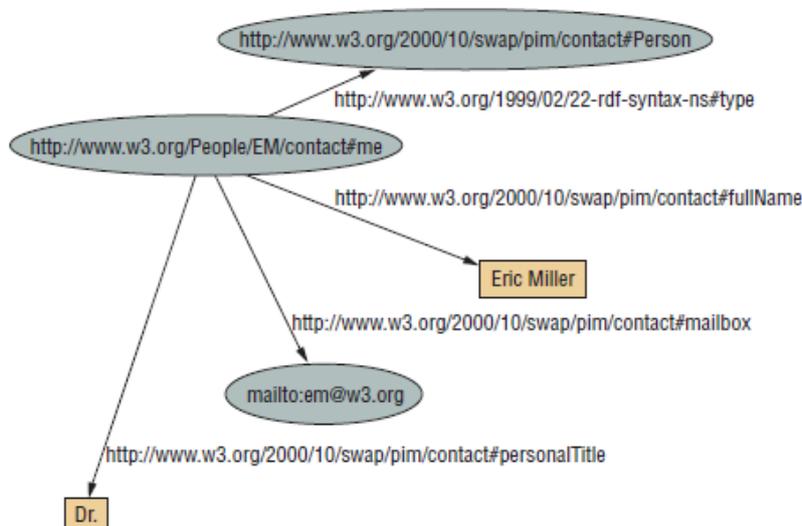


Fig. 1. Grafo de datos RDF, tomado de [5].

de SPARQL, y se mostrará una clasificación de las mismas de acuerdo con el motor de búsqueda que manejan. En la sección 3) Desarrollo y prototipo de DBPedia Search, presentaremos las fases de desarrollo de esta herramienta y el prototipo final; dentro de los aspectos que se abordarán están: tecnología utilizada, recopilación de datos, análisis de datos, entre otros. Para la sección 4) Experimentos y resultados, mostraremos el uso de la herramienta en diferentes Sistemas Operativos (S.O.) y explicaremos brevemente los resultados obtenidos en las pruebas de rendimiento. En la sección 5) Comparativo con otras herramientas, realizaremos un comparativo técnico y, de desarrollo y funcionamiento entre las herramientas presentadas en la Sección 2 y la nuestra. Y la última parte son 6) Conclusiones y trabajo futuro, en esta sección presentaremos nuestro punto de vista sobre la herramienta y describiremos algunas propuestas para mejorarla.

2. Trabajo relacionado

En esta sección introducimos como estado del arte, herramientas que también han abordado búsquedas semánticas. Para ello, nos centraremos en tres enfoques de búsqueda de las numerosas que hay, debido al impacto que tienen hacia nuestra herramienta, estos son:

1. Motores de búsqueda basados en forma. Estos motores se basan en formas complejas que toman ventaja de tener la información organizada en portales semánticos.

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-
ns#"
xmlns:contact="http://www.w3.org/2000/10/swap/pim/contact#">

  <contact:Person rdf:about="http://www.w3.org/People/EM/
  contact#me">
    <contact:fullName>Eric Miller</contact:fullName>
    <contact:mailbox rdf:resource="mailto:em@w3.org"/>
    <contact:personalTitle>Dr.</contact:personalTitle>
  </contact:Person>
</rdf:RDF>
```

Fig. 2. Parte de RDF en RDF/XML, tomado de [5].

2. Motores de búsqueda basados en palabra clave (como el descrito en este artículo). Estos motores se basan en una palabra o término dado, dando como resultado la visualización de información semántica.
3. Herramientas de pregunta/respuesta que utilizan datos semánticos. Estas herramientas permiten al usuario realizar una pregunta con el fin de extraer términos más específicos que les permitan buscar una respuesta directa en lugar de numerosos resultados.

La herramienta de búsqueda SHOE [9] muestra una serie de controles complejos en una forma. Esta forma permite al usuario construir consultas semánticas que se llevan a cabo a través de diferentes fuentes de información. Esta herramienta es un claro ejemplo del primer grupo, los motores de búsqueda basados en forma. La desventaja de este tipo de herramientas, es que el usuario necesita entender cómo trabajan las relaciones semánticas, para así poder construir una buena consulta desde la forma. La herramienta de búsqueda SHOE es impulsada por el lenguaje SHOE (*Simple HTML Ontology Extensions*), una alternativa a los estándares modernos como RDF y OML (*Ontology Markup Language*). Existen algunos ejemplos comerciales de esta categoría como Yummly [10], motor de búsqueda en la Web para comida, cocina y recetas; basado en datos de la Web Semántica.

TAP [10], construido sobre la interfaz de consulta GetData [11], permite al usuario buscar datos semánticos utilizando palabras clave. SemSearch [12] también introduce una interfaz en la que se teclean algunas palabras para realizar una búsqueda; esta herramienta fue construida para los usuarios denominados ‘usuarios ingenuos’, usuarios que no necesariamente conocen cómo está organizada la Web Semántica. Es importante destacar que este tipo de búsqueda (basada en palabras clave) es la que se realiza en la herramienta presentada en este artículo.

Evi [13], uno de los pocos productos comerciales basados en la Web Semántica, es la clara representación de una herramienta de pregunta/respuesta basada en datos semánticos. Fue desarrollado en Cambridge y se presenta como una aplicación móvil; utiliza el Procesamiento de Lenguaje Natural (PLN) y técnicas de búsquedas semánticas. AquaLog [14] es un ejemplo no comercial de esta categoría, es una solución portátil que puede ser adaptada a cualquier ontología dada, también utiliza tecnologías de PLN para formular tripletas ontológicas.

3. Desarrollo y prototipo de DBPedia Search

De acuerdo con la investigación realizada sobre la Web Semántica y el uso de la semántica con SPARQL, desarrollamos una herramienta cuyo objetivo principal es analizar los datos almacenados en las bases de datos de tripletas con el fin de realizar la clasificación de estos datos en categorías y de esta manera, construir gráficas de barras, mapas de calor y mapas de geolocalización, como resultado de la consulta realizada por el usuario. Las etapas de desarrollo y la construcción del prototipo se explicarán en 5 fases: tecnología utilizada, recopilación de datos, análisis de datos, visualización de las estadísticas y el prototipo.

Tecnología utilizada

Decidimos desarrollar la herramienta con PHP 5, por ser una tecnología de fácil instalación y porque es una tecnología en la que tenemos la experiencia suficiente para desarrollar aplicaciones para la Web. Se utilizó la librería ARC2 para conectar los *endpoints* con la herramienta, y la librería D3.js con JavaScript para permitir la visualización de los datos al usuario final.

A continuación, describiremos las librerías utilizadas:

- ARC2 es una librería de PHP 5.3 que funciona únicamente con triples semánticas (RDF), y un *endpoint* público utilizando SPARQL. También proporciona un almacenamiento en tripletas basado en MySQL con soporte para SPARQL [15].
- D3.js es una librería JavaScript que ayuda a manipular documentos basados en datos utilizando HTML, SVG y CSS. D3 combina componentes de visualización de gran alcance y un enfoque basado en datos para la manipulación DOM [16].

Recopilación de datos

Este es el primer paso para el desarrollo de la herramienta. En esta fase obtenemos la información que necesitamos para llevar a cabo el análisis (esta información será mencionada en las siguientes fases), como los tipos de datos. Los tipos de datos expresan la información contenida en las tripletas; por ejemplo, los tipos de datos en DBpedia son todas las categorías. Esta fase se realiza una única vez para cada *endpoint* de SPARQL.

Un aspecto relevante es que los datos están actualizados en todo momento, ya

que la herramienta trabaja directamente con el *endpoint* de SPARQL.

Análisis de datos

En este paso se analizan los datos obtenidos de la consulta realizada por el usuario, con el fin de encontrar una manera adecuada para mostrar la información resultante. Para cada consulta se mostrarán, en el mejor de los casos, cuatro elementos: una lista con los datos resultantes; a través de la información relacionada con los países, la herramienta mostrará un mapa de calor y geolocalización; y con la información en común, la herramienta construirá gráficas de barras.

El proceso completo se describe a continuación:

- Paso 1: encuentra el URI correcto.
Este paso revisa ¿cuál es el URI más utilizado? Esto es útil si manejamos una gran base de datos de tripletas y la base de datos tiene información repetida.
- Paso 2: lista de los datos resultantes.
En este paso se realiza una consulta sencilla para encontrar algunos ejemplos de datos que constituyen los resultados de la consulta.
- Paso 3: encontrar una propiedad relacionada con algún país, ciudad, estado o lugar.
La herramienta realiza una búsqueda entre las propiedades para encontrar una o más propiedades que contengan información sobre los países o lugares más específicos. Esto es útil cuando la información recopilada es sobre personas y sus nacionalidades, nombre del país de nacimiento, ubicaciones de empresas, entre otros aspectos.
En el caso de las ciudades, estados o lugares más específicos, buscamos su latitud y longitud para obtener una ubicación más exacta; y para los países, buscamos por sus nombres.
- Paso 4: conteo de datos para cada país o lugar.
Una vez que la herramienta ha finalizado el paso 3, esta fase encuentra la mejor propiedad para describir el país y/o lugar, y de acuerdo con esta propiedad se realiza el conteo de los datos en cada país y/o lugar para visualizar el mapa de calor y su geolocalización.
- Paso 5: obtención de propiedades comunes para la palabra buscada.
En el último paso la herramienta agrupa las propiedades comunes y, hace el conteo de los datos contenidos en estas propiedades para construir las gráficas de barras. Por ejemplo, si estamos buscando gente de México las propiedades comunes podrían ser Nombre, Apellido, Ciudad de nacimiento, Fecha de nacimiento, etc.

Es importante mencionar que las consultas, dentro de la herramienta, se realizan utilizando el lenguaje SPARQL, aunque para el usuario final, este hecho es transparente. Un ejemplo de un query en SPARQL, utilizado por la herramienta para esta fase, se muestra en la Figura 3.

Visualización de las estadísticas

En este paso se realiza un tratamiento de la información, en el que se analizan los tipos de datos en las tripletas para combinar los tipos de datos que tengan

```
SELECT ?res ?property ?value
WHERE {
    ?res pgn:tipo <http://mipagina.com/persona> .
    ?res ?property ?value
}
ORDER BY ?res
LIMIT 500
```

Fig. 3. Filtrar y mostrar la lista de resultados de la palabra clave “personas”.

el mismo nombre (aunque diferente URI).

Una vez que se realiza el tratamiento, por medio de la librería D3.js se visualizan los gráficos en Front-End. La herramienta construye un mapa de calor y geolocalización, y gráficas de barras para la visualización de la información resultante y además, presenta una lista de los resultados dentro de la búsqueda. El Front-End se presentará en la Sección de Experimentos y resultados.

Prototipo

Para explicar esta fase, construimos un diagrama de bloques para mostrar cómo se constituye el Back-End de la herramienta, el diagrama se muestra en la Figura 4.

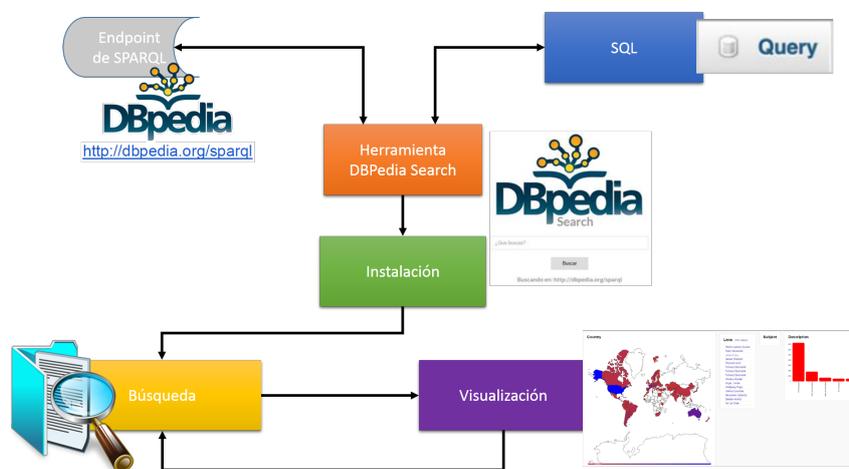


Fig. 4. Diagrama de bloques.

La Base de datos Relacional (SQL) y el *endpoint* de SPARQL son dos servicios independientes que están conectados a la herramienta. La Base de datos Relacional se utiliza para guardar información del *endpoint*; es decir,

información de las URIs y las propiedades encontradas. Esto con la finalidad de no realizar la fase de Análisis de datos cada vez que se realiza la misma consulta. Convirtiéndose en una pequeña caché que mejora la velocidad de la búsqueda y ahorra tiempo en la ejecución de la consulta. En el caso de que se agreguen, modifiquen o eliminen tripletas en la base de datos de tripletas (ontología), el usuario final tiene la seguridad de que la información que obtiene en su consulta está actualizada, ya que el programa se enlaza directamente con la base de datos de tripletas y obtiene todos los resultados en tiempo real.

La instalación de la herramienta sólo se realiza una vez, aunque se cambie de *endpoint*, lo único que se debe llevar a cabo es el borrado de los registros que se tienen almacenados en la Base de datos Relacional. Por lo que, la herramienta únicamente itera entre las actividades de búsqueda y visualización de la información consultada, como se muestra en la Figura 4. De esta manera, se forma un ciclo entre las fases de Análisis de datos y Visualización de las estadísticas.

4. Experimentos y resultados

Para llevar a cabo las pruebas de la herramienta, se insertó directamente la dirección del *endpoint* de la versión en inglés de la ontología DBpedia en la Base de datos Relacional; una vez que se realizaron varias consultas, decidimos cambiar el *endpoint* para trabajar con otras bases de datos de tripletas, con la finalidad de verificar que la herramienta funcione correctamente con diferentes *endpoints*.

En esta sección, únicamente mostramos tres capturas de pantalla de nuestra herramienta en diferentes Sistemas Operativos; una impresión de pantalla por Sistema Operativo.

Windows 8.1

En este Sistema Operativo fue instalado el *endpoint* de SPARQL de la versión en español de la DBpedia [8], como se muestra en la Figura 5.

Endpoint: <http://es.dbpedia.org/sparql>

Ubuntu 14.04

En este Sistema Operativo fue instalado el *endpoint* de SPARQL de la Biblioteca del Congreso Nacional de Chile/BCN [17], como se muestra en la Figura 6.

Endpoint: <http://datos.bcn.cl/sparql>

Mac OS

En este Sistema Operativo fue instalado el *endpoint* de SPARQL de *Serendipity* [18], como se muestra en la Figura 7.

Endpoint: <http://serendipity.utpl.edu.ec/lod/sparql>

Para obtener el rendimiento de nuestra herramienta entre los Sistemas Operativos, utilizamos el mismo endpoint (versión en inglés de la ontología DBpedia) y realizamos las mismas consultas en cada uno. Los resultados del Sistema Operativo Mac OS se indican en la Tabla 1, los del S.O. Windows en la Tabla 2 y los del S.O. Linux en la Tabla 3. [h]



Fig. 5. Búsqueda en español.

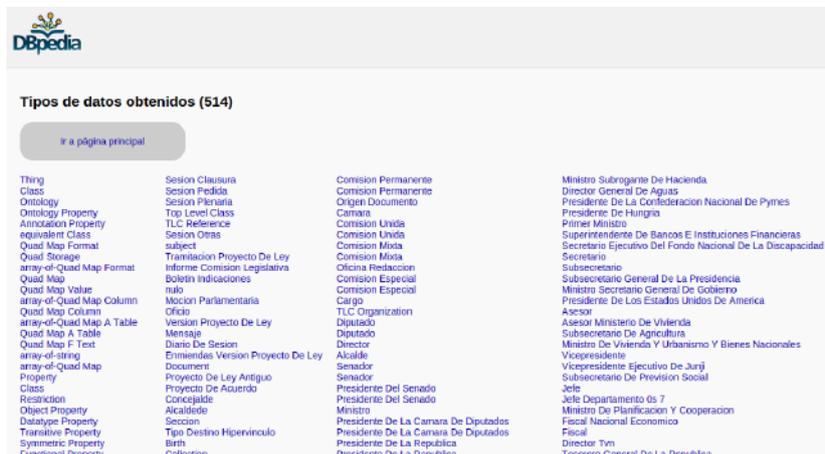


Fig. 6. Tipos de datos (categorías).



Fig. 7. Visualización de los datos de Serendipity.

Tabla 1. Rendimiento de la herramienta en OS X 10.10.3.

Query	Primera búsqueda	Siguientes búsquedas	No. Resultados
<i>Australia International Soccer Players</i>	39.7 seg.	10.7 seg.	329
<i>Social Scientist</i>	46.3 seg.	8.6 seg.	10,364
<i>Computer Game Program</i>	48.9 seg.	11.4 seg.	10,000
Visualización de la herramienta	0.0433 No. resultados/seg.	0.0114 No. resultados/seg.	

Tabla 2. Rendimiento de la herramienta en Windows 8.

Query	Primera búsqueda	Siguientes búsquedas	No. Resultados
<i>Australia International Soccer Players</i>	41 seg.	10.65 seg.	329
<i>Social Scientist</i>	47.1 seg.	8.52 seg.	10,364
<i>Computer Game Program</i>	48.2 seg.	11.7 seg.	10,000
Visualización de la herramienta	0.0446 No. resultados/seg.	0.01143 No. resultados/seg.	

La última fila de cada tabla (Visualización de la herramienta) representa el rendimiento de la herramienta al momento de ser visualizada en el navegador; para obtener estos valores se utilizó la herramienta *Page Speed Monitor* de [19].

De acuerdo con los resultados registrados en las Tablas 1, 2 y 3, el Sistema Operativo que ofrece un mejor tiempo de respuesta y rendimiento es Linux - Fedora 17, aunque la diferencia no es tan significativa en los otros S.O.

5. Comparativo con otras herramientas

Con la finalidad de identificar las ventajas y desventajas de nuestra herramienta respecto de las herramientas presentadas en la Sección de Trabajo rela-

Tabla 3. Rendimiento de la herramienta en Linux - Fedora 17.

Query	Primera búsqueda	Siguientes búsquedas	No. Resultados
<i>Australia International Soccer Players</i>	37.52 seg.	10.4 seg.	329
<i>Social Scientist</i>	44.1 seg.	11.1 seg.	10,364
<i>Computer Game Program</i>	45.3 seg.	10.9 seg.	10,000
Visualización de la herramienta	0.0409 No. resultados/seg.	0.01125 No. resultados/seg.	

cionado, construimos dos tablas comparativas. La Tabla 4 está enfocada en los aspectos técnicos, dentro de los cuales consideramos el lenguaje de programación utilizado para construir el motor de búsqueda, el nombre del framework/software de almacenamiento de la información en tripletas y mencionar si la herramienta utiliza un *endpoint* de SPARQL. La Tabla 5 contiene características centradas en el desarrollo y funcionamiento de la herramienta, estas características son: indicar si la herramienta presenta una interfaz amigable para el usuario y en qué plataformas funciona; si la herramienta es portátil; si se considera escalable; el tipo de enfoque (o grupo) al que pertenece el motor de búsqueda (de los mencionados en la Sección 2); si utiliza Lenguaje Natural en las consultas y; si es comercial.

Tabla 4. Comparativo: aspectos técnicos de las herramientas.

Motor de búsqueda	Lenguaje de programación	Almacenamiento en tripletas	Endpoint
<i>SHOE</i>	JAVA	Parka KB	No disponible
<i>Yummlly</i>	NodeJS	No disponible	No disponible
<i>TAP</i>	No disponible	No disponible	GetData
<i>SemSearch</i>	JAVA	Repositorios de datos semánticos	Múltiples ontologías
<i>Evi</i>	JAVA y Objective C	No disponible	No disponible
<i>AquaLog</i>	No disponible	Sí	Sí
<i>DBPedia Search</i>	PHP	MySQL (aunque no es un almacenamiento en tripletas)	Intercambiable

6. Conclusiones y trabajo futuro

Un aspecto que consideramos importante mencionar es que nuestra herramienta funciona en diferentes Sistemas Operativos, aunque para cada uno de ellos fueron modificadas distintas características de las tecnologías utilizadas; estas modificaciones fueron debidamente documentadas para futuras implementaciones. Esto nos permite difundir rápidamente nuestra aplicación con el fin de validar nuestra herramienta con la mayor cantidad posible de usuarios finales.

El hecho de agregar gráficos en la visualización de los resultados en nuestra herramienta, marca una diferencia notable con las herramientas que se presentan en este documento, ya que ninguna de ellas expone sus resultados utilizando gráficos; lo que representa una característica importante en el análisis de la información para grandes cantidades de datos (Big Data).

Aunque, existe un problema que no podemos erradicar sin el apoyo de las organizaciones enfocadas en trabajar con la Web Semántica, SPARQL y los endpoints; este consiste en que existe una fuerte dependencia en el mantenimiento, disponibilidad y formato de los *endpoints*. Por esto, aunque la herramienta funcione correctamente, si los *endpoints* de SPARQL no se encuentran actualizados, la herramienta no presentará información útil para el usuario.

Como el trabajo futuro consideramos mejorar los aspectos que se enumeran a continuación:

1. Especificar las características de los mapas de calor; por ejemplo, la variación de la paleta de colores.
2. Realizar el tratamiento de las propiedades para combinar categorías comunes, aunque estas categorías contengan diferentes caracteres en sus nombres y/o se encuentren en otros idiomas. Por ejemplo, Lugar de nacimiento, Lugarnacimiento y *Birth Place*.
3. Construir de acuerdo con el tipo de información el (los) gráfico (s) más adecuado (s) para la visualización. De igual manera, permitir que el usuario valide los gráficos presentados por la herramienta, ya sea eliminando o agregando un gráfico.

Agradecimientos. A CONACYT por el apoyo de beca doctoral. A los estudiantes de Doctorado en Ciencias Computacionales del Tecnológico de Monterrey, por su apoyo en la etapa de pruebas.

Referencias

1. Spanos, D-E., Stavrou, P., Mitrou, N.: Bringing Relational Databases into the Semantic Web: A Survey. In: IOS Press, pp. 1–41 (2012)
2. Gutierrez, C., Hurtado, C., Mendelzon, A. O.: Foundations of Semantic Web Databases. In: ACM, PODS, pp. 95–106 (2004)
3. Recuperación y organización de la información a través de RDF usando SPARQL, <https://ggomez.files.wordpress.com/2008/09/informe-sparql.doc>
4. Sakr, S., Al-Naymat, G.: Relational Processing of RDF Queries: A Survey. In: SIGMOD Record, pp. 23–28 (2009)
5. Shadbolt, N., Hall, W., Berners-Lee, T.: The Semantic Web Revisited. In: IEEE Intelligent Systems, pp. 96–101 (2006)
6. Acosta, M., Vidal, M-E., Lampo, T., Castillo, J., Ruckhaus, E.: ANAPSID: An Adaptive Query Processing Engine for SPARQL Endpoints. In: Lecture Notes in Computer Science, The Semantic Web – ISWC, vol. 7031, pp. 18–34 (2011)
7. DBpedia. <http://dbpedia.org/>
8. Spanish DBpedia. <http://es.dbpedia.org/index-en.html>
9. Heflin, J., Hendler, J.: Searching the Web with SHOE. In: Artificial Intelligence for Web Search, AAAI Workshop, WS-00-01, pp. 35–40 (2000)
10. Semantic Search. http://www.willita.de/teaching/semweb14w/slides/4S_SemanticSearch.handout.pdf

11. Guha R., McCool R.: TAP: A Semantic Web Platform. *Computer Networks*, vol. 42 (5), pp. 557–577 (2003)
12. Lei, Y., Uren V., Motta E.: SemSearch: A Search Engine for the Semantic Web. In: *EKAW'06 Proceedings of the 15th international conference on Managing Knowledge in a World of Networks*, vol. 4248, pp. 238–245 (2006)
13. Evi Technologies Ltd. <https://www.evi.com/>
14. Lopez, V., Pasin M., Motta E.: AquaLog: An Ontology-Portable Question Answering System for the Semantic Web. In: *The Semantic Web: Research and Applications*, vol. 3532, pp. 546–562 (2005)
15. ARC RDF Classes for PHP. <https://github.com/semsol/arc2>
16. D3 Data-Driven Documents. <http://d3js.org/>
17. Biblioteca del Congreso Nacional de Chile / BCN. <http://www.bcn.cl/>
18. Serendipity. <http://datahub.io/es/dataset/serendipity>
19. Page Speed Monitor. <https://chrome.google.com/webstore/detail/apptelemetry-page-speed-m/anlomjepbdgcgkebglgfkinmdjgelhd?hl=en>

Tabla 5. Comparativo sobre los aspectos del desarrollo de las herramientas.

Motor de búsqueda	Amigable para el usuario	Portátil	Escalable	Enfoque	Lenguaje Natural	Co-mercial
<i>SHOE</i>	El plugin no se encuentra actualizado; por lo tanto, no se puede ver en los navegadores web actuales	No mencionado en el artículo	Sí	Basado en forma	No	No
<i>Yummly</i>	Sí, disponible en la Web y como una aplicación móvil	No, aunque tiene un API	No, está construido específicamente para ontologías de comida	Basado en forma	Sí	Sí
<i>TAP</i>	Las demostraciones ya no están disponibles	Sí	Sí, a través de GetData	Basado en palabra clave	No	No
<i>SemSearch</i>	Sí, aunque no con el detalle de otros motores de búsqueda	No mencionado en el artículo	Sí	Basado en palabra clave	No	No
<i>Evi</i>	Sí, disponible en aplicaciones móviles	No, sólo privado	Sí	Pregunta / Respuesta	Sí	Sí
<i>AquaLog</i>	Las demostraciones ya no están disponibles	Sí	Sí	Pregunta / Respuesta	Sí	No
<i>DBPedia Search</i>	Sí, a pesar de que solo puede ser visto en la Web, es la única herramienta que utiliza la visualización de datos por medio de gráficos	Sí, puede utilizar otras ontologías además de DBpedia	Puede utilizar sólo una ontología	Basado en palabra clave	No	No, por el momento