Revista Venezolana de Computación
ISSN: 2244-7040
http://www.svc.net.ve/revecom
Vol. 1, No. 1, pp. 34-42, Junio 2014
Selección de los Mejores Artículos de CoNCISa 2013

Retos para el Procesamiento Semántico de Datos Enlazados en la Nube de los Datos Abiertos Enlazados

Maria-Esther Vidal, Maribel Acosta, Ana Alvarado, Oriana Baldizán, Alexander Baranya,
Simón Castillo, Giuseppe De Simone, Marlene Goncalves, Hancel González, Alexandra La Cruz,
Gabriela Montoya, Guillermo Palma, Edna Ruckhaus
{mvidal, macosta, faalvarado, obaldizan, abaranya, scastillo}@ldc.usb.ve
{gdsimone, mgoncalves, hgonzalez, alacruz, gmontoya, gpalma, ruckhaus}@ldc.usb.ve

Departamento de Computación, Universidad Simón Bolívar, Caracas, Venezuela

Resumen: Este artículo describe los proyectos desarrollados por los integrantes del Grupo de la Web Semántica en la Universidad Simón Bolívar en el contexto de la Nube de los Datos Abiertos Enlazados. Los principales retos abordados en estos proyectos son los siguientes: Descomposición de consultas SPARQL y técnicas de procesamiento para escalar a federaciones con una gran número de fuentes de datos; Estrategias para explotar anotaciones semánticas de ontologías médicas para mejorar la calidad del renderizado de imágenes médicas; Técnicas de *ranking* basadas en *skyline* para de forma eficiente identificar los recursos que satisfacen consultas multi-objetivo; Técnicas de limpieza en Datos Enlazados para identificar ambiguedades entre datos enlazados y sugerir posibles inconsistencias y falta de completitud; Técnicas de Minería de grafos para descubrir patrones entre grafos anotados; Herramientas para la evaluación de la calidad y el rendimiento de las máquinas de consultas SPARQL sobre documentos SPARQL. Finalmente, se presenta un resumen de los resultados de los estudios experimentales los cuales muestran la calidad de las estrategias desarrolladas y los casos de uso donde las arquitecturas propuestas han sido aplicadas.

Palabras Clave: Web Semántica; Datos Abiertos Enlazados; Optimización y Ejecución de Consultas; Ontologías Médicas; Técnicas de Limpieza de Datos; Manejo de Imágenes Médicas; Minería de Datos.

Abstract: In this paper, we provide an overview of the projects developed by the Semantic Web group of the University Simón Bolívar in the context of the Linking Open Data Cloud, that address the following challenges: Semantic Query Decomposition and Processing techniques to scale up to federations of very large linked datasets; Strategies to exploit semantic annotations of medical ontologies to improve the quality of medical image rendering; A skyline-based ranking approach to efficiently rank resources that fulfill multi-objective queries; Linked Data Cleansing techniques to identify ambiguities among the linked data, and suggest possible inconsistencies and incompleteness; Mining techniques to discover patterns among linked annotated graphs; Benchmarking tools to evaluate the quality and performance of the existing RDF engines which are used to execute SPARQL queries against federation endpoints. Finally, we summarize experimental results that show the quality of the developed strategies and the use cases where the proposed architecture has been applied.

Keywords: Semantic Web; Linked Open Data Cloud; Query Optimization and Execution; Biomedical Ontologies; Cleaning Data; Management of Medical Images; Data Mining.

I. Introducción

En la última década el número de conjuntos de datos en la nube de los datos abiertos enlazados (LOD *cloud*) ha explotado así como el número de *endpoints* de SPARQL que acceden y manejan estos conjuntos de datos. Usuarios más que nunca pueden recuperar datos que satisfacen sus requerimientos al buscar o consultar alguna de las fuentes disponibles. Esta democratización de información sienta las bases para el descubrimiento de propiedades y relaciones entre datos y enlaces que no podrían ser identificados años anteriores. Esto ha originado problemas abiertos en las tareas tradicionales de las máquinas federadas de procesamiento de datos, *benchmarking*, visualización de imágenes, *ranking* multi-objetivo, limpieza de datos y descubrimiento de patrones.

Endpoints de SPARQL que existen actualmente para acceder y manejar conjuntos de datos enlazados, deben ser capaces de ejecutar cualquier consulta SPARQL. Sin embargo, algunos *endpoints* no son capaces de resolver consultas cuyo tiempo de ejecución o cardinalidad de la respuesta sobrepasa un cierto valor, mientras otros simplemente detienen su ejecución sin

producir respuesta alguna. Sin tener la tecnología lista para el manejo de cualquier consulta sobre los *endpoints*, existe la necesidad de desarrollar técnicas de descomposición de consultas que reescriben las consultas en subconsultas que pueden ser ejecutadas. Adicionalmente, los datos accesibles en el Web están usualmente caracterizados por: *i*) ausencia de estadísticas, *ii*) condiciones impredecibles al ejecutar las consultas remotas, y *iii*) características cambiantes.

El paradigma de optimización y luego ejecución que ha sido tradicionalmente usado para la identificación de planes de ejecución eficientes, no puede adaptarse a cambios no predecibles en los datos; así, técnicas de procesamiento de consultas adaptativas son necesarias. Aunque los *endpoints* de SPARQL y las iniciativas para impulsar el desarrollo de la nube de datos enlazados abiertos sientan las bases para el acceso de largos volúmenes de datos, existen aún aplicaciones donde es importante identificar las tuplas que mejor satisfacen una consulta o requerimiento. Basándonos en trabajos relacionados, proponemos una solución para este problema de *ranking* y desarrollamos una técnicas para identificar los recursos que mejor satisfacen una consulta multi-objetivo, i.e., los recursos para los cuales no existe algún otro recurso que sea mejor en todos las condiciones del criterio multi-objetivo. Este conjunto de puntos no dominados se conoce como *skyline*.

Además, colecciones en la Nube de los Datos Abiertos Enlazados facilitan a los científicos la minería de los conjuntos enlazados para descubrir patrones o sugerir potenciales nuevas asociaciones. Para abordar el problema de descubrimiento de patrones, hemos desarrollado métricas de similitud que midan relaciones entren dos conceptos, y así, explorar las evidencias de las anotaciones, hacer predicciones, encontrar *outliers* y evaluar hipótesis. Sin embargo, para asegurar resultados confiables, datos enlazados deben cumplir altos estándares de calidad. En consecuencia, herramientas para detectar posibles ambigüedades y problemas de calidad son necesarias. Hemos implementado técnicas basadas en Redes Bayesianas y un sistema de reglas para analizar la calidad de los datos y proponer enlaces para resolver estas ambigüedades.

En este artículo describimos SEPIAS (*A Semantic Data Processing and Cleansing, analysis and visualization*) una herramienta para manejar y procesar datos enlazados en el Nube de los Datos Abiertos Enlazados. SEPIAS está compuesto por siete subsistemas que proveen técnicas para ejecutar las siguientes tareas: *i*) Descomposición y Procesamiento Semántico de Consultas, *ii*) Renderizado de Imágenes Médicas basado en Anotaciones Semánticas, *iii*) *Ranking* basado en skyline, *iv*) Limpieza de Datos Enlazados, *v*) Minería de Grafos Anotados, y *vi*) Herramientas de *Benchmarking*.

Este artículo está compuesto de cuatro secciones. En la Sección II, se motiva el trabajo realizado a través de un ejemplo. Luego en la Sección III, se describe SEPIAS, y se detallan cada uno de sus componentes y como los mismos interactúan. Finalmente, la Sección IV presenta nuestras conclusiones y trabajos futuros.

II. Ejemplo Motivador

Considere la siguiente consulta expresada en SPARQL 1.0: *Seleccionar las enfermedades y los genes asociados a drogas que han sido probadas en ensayos clínicos donde el Cáncer de Próstata fue estudiado.*

```
PREFIX linkedct: <http://data.linkedct.org/resource/linkedct>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema>
PREFIX diseasome: <http://www4.wiwiss.fu-berlin.de/diseasome/resource/diseasome>
(0) SELECT DISTINCT ?II ?D ?GN2
(1) WHERE {
(2)   ?CT1 linkedct:condition?C1 .
(3)   ?CT1 linkedct:intervention ?I .
(4)   ?I linkedct:intervention_type "Drug" .
(5)   ?C1 rdfs:seeAlso ?D.
(6)   ?I rdfs:seeAlso ?II.
(7)   ?C linkedct:condition_name "Prostate Cancer" .
(8)   ?CT linkedct:intervention ?I .
(9)   ?CT linkedct:condition ?C .
(10)  ?D diseasome:associatedGene ?GN2 .
(11)  ?D diseasome:possibleDrug ?II}
```

La respuesta de esta consulta está compuesta por 192 tuplas cuando los datos son recuperados de Diseasome[1] y LinkedCT[2]. Sin embargo, si la consulta se ejecuta contra los *endpoints* existentes de Diseasome o LinkedCT, la respuesta es vacía. Este problema es generado por la necesidad de atravesar enlaces entre los conjuntos de datos para responder la consulta y la mayoría de los *endpoints* existentes han sido creados para consultas "ligeras" que no requieran navegar a través de varios conjuntos de datos. Otro problema que se ilustra en este ejemplo, es causado por el nivel de confianza que los usuarios tengan de los *endpoints* consultados. Por ejemplo, si los enlaces `rdfs:seeAlso` no son almacenados en estos conjuntos de datos, la respuesta podría ser incompleta, y drogas importantes podrían no haberse incluido en la respuesta. Adicionalmente, un científico podría estar interesado en analizar el grado de relación entre las enfermedades y los genes recuperados, para poder determinar posibles nuevas asociaciones, y así, por ejemplo, sugerir nuevos tratamientos. Finalmente, técnicas de *ranking* podrían ayudar a identificar las enfermedades y genes cuyas asociaciones satisfacen un cierto criterio de calidad o confianza, mientras que herramientas de *benchmarking* son requeridas para evaluar y entonar las diferentes técnicas de procesamiento que son necesarias para ejecutar estas consultas. En este artículo se describen proyectos desarrollados en el área de la Web Semántica y se abordan algunos de los retos brevemente descritos en este ejemplo.

III. La Arquitectura de SEPIAS

La Figura 1 presenta la arquitectura de SEPIAS, un sistema para el procesamiento semántico y análisis de datos RDF los cuales son accesibles a través de *endpoints* que respetan el protocolo SPARQL. Esta arquitectura esta compuesta de varios subsistemas; `ANAPSID`, `DEFENDER`, `FRAGOLA`, `LiQuate`, `ANISE` y `Patterns in ANnotation Graphs`. Usando SEPIAS el usuario puede enviar una consulta en SPARQL y obtener respuestas que corresponden a unión de los datos publicados en la federación de los *endpoints* SPARQL; las

[1] http://www4.wiwiss.fu-berlin.de/diseasome/sparql
[2] http://data.linkedct.org/sparql

técnicas de descomposición, optimización y ejecución implementadas por ANAPSID y DEFENDER permiten identificar subconsultas que pueden ser ejecutas por los *endpoints* SPARQL disponibles, planes de ejecución que minimizan el tiempo de ejecución, y ejecuciones que se adaptan a las condiciones de los *endpoints*. Las consultas SPARQL pueden ser enviadas no sólo por usuarios sino por los otros componentes de la herramienta, es decir: FRAGOLA, LiQuate, ANISE y Patterns in ANnotation Graphs. FRAGOLA implementa técnicas de *ranking* para resolver consultas multi-objetivo; para recuperar los datos se comunica con DEFENDER a través de las consultas SPARQL correspondientes. LiQuate se basa en una red Bayesiana para identificar inconsistencias en datos enlazados; para recuperar los datos que permiten construir la red Bayesiana interactúa con DEFENDER a través de las consultas SPARQL. De forma similar Patterns in ANnotation Graphs hace uso de DEFENDER para obtener los datos sobre los cuales se aplicarán las técnicas de minería de datos, así como, métricas de similitud para determinar relaciones entre los datos. Finalmente, ANISE hace uso de DEFENDER para recuperar la información de los términos que aparecen en las ontologías y que sirven de base para el proceso de razonamiento implementado a nivel del *segmentador*. A continuación presentamos una breve descripción de las principales propiedades que caracterizan a los subsistemas que componen a SEPIAS.

A. ANAPSID y DEFENDER Soluciones a Consultas SPARQL contra Federaciones de Endpoints

ANAPSID es una máquina de procesamiento adaptativa de datos accesibles a través de *endpoints* SPARQL [1]. ANAPSID identifica un plan de ejecución pero adapta sus operadores en base a la disponibilidad de los datos. Así, ANAPSID implementa una técnica a nivel del operador y es capaz de recuperar información desde la federación de los *endpoints* SPARQL y adaptar la ejecución de forma dinámica dependiendo de la carga de trabajo y disponibilidad de los *endpoints*. DEFENDER [12], [11] es un des-compositor de consultas sobre *endpoints* SPARQL. Primero, patrones de tripletas en la consulta se descomponen en subconsultas simples que pueden ser completamente ejecutadas por al menos un *endpoint*. Luego, las subconsultas son combinadas en un árbol tipo arbusto donde el número de *joins* se maximiza y la altura del árbol se minimiza. DEFENDER usa descripciones de los *endpoints* para seleccionar los *endpoints* SPARQL que pueden contestar cada patrón de tripleta; patrones de tripletas que pueden ser ejecutadas por los mismos *endpoints* se agrupan juntas. DEFENDER se implementó sobre ANAPSID e implementa adaptividad al nivel de las fuentes, resolviendo el problema de decisión de encontrar los *endpoints* SPARQL que pueden ejecutar una descomposición dada de la consulta de acuerdo a las condiciones actuales de la federación. DEFENDER está compuesto por un *Planificador de Consultas*, un *Motor de Consultas Adaptativo* y un *Catálogo de la descripción de los endpoints*. El *Planificador de Consultas* de DEFENDER está constituido principalmente por dos componentes: el *des-compositor de consultas* y el *Optimizador de Consultas Basado en una Heurística*. El primero divide los conjuntos de patrones de tripletas de consultas SPARQL 1.0, en subconjuntos de patrones de tripletas (sPT) que pueden ser ejecutados por un mismo *endpoint*. El des-compositor de consultas comienza creando una única subconsulta en sPT, entonces esta se mezcla con las subconsultas que comparten exactamente una variable y se repite el procedimiento hasta que se alcanza un punto fijo en el proceso de creación de subconsultas. Entonces los sPT que comparten una variable con cualquier sPT son agregados. Una vez que la consulta se vuelve a escribir en SPARQL 1.1, las técnicas de optimización basadas en heurísticas son aplicadas para generar un plan de árbol tipo arbusto, donde las hojas corresponde a subconsultas de los sPT previamente identificadas. Las técnicas de optimización no se basan en la información estadística recolectada de los *endpoints*, solamente hacen uso de la información sobre los predicados de los conjuntos de datos accesibles a través de los *endpoints*. Un algoritmo basado en una heurística ávida es implementado, éste atraviesa el espacio de planes en iteraciones y tiene como salida un plan de árbol tipo arbusto de una consulta SPARQL 1.1, en donde el número de *joins* es maximizado y la altura del árbol es minimizado. De esta forma el costo de ejecución es reducido. La Figura 2 compara el plan de ejecución generado por DEFENDER y otro generado por un sistema como FedX [17]. Cada hoja de los árboles es anotada con los *endpoints* donde las subconsultas se ejecutarán. Como puede observar, las subconsultas identificada en el grupo de DEFENDER tienen un número más largo de patrones de tripletas. Además la altura de árbol en más pequeña en el plan producido por DEFENDER. Estas dos propiedades de los planes producidos por DEFENDER, reduce el número de datos transferidos desde los *endpoints* y puede disminuir el tiempo total de ejecución. Estudiamos el rendimiento de los planes introducidos por DEFENDER y mostramos que estos planes son competitivos con los planes generados por los motores RDF existentes. Un portal que publica nuestros resultados experimentales, puede ser visitado en http://www.defender.ldc.usb.ve. El portal de DEFENDER presenta el comportamiento de 36 consultas SPARQL contra las colecciones de FedBench[3]: *Cross-Domain, Linked Data y Life Science* [16]. Estas consultas incluyen 25 consultas de FedBench y 11 consultas complejas[4]. Las consultas complejas están compuestas de entre 6 y 48 patrones de tripletas y pueden ser descompuestas en hasta 8 subconsultas. Las colecciones de FedBench fueron ejecutadas a través de los nueve (9) *endpoints* de Virtuoso[5] con un tiempo máximo de espera de 240 segundos o 71.000 tuplas. Se consideraron diferentes federaciones donde los datos fueron particionados siguiendo diferentes criterios de fragmentación: horizontal, vertical e híbrido. Se evaluó el comportamiento de tres máquinas de ejecución de consultas: ARQ, FedX y DEFENDER. Como resultado de la evaluación es el siguiente:

Parámetros que afectan la completitud de la respuesta a las consultas. Las consultas estudiadas están confor-

[3] http://code.google.com/p/fbench
[4] http://www.ldc.usb.ve/~mvidal/FedBench/queries/ComplexQueries
[5] http://virtuoso.openlinksw.com

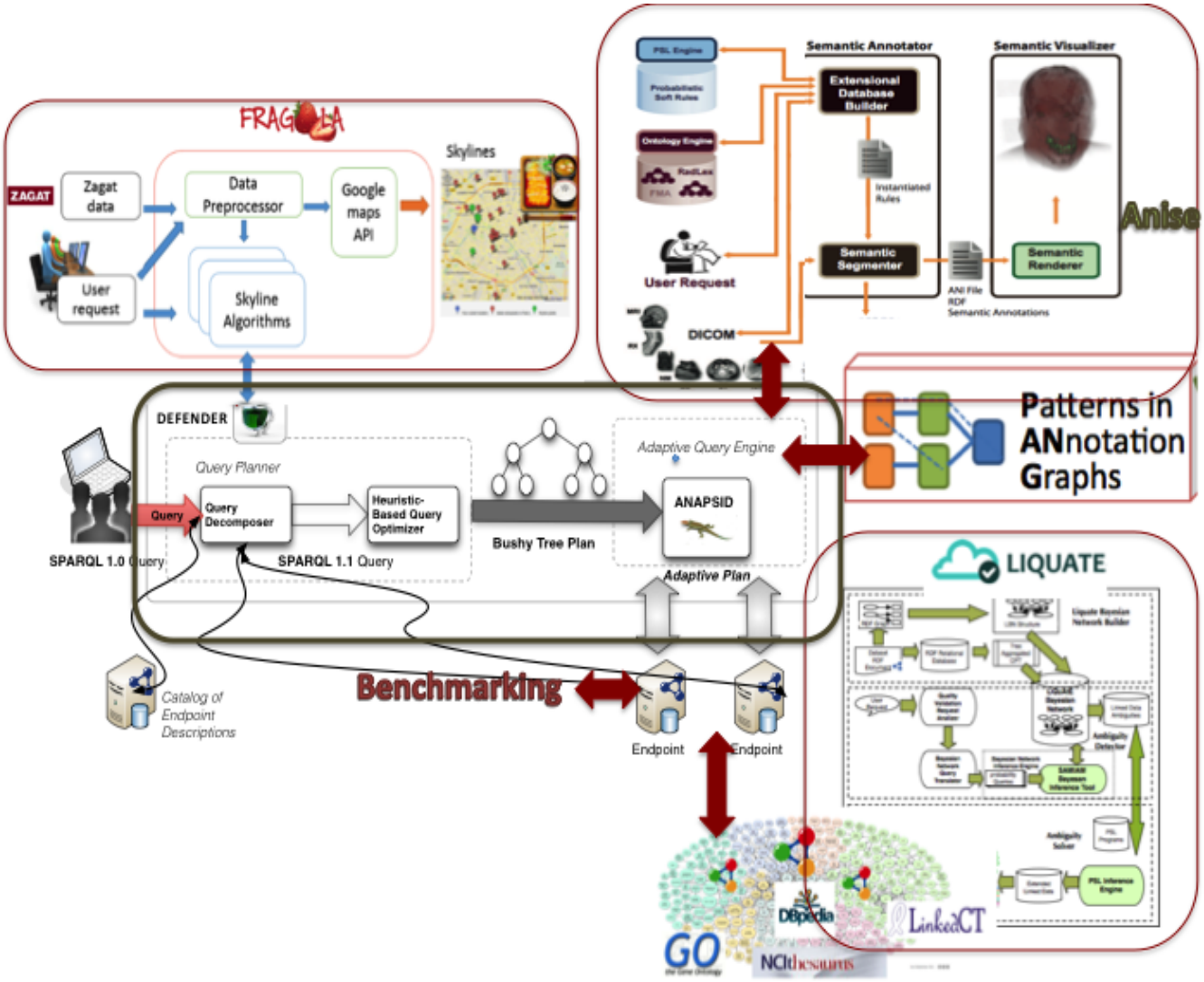
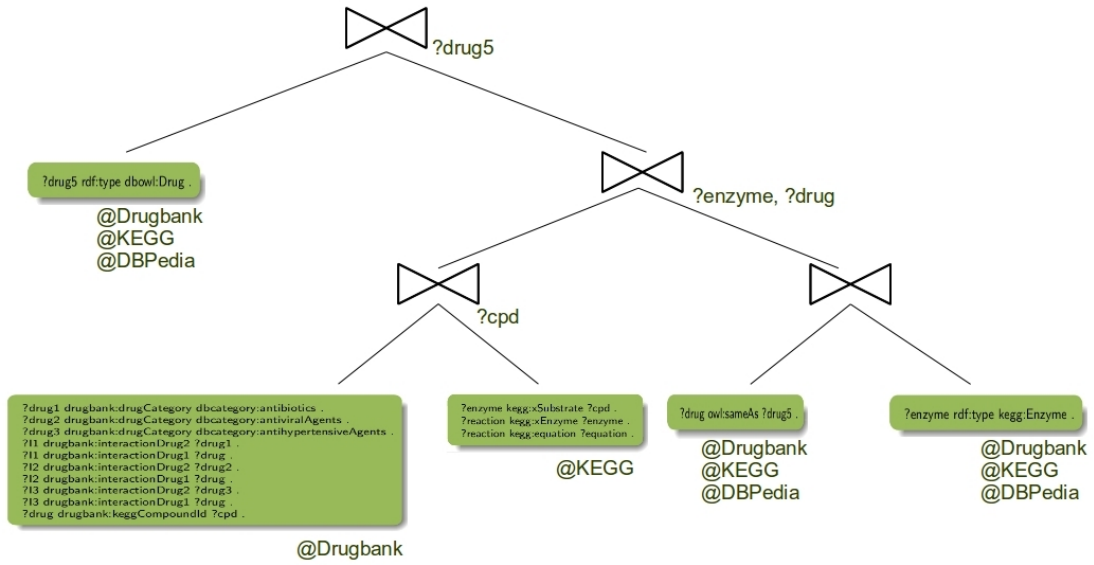


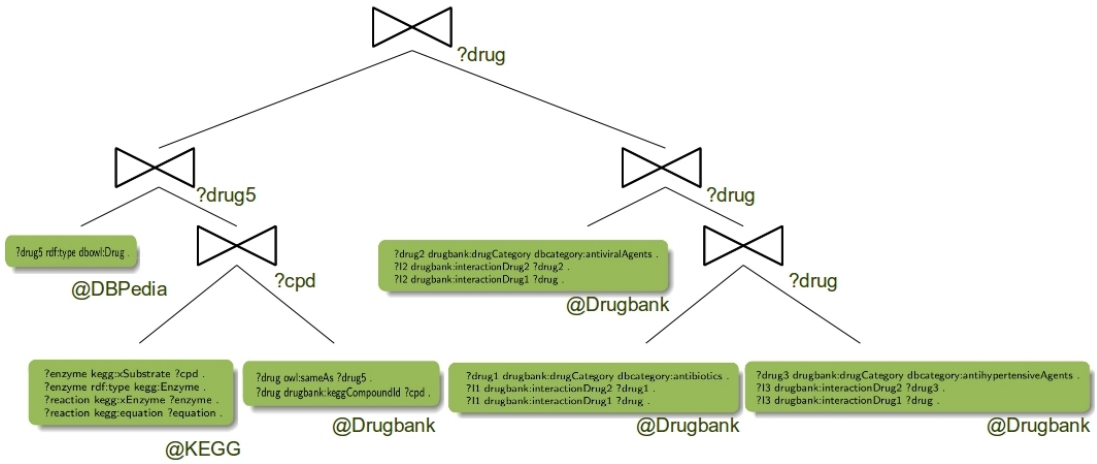
Figure 1: La Arquitectura de SEPIAS

madas de predicados generales tales como `owl:sameAs` y `rdfs:seeAlso` que están presentes en la mayoría de los conjuntos de datos. Sin embargo, sólo un número muy reducido de *endpoints* que exportan tripletas con estos predicados, produce la respuesta necesaria para contestar la consulta. Esto trae como consecuencia que la máquina de ejecución debe implementar técnicas de selección de consultas para determinar cuales son los *endpoints* relevantes para evitar transferencia de datos no relevantes y garantizar completitud de las respuestas. Se pudo observar que el tiempo de ejecución de los planes producidos por FedX se puede ver afectado por este tipo de predicados, ya que esta máquina de ejecución trata de contactar todas los *endpoints*. Por otro lado, `DEFENDER` implementa distintas heurísticas para determinar cuales son los *endpoints* relevantes; las estrategias implementadas permiten reducir el tiempo de ejecución pero en algunos casos comprometen la completitud de la respuesta de la consulta. En conclusión pudimos observar que el tipo de predicados en los patrones de tripletas de las consultas pueden afectar el comportamiento de las máquinas de ejecución en ambientes de federaciones de *endpoints*.

Efectos del tipo de consulta en el tiempo de ejecución y en la completitud de la consulta. Se estudiaron planes de ejecución optimales para las diferentes consultas, es decir, planes lineales izquierdos y tipo arbusto. Estos planes fueron evaluados en ARQ y `DEFENDER`. Se pudo observar que los planes tipo arbusto reducen el tiempo de ejecución en al menos un orden de magnitud en ambas máquinas. También el estudio sugiere que `DEFENDER` puede ejecutar de manera más eficiente consultas con un gran número de patrones de tripletas; cabe destacar que FedX y ARQ no son capaces de producir respuesta en al menos en 30 minutos para las consultas complejas estudiadas. Finalmente, el tiempo requerido para ejecutar las subconsultas identificadas por `DEFENDER` es mucho menor que el requerido por FedX y ARQ. En base a los resultados observados se puede concluir que el tipo de consulta si afecta el comportamiento de los máquinas de ejecución.



(a) Planes Basados en Grupos Exclusivos (Sistemas como FedX)



(b) Planes Producidos por DEFENDER

Figure 2: Comparación de Planes Producidos por Sistemas como FedX y DEFENDER

B. Fragola: Una Propuesta para Identificar Los Mejores Lugares Gastronómicos

FRAGOLA es un sistema de clasificación jerárquica que implementa el algoritmo *skyline* FOPA (*Final Object Pruning Algorithm)* [2] que es capaz de producir de manera eficiente todo el conjunto de datos de los puntos de *skyline* y escala hasta grandes conjuntos de datos. FOPA se basa en tablas particionadas verticalmente ordenadas para almacenar datos RDF recuperados de la federación de *endpoints* SPARQL, información sobre los valores observados hasta ahora e índices de la tablas particionadas verticalmente para podar el espacio de los puntos dominantes. FOPA es apto para identificar el *skyline* para conjuntos de datos grandes en menos tiempo que los enfoques del estado del arte. Como una prueba del concepto, desarrollamos FRAGOLA (*Fabulous RAnking of GastrOnomy LocAtions)*, una herramienta que implementa FOPA y clasifica jerárquicamente lugares gastronómicos basado en criterios multi-objetivos.

El sistema FRAGOLA fue desarrollado sobre el conjunto de datos multidimensional de restaurantes en París que es proporcionado por Zagat[6]. La ontología del restaurante[7] es usada para describir cada restaurante, mientras que Geonames[8] es usado para describir la localización geo-espacial. La respuesta a una consulta multi-objetivo es un conjunto de restaurantes que no son comparables, que componen el *skyline* con respecto a los atributos y directivas consideradas en la consulta. FRAGOLA está compuesto de los siguientes componentes: *i*) un *Pre-procesador de Datos*, *ii*) un *Motor de Skyline* y *iii*) el *API de Google Maps*. El pre-procesador de datos transforma los datos proporcionados por Zagat en la escala $[0, 1]$ y calcula la distancia entre la posición geo-espacial actual de un usuario y la de cada restaurante. Un motor de *skyline* implementa tres algoritmos de *skyline*: IDSA, desarrollado por Balke et al [4], RSJFH, introducido por Chen et al [7] y FOPA. El API de

[6] http://www.zagat.com/paris
[7] http://schema.org/Restaurant
[8] http://www.geonames.org

Google Maps es usado para visualizar el conjunto *skyline* de restaurantes en el mapa. La Tabla I reporta el número promedio de: lecturas, *joins*, comparaciones y podas realizados por cada algoritmo; los mejores resultados se resaltan en **negrillas**. Se observa que FOPA puede reducir el número de comparaciones hasta en dos órdenes de magnitud, mientras que el número de lecturas y *joins* es reducido en un orden de magnitud; los resultados observados sugieren que la combinación de ordenamiento de los datos y técnicas de poda pueden reducir el tipo de ejecución de consultas multi-objetivo. El demo está publicado en http://fragola.ldc.usb.ve.

Table I: Rendimiento de RSJFH, IDSA y FOPA

Algoritmos	Lecturas	*Joins*	Comparaciones	Podas
RSJFH	119	119	139	**107**
IDSA	49	40	5417	0
FOPA	**28**	**27**	**70**	16

C. Limpiando Datos Enlazados con LiQuate

`LiQuate` es una herramienta capaz de identificar ambigüedades entre la datos enlazados de la federación de *endpoints* SPARQL y sugiere posibles inconsistencias e incompletitudes. LiQuate también implementa un enfoque de doble pliegue que combina Redes Bayesianas y sistemas basados en reglas para analizar la calidad de los datos y proponer nuevos enlaces que resuelven las ambigüedades detectadas. `LiQuate` ha sido construido sobre los conjuntos de datos enlazados de *Life Science* que mantiene datos relacionados con intervenciones, condiciones, drogas, enfermedades y las relaciones entre ellos. *LiQuate* recibe una *solicitación de validación de calidad* la cual es expresada como una o más consultas de evidencias contra la Red Bayesiana. La respuesta de una *solicitación de validación de calidad* es un número en el rango $[0.0 : 1.0]$ que indica la probabilidad de que los datos y enlaces sean inconsistentes o incompletos. Actualmente pueden ser expresadas tres tipos de solicitudes de validación de calidad: *i*) probabilidad de que una etiqueta o nombre de un tipo o recurso es redundante, *ii*) probabilidad de enlaces incompletos en un conjunto de recursos dado y *iii*) probabilidad de enlaces inconsistentes. `LiQuate` está compuesto por tres componentes: el *Constructor de Redes Bayesianas de LiQuate*, el *Detector de Ambigüedad* y el *Resolvedor de Ambigüedad*. El demo de `LiQuate` está publicado en http://liquate.ldc.usb.ve.

D. Descubriendo Patrones en Datos Enlazados

`Patterns in ANnotation Graphs` [3] es una herramienta que permite a científicos identificar patrones in conjuntos de datos de grafos anotados. Éste provee diferentes métricas de similitud para computar distancias o el grado de relación entre dos conceptos científicos. Los grafos anotados son obtenidos de la federación de *endpoints* SPARQL, haciendo uso de las técnicas de procesamiento semántico de consultas implementadas por `DEFENDER` y `ANAPSID`. Un primer paso para descubrir patrones complejos requiere determinar el grado de relación (o similitud) de un par de conceptos, basado en sus anotaciones con respecto a una o más ontologías. Un ejemplo es identificar el grado de relación o similitud entre pares (droga, droga), basado en la anotación de evidencia de enfermedades (condiciones) en una ontología dada. Esto puede guiar a descubrimientos de nuevos objetivos para drogas existentes, o se puede predecir potenciales efectos colaterales de drogas existentes. Nosotros abordamos el reto del análisis de datos enlazados a gran escala de los conjuntos de datos de grafos anotados, usando el conocimiento semántico de las ontologías. Para explicar esto, considere dos drogas `Brentuximab vedotin` y `Catumaxomab`. La Figura 3 representa un subgrafo de un grafo de anotación; intervenciones están en verde, condiciones están en rectángulos rosados, y términos de la ontología NCI *Thesaurus* están encerrados en óvalos rojos. Cada camino entre un par de condiciones, ej., `Carcinoma` y `Anaplastic Large Cell Lymphoma` a través de NCI *Thesaurus* es identificado usando círculos rojos los cuales representan términos de la ontología NCI *Thesaurus*. El número de círculos rojos representa la longitud del camino. Para simplificar la Figura, sólo ilustramos los caminos desde el término `Carcinoma`.

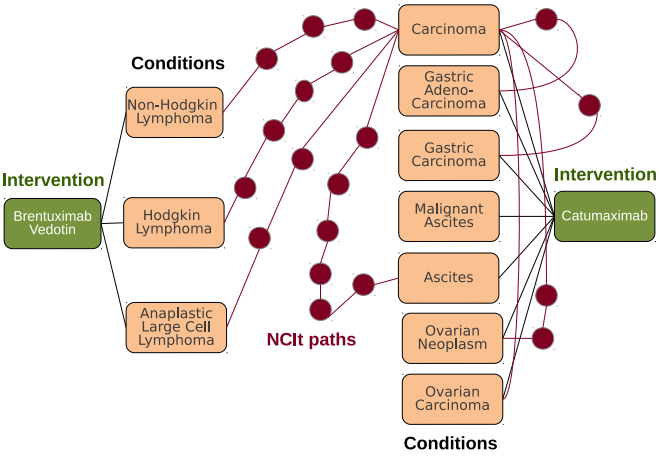


Figure 3: Subgrafo de Anotaciones de las Drogas `Brentuximab Vedotin` y `Catumaxomab`

Nuestro objetivo es identificar patrones de conectividad que puedan ser utilizados para proporcionar un resumen completo de la relación de los conceptos conectados. Esta investigación ha sido reportada en [6], [13], [14]. Definimos dos tipos de métricas de similitud: *i*) métricas de similitud topológica que mide el grado de relación en términos de la cercanía de los términos de vocabulario controlado en una taxonomía o una ontología dada y *ii*) métricas de similitud basadas en anotaciones que miden el grado de relación entre dos entidades anotadas en términos de la similitud de sus anotaciones.

En primer lugar definimos una métrica de distancia taxonómica llamada d_{tax}[6]. La intuición detrás de la métrica d_{tax} es capturar la distancia taxonómica entre dos vértices con respecto a la profundidad del ancestro común de estos dos vértices. Además, d_{tax} trata de asignar los más bajos valores de distancia taxonómica a pares de vértices que están (1) a una profundidad más grande en la ontología (2) están más cercanos a su ancestro común más bajo. Un valor cercano a 1.0 indica que ambos vértices son generales o que que el ancestro

Table II: Similitud Conjunto de Datos 1: $(1 - d_{tax})$ para SNOMED, MeSH, y NCIt

Términos Médicos	Phy	Cod	SNOMED	MeSH	NCIt
Renal Insufficiency - Kidney Failure	**4.00**	**4.00**	**1.00**	**1.00**	**1.00**
Heart - Myocardium	**3.30**	3.00	**0.77**	**0.80**	0.20
Stroke - Infarction	**3.00**	2.80	0.31	**0.80**	**0.87**
Abortion - Miscarriage	3.00	**3.30**	**0.89**	0.00	**0.92**
Delusions - Schizophrenia	3.00	2.20	0.00	0.00	0.80
Congestive heart failure - Pulmonary edema	3.00	**1.40**	0.50	0.00	0.59
Metastasis - Adenocarcinoma	**2.70**	**1.80**	**0.83**	**0.25**	0.00
Calcification Stenosis	2.70	**2.00**	**0.55**	0.00	0.40
Diarrhea - Stomach cramps	2.30	**1.30**	0.29	0.75	0.42
Mitral Stenosis - Atrial Fibrillation	**2.30**	1.30	**0.63**	0.50	0.53
Chronic obstructive pulmonary disease - Lung infiltrates	**2.30**	1.90	0.70	-	0.13
Rheumatoid Arthritis - Lupus	**2.00**	**1.00**	**0.50**	**0.00**	0.86
Brain tumor - Intracranial hemorrhage	2.00	**1.30**	0.63	0.63	**0.17**
Carpal Tunnel Syndrome - Osteoarthritis	2.00	**1.00**	0.33	**0.00**	0.33
Diabetes Mellitus - Hypertension	**2.00**	**1.00**	0.64	**0.00**	0.17
Acne - Syringe	2.00	**1.00**	**0.00**	**0.00**	**0.00**
Antibiotic - Allergy	1.70	**1.00**	**0.00**	**0.00**	**0.00**
Cortisone - Total knee replacement	1.70	**1.00**	**0.00**	**0.00**	**0.00**
Pulmonary Embolism - Myocardial Infarction	**1.70**	1.20	0.36	0.29	0.63
Pulmonary Fibrosis - Lung Cancer	1.70	1.40	0.75	0.67	0.60
Cholangiocarcinoma - Colonoscopy	1.30	**1.00**	**0.00**	**0.00**	**0.00**
Lymphoid hyperplasia - Laryngeal cancer	1.30	**1.00**	0.43	**0.00**	0.36
Multiple Sclerosis - Psychosis	**1.00**	**1.00**	0.44	**0.00**	0.33
Appendicitis - Osteoporosis	**1.00**	**1.00**	0.31	**0.00**	0.50
Rectal polyp - Aorta	**1.00**	**1.00**	**0.00**	-	**0.00**
Xerostomia - Liver Cirrhosis, Alcoholic	**1.00**	**1.00**	**0.00**	**0.00**	0.14
Peptic Ulcer - Myopia	**1.00**	**1.00**	0.23	**0.00**	0.15
Depression- Cellulitis	**1.00**	**1.00**	**0.00**	**0.00**	0.31
Varicose vein - Entire knee meniscus	**1.00**	**1.00**	0.13	-	**0.00**
Hyperlipidemia - Metastasis	**1.00**	**1.00**	0.33	**0.00**	**0.00**

común más bajo es cercano a la raíz de la ontología. En consecuencia, (1 - d_{tax}) será usado como métrica de similitud ontológica o grado de relación entre dos nodos. En segundo término, proponemos una nueva métrica de similitud llamada *AnnSim*, que mide el grado de relación entre dos entidades en términos de la similitud o grado de relación del conjunto de sus anotaciones. Modelamos *AnnSim* como un apareamiento 1-a-1 de máximo peso de un grafo bipartito. La computación de *AnnSim* primero requiere construir a grafo bipartito completo, computando todos los pares de similitudes entre los términos y luego entonces determinando el apareamiento 1-a-1 de máximo peso del grafo bipartito. Realizamos un estudio experimental para evaluar la calidad de las métricas definidas; estudiamos la capacidad de predicción de nuestras métricas en varios conjuntos de datos.

Conjunto de datos 1: 30 pares de enfermedades del *Mayo Clinic Benchmark*; cada par se codifica por su similitud desde 1.0 (menos similar) hasta 4.0 (más similar). La codificación fue llevada a cabo por 3 médicos (**Phy**) y 10 codificadores médicos de la Clinica Mayo (**Cod**) [10], [15]. Las enfermedades fueron anotadas con NCI *Thesaurus* versión 12.05d[9]. El conjunto de datos 1 es usado para comparar $(1 - d_{tax})$ usando SNOMED[10], MeSH[11], and the NCI *Thesaurus*.

Conjunto de datos 2: 12 drogas anticancer en la intersección de anticuerpos monoclonales y agentes antineoplásticos: `Alemtuzumab`, `Bevacizumab`, `Brentuximab vedotin`, `Cetuximab`, `Catumaxomab`, `Edrecolomab`, `Gemtuzumab`, `Ipilimumab`, `Ofatumumab`, `Panitumumab`, `Rituximab`, and `Trastuzumab`. Las drogas fueron asociadas con condiciones y enfermedades en ensayos clínicos de LinkedCT de septiembre 2011 y cada enfermedad fue enlazada a su correspondiente término en el NCI *Thesaurus* versión 12.05d.

Primero, anotamos las 30 enfermedades del conjunto de datos 1 con sus términos correspondientes de SNOMED, MeSH y NCI *Thesaurus*. Las puntuaciones determinadas por $(1-d_{tax})$ fueron comparadas con la evaluación realizada por los médicos y codificadores. Los resultados revelan que d_{tax} es exitoso en la computación de valores altos de similitud para los pares que también son clasificados con alta puntuación por los médicos y codificadores. La Tabla II ilustra los resultados; celdas Vacías (-) representan términos que no aparecen en la ontología; valores resaltados en **negrillas** muestran una correlación relevante dada por el médico, el codificador y la métrica correspondiente. Además, condujimos una evaluación extensiva de patrones de conectividad en el conjunto de datos 2. Observamos que *AnnSim* consistentemente asigna altos valores de similitud a drogas que son usadas para tratar enfermedades similares. Los detalles del conjunto de datos 2 como sus anotaciones y los valores de similitud ente los pares que se obtienen con *AnnSim* pueden encontrados en http://pang.umiacs.umd.edu/AEDdemo.html.

E. ANISE: Una Herramienta para el Procesamiento de Imágenes Médicas

`ANISE` [5] es una herramienta que permite anotar semánticamente imágenes médicas, con el fin de permitir la visualización del tejido de interés con mejor precisión y calidad. ANISE se basa en la utilización del conocimiento

[9] http://ncit.nci.nih.gov
[10] http://www.nlm.nih.gov/research/umls/Snomed/snomed_main.html
[11] http://www.nlm.nih.gov/mesh

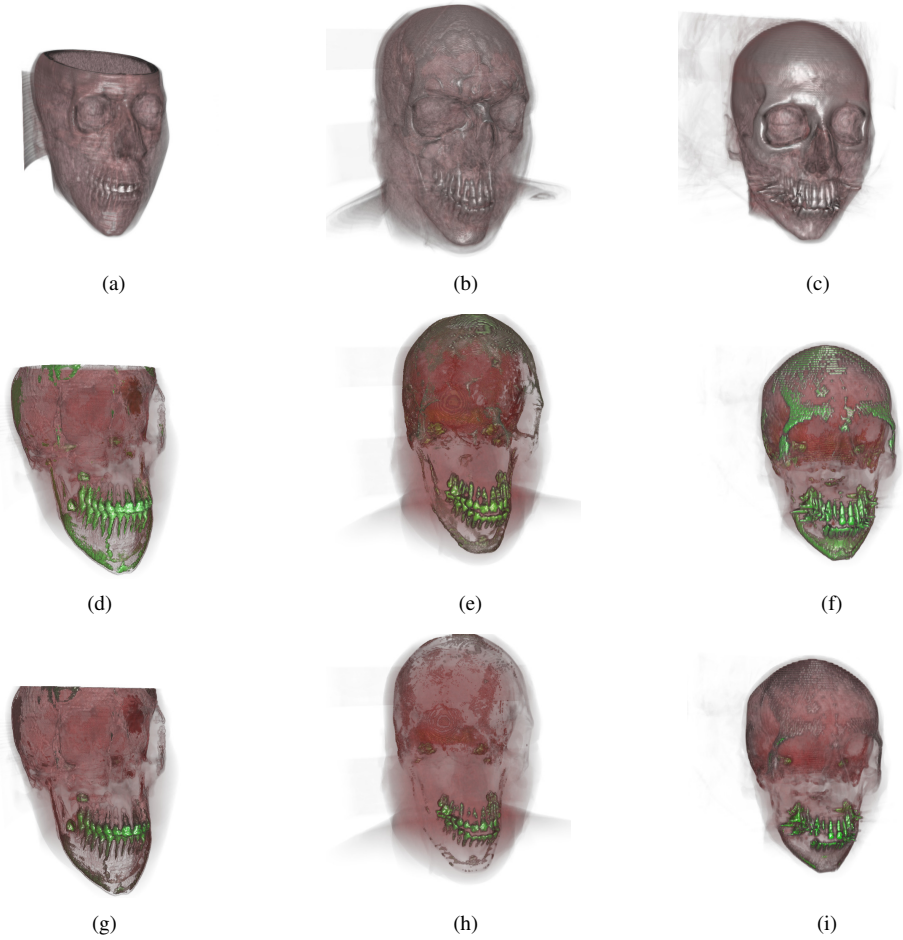


Figure 4: Resultados de Ejecutar las Técnicas Propuestas en Tres Volúmenes de Imágenes: (a) skewed_head (b) visible_human and (c) ct_head. Resultados del renderizado de tejidos usando el sistema de reglas aplicadas a: applied to (a) (b) (c) en los volumes mostrados en (d) (e) (f). Resultados de aplicar las reglas de segmentación se muestran en: (g) (h) (i)

expresado en las ontologías médicas FMA[12] y RadLex[13] y un sistema basado en reglas definidos en PSL [9]. Como resultado se obtiene los datos volumétricos particionados en un sub-volumen, el cual contiene el tejido u órgano de interés anotado, utilizando los términos correspondientes de las ontologías FMA y RadLex. La estrategia implementada en ANISE está compuesta por tres fases: *i*) segmentación semántica *ii*) descripción de los recursos, y *iii*) renderizado semántico del volumen. ANISE recibe como entrada un archivo DICOM, datos del usuario y un conjunto de ontologías. El archivo DICOM corresponde a la imagen médica, y está compuesto por metadatos que describen las características de la imagen, así como la región sub-volumétrica del área que enmarca el tejido de interés y un punto semilla. Los datos del usuario indican el tejido que se quiere resaltar de la imagen y las propiedades ópticas a utilizar para su visualización. Suponiendo que el área que contiene al tejido de interés y el punto semilla pueden ser definidos manualmente por un experto radiólogo o por una máquina de aprendizaje como la propuesta por Criminisi et al. [8]. Nosotros estudiamos la calidad de ANISE en tres imágenes médicas de tomografías computarizada del cráneo. Estas imágenes son comúnmente utilizadas en el área de computación gráfica para estudiar la calidad de las tareas segmentación y *rendering*[14]. El estudio tiene como objetivo mostrar la calidad de la visualización cuando los dientes son visualizados en un color particular. Dado que la densidad de los dientes es la misma que la de otros tejidos, la visualización puede ser no precisa y nuestro objetivo es estudiar el impacto de la semántica codificada en las ontologías médicas FMA y RadLex en la calidad de la

[12] http://sig.biostr.washington.edu/projects/fm/AboutFM.html
[13] https://www.rsna.org/RadLex.aspx
[14] http://www9.informatik.uni-erlangen.de/External/vollib

visualización. La idea es que ANISE hace uso de un sistema de razonamiento sobre lógica imprecisa y el sistema manejador de ontologías de Jena, para determinar el órgano o tejido encerrado por en las diferentes regiones que conforman la imagen. La Figura 4 ilustra el resultado del proceso de visualización tradicional y el implementado en ANISE. Figuras 4 (a), (b) and (c) muestra las imágenes estudiadas, mientras que las Figuras 4 (d), (e) and (f) presentan las imágenes donde los dientes se visualizan de verde haciendo uso del proceso tradicional de visualización. Finalmente, las Figuras 4 (g), (h) and (i) ilustran las imágenes resultantes de aplicar las técnicas propuestas. Como puede observarse, ANISE permite realizar una visualización más precisa de los tejidos que pertenecen a los dientes.

IV. Conclusiones y Trabajos Futuros

Presentamos SEPIAS, una arquitectura que integra herramientas para consultar, curar, analizar y visualizar datos en la Nube de los Datos Abiertos Enlazados. Abordamos los retos de escalar conjuntos de datos enlazados muy grandes, gestionar, clasificar y procesar federaciones de *endpoints* de SPARQL. Además, problemas como la limpieza de los datos enlazados, descubrir patrones de conectividad y visualizar imágenes médicas han sido considerados. En todos los casos la semántica codificada en los conjuntos de datos fueron usadas para mejorar la calidad de los técnicas implementadas. Reportamos estudios experimentales que revelan el rendimiento y la calidad de los enfoques propuestos. Resultados empíricos iniciales corroboran nuestra hipótesis que la semántica codificada en las ontologías y en los conjuntos de datos son importantes en cada una de las tareas que intentamos resolver. Los efectos de la semántica no sólo pueden ser observados en el tiempo de ejecución de una consulta, sino también en la calidad de la salida producida por cada herramienta.

En el futuro planificamos extender las técnicas de consultas y clasificación propuestas para tratar datos dinámicos, ej., datos de trasmisión producidos por redes de sensores. Además, tenemos previsto definir otras métricas de similitud para identificar la relación en los datos enlazados que están anotados con términos de diferentes ontologías. Finalmente, ANISE será extendido con un nueva estrategia de renderizado y visualización. Asimismo, serán agregadas nuevas reglas de segmentación que describen propiedades particulares de nuevos tejidos. En consecuencia, ANISE será capaz de segmentar nuevos tejidos, por ejemplo, vasos sanguíneos.

Agradecimiento

Esta investigación ha sido parcialmente apoyada por DID-USB.

References

[1] M. Acosta, M.-E. Vidal, T. Lampo, J. Castillo, and E. Ruckhaus, *ANAPSID: AN Adaptive query ProcesSing engIne for SPARQL enDpoints*, in proceedings of the International Semantic Web Conference (ISWC), Bonn, Germany, 2011.

[2] A. Alvarado, O. Baldizan, M. Goncalves, and M.-E. Vidal, *FOPA: A Final Object Pruning Algorithm to Efficiently Produce Skyline Points*, in proceedings of the 24th International Conference of Database and Expert Systems Applications (DEXA), Prague, Czech Republic, 2013.

[3] P. Anderson, A. Thor, J. Benik, L. Raschid, and M.-E. Vidal, *PAnG: Finding Patterns in Annotation Graphs*, in proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2012, Scottsdale, USA, 2012.

[4] W.-T. Balke, U. Guntzer, and J. X. Zheng, *Efficient Distributed Skylining for Web Information Systems*, in proceedings of the International Conference on Extending Database Technology (EDBT), Heraklion, Greece, 2004.

[5] A. Baranya, L. Landaeta, A. L. Cruz, and M.-E. Vidal, *A Workflow for Improving Medical Visualization of Semantically Annotated CT-Images*, in proceedings of the Joint Workshop on Semantic Technologies Applied to Biomedical Informatics and Individualized Medicine (SATBI+SWIM) collocated with the International Semantic Web Conference (ISWC), Boston, USA, 2012.

[6] J. Benik, C. Chang, L. Raschid, M. E. Vidal, G. Palma, and A. Thor, *Finding Cross Genome Patterns in Annotation Graphs*, in proceedings of Data Integration in the Life Sciences (DILS), College Park, USA, 2012.

[7] L. Chen, S. Gao, and K. Anyanwu, *Efficiently Evaluating Skyline Queries on RDF Databases*, in proceedings of the Extended Semantic Web Conference (ESWC) 2011, Heraklion, Greece, 2011.

[8] A. Criminisi, J. Shotton, and E. Konukoglu, *Decision Forests: A Unified Framework for Classification, Regression, Density Estimation, Manifold Learning and Semi-Supervised Learning*, Foundations and Trends in Computer Graphics and Vision, vol. 7, no. 2-3, 2012.

[9] A. Kimmig, S. H. Bach, M. Broecheler, B. Huang, and L. Getoor, *A Short Introduction to Probabilistic Soft Logic*, in NIPS Workshop on Probabilistic Programming: Foundations and Applications, 2012.

[10] B. McInnes, T. Pedersen, and S. Pakhomov, *UMLS-Interface and UMLS-Similarity: Open Source Software for Measuring Paths and Semantic Similarity*, in proceedings of the AMIA Symposium, 2009.

[11] G. Montoya, M.-E. Vidal, and M. Acosta, *A Heuristic-Based Approach for Planning Federated SPARQL Queries*, in proceedings of the Third International Workshop on Consuming Linked Data (COLD) collocated with the International Semantic Web Conference (ISWC), Boston, USA, 2012.

[12] G. Montoya, M.-E. Vidal, and M. Acosta, *DEFENDER: a DEcomposer For quEries agaiNst feDERations of endpoints*, in proceedings of Workshops and Demo Papers at the Extended Semantic Web Conference (ESWC), Heraklion, Greece, 2012.

[13] G. Palma, M.-E. Vidal, L. Raschid, and A. Thor, *Exploiting Semantics from Ontologies and Shared Annotations to Partition Linked Data*, in proceedings of Data Integration in the Life Sciences (DILS), Lisbon, Portugal, 2014.

[14] G. Palma, M.-E. Vidal, E. Haag, L. Raschid, and A. Thor, *Measuring Relatedness between Scientific Entities in Annotation Datasets*, in ACM Conference on Bioinformatics, Computational Biology and Biomedical Informatics. ACM-BCB, Washington, USA, 2013.

[15] T. Pedersen, S. Pakhomov, S. Patwardhan, and C. Chute, *Measures of Semantic Similarity and Relatedness in the Biomedical Domain*, Journal of Biomedical Informatics, vol. 40, no. 3, 2007.

[16] M. Schmidt, O. Görlitz, P. Haase, G. Ladwig, A. Schwarte, and T. Tran, *FedBench: A Benchmark Suite for Federated Semantic Data Query*, in processings of the International Semantic Web Conference (ISWC), Bonn, Germany, 2011.

[17] A. Schwarte, P. Haase, K. Hose, R. Schenkel, and M. Schmidt, *FedX: Optimization Techniques for Federated Query Processing on Linked Data*, in proceedings of the International Semantic Web Conference (ISWC), Bonn, Germany, 2011.