



Ingeniare. Revista Chilena de Ingeniería

ISSN: 0718-3291

facing@uta.cl

Universidad de Tarapacá

Chile

Rodríguez, Tania; Aguilar, Jose

Implementación del Marco Ontológico Dinámico Semántico

Ingeniare. Revista Chilena de Ingeniería, vol. 25, núm. 3, septiembre, 2017, pp. 430-448

Universidad de Tarapacá

Arica, Chile

Disponible en: <http://www.redalyc.org/articulo.oa?id=77252700007>

- ▶ Cómo citar el artículo
- ▶ Número completo
- ▶ Más información del artículo
- ▶ Página de la revista en redalyc.org

redalyc.org

Sistema de Información Científica

Red de Revistas Científicas de América Latina, el Caribe, España y Portugal
Proyecto académico sin fines de lucro, desarrollado bajo la iniciativa de acceso abierto

Implementación del Marco Ontológico Dinámico Semántico

Implementation of the Dynamic Semantic Ontological Framework

Taniana Rodríguez¹* Jose Aguilar¹

Recibido 13 de enero de 2016, aceptado 25 de julio de 2016

Received: January 13, 2016 Accepted: July 25, 2016

RESUMEN

En este trabajo se implementa el Marco Ontológico Dinámico Semántico (MODS) especificado en [1]. El MODS permite formalizar e interpretar las consultas realizadas por los usuarios en lenguaje natural. Para ello, MODS transforma la consulta en un formato de Representación del Significado de la Consulta (RSC), el cual está basado en un lenguaje ontológico (OWL) para poder ser usado por la Web Semántica, utilizando los diferentes componentes que lo integran. El MODS está compuesto por un lexicón, una ontología lingüística, una ontología de tareas y una ontología de dominio. De esta manera, el MODS utiliza mecanismos de la semántica ontológica y del procesamiento del lenguaje natural, para procesar las consultas de los usuarios. En específico, en este artículo se define ontológicamente, usando lógica descriptiva, a cada uno de los componentes del MODS. Además, se realiza la implementación ontológica de cada componente, usando en cada caso adicionalmente, específicas librerías. Por ejemplo, *OpenNLP* es usada por la ontología de tareas para el procesamiento de textos, el Framework *Jena* es usado por todas las ontologías del MODS para la manipulación de las mismas, y *Pellet* es usado como el razonar en las ontologías del MODS. Finalmente, se presentan experimentos usando dicha implementación del MODS.

Palabras clave: Ontologías, procesamiento de lenguaje natural, Web Semántica.

ABSTRACT

In this work the Dynamic Semantic Ontological Framework (MODS) specified in [1] is implemented. MODS formalizes and interprets queries made by users in natural language. For this, MODS transforms the query into a format of representation of the meaning of the Query, which is based on an ontology language (OWL) in order to be used by the Semantic Web, using the different components that comprise it. MODS is composed of a lexicon, a linguistic ontology, an ontology of tasks and a domain ontology. Thus, MODS uses mechanisms of the ontological semantic and of the natural language processing, to process the user queries. Specifically, this article defines, using ontologies on descriptive logic, each component of MODS. In addition, the ontological implementation of each component is carried out, using in each case additionally specific libraries. For example, OpenNLP is used by the ontology of tasks for the words processing, the Jena Framework is used by all the ontologies of MODS to handle them, and Pellet is used as the reasoning mechanism in the ontologies of MODS. Finally, experiments are presented using this implementation of MODS.

Keywords: Ontology, natural language processing, Semantic Web.

¹ Centro de Estudios en Microelectrónica y Sistemas Distribuidos (CEMISID). Universidad de Los Andes, Mérida, Venezuela.
E-mail: taniana@ula.ve; aguilar@ula.ve

* Autor de correspondencia

INTRODUCCIÓN

En este trabajo se presenta la especificación ontológica y la implementación del Marco Ontológico Dinámico (MODS) de la arquitectura propuesta en [1], con el fin de que sea un interpretador de consultas en lenguaje natural para la Web. MODS permite formalizar e interpretar las consultas en lenguaje natural enviadas a la Web. En este trabajo se describen ontológicamente los diferentes componentes de esa arquitectura, usando lógica descriptiva. Después, se realiza la implementación de cada una de las ontologías utilizadas por el MODS, para culminar con experimentos realizados sobre dicha implementación. Así, en este trabajo se presenta la especificación ontológica, la implementación y la experimentación de la propuesta presentada en [1].

Entre los trabajos similares que se han realizado a este, se pueden mencionar los siguientes: El sistema MESIA es un modelo computacional, para la extracción selectiva de información de textos cortos [2]. En particular, MESIA facilita la comunicación del usuario con el motor de búsqueda AltaVista (<http://www.altavista.com>). El sistema recibe la consulta del usuario escrita en lenguaje natural, y la convierte en una consulta booleana. Además, realiza una expansión de la consulta mediante recursos lingüísticos. Finalmente, durante el proceso de búsqueda se produce una expansión de los resultados, después de la búsqueda, MESIA incorpora información sobre el dominio, permitiendo la expansión semántica de los resultados. Además, una vez identificado el tema de la consulta, a los resultados obtenidos se le añaden enlaces sobre asuntos relacionados con dicho tema [2]. A continuación, se describe brevemente la arquitectura MESIA. El sistema utiliza dos recursos lingüísticos para la expansión de la consulta, que son: ARIES, es un analizador léxico morfológico para el español, el que está formado por un léxico español de aproximadamente 38.500 lemas y 600 morfemas flexivos [3]. Para su integración con MESIA se realizó la traducción de su base léxica a PROLOG, y EuroWordNet (EWN), es una base de datos semántica en español, que permite obtener sinónimos de cada palabra [2]. Para la integración de ARIES con EWN se usaron patrones sintácticos, que permiten segmentar frases, es decir, descomponer la consulta en sintagmas nominales, preposicionales y verbales, con el fin de resolver la ambigüedad del proceso morfológico producido por ARIES,

y permitir el análisis sintáctico para extraer los términos relevantes de la consulta. Esos términos luego son expandidos por EWN, y el resultado es enviado en paralelo al motor de búsqueda AltaVista y a la ontología de MESIA. Dicha ontología, implementada en XML, consiste en un árbol de conceptos relacionados con cada uno de los temas que pueden encontrarse en un dominio dado, lo que permite incorporar información semántica sobre el dominio al proceso de búsqueda. Los nodos son palabras clave de los temas, e incluyen un conjunto de enlaces relacionados con los mismos [4]. Los resultados, tanto del motor de búsqueda como de la ontología, se ordenan y se mezclan, para luego ser presentados al usuario.

El sistema QACID es un sistema de búsqueda de respuestas basado en ontologías, que permite asociar preguntas en lenguaje natural con una representación formal de datos [5]. QACID se compone de dos núcleos principales: una base de conocimiento; y un módulo de implicación textual. Para crear la base de conocimiento, el sistema recoge preguntas de usuarios realizadas sobre un dominio específico. Dichas preguntas son analizadas y agrupadas en función de la información que solicitan. A cada agrupación se le asocia manualmente una sentencia SPARQL, quien describe un patrón de consulta específico. El módulo de implicación textual relaciona las preguntas realizadas al sistema en lenguaje natural con las preguntas agrupadas anteriormente, con el objetivo de asociar a cada pregunta su sentencia SPARQL correspondiente. Este módulo está compuesto por deducciones que infieren la relación entre dichas preguntas y las sentencias SPARQL [5]. A continuación, se describe brevemente la arquitectura de QACID: El proceso comienza con el análisis de la pregunta, con el objetivo de obtener la representación formal de la misma (se analiza morfológicamente la pregunta y se detectan sus entidades, utilizando para ello un diccionario de conceptos ontológicos). La representación formal de la consulta es procesada, con el fin de determinar las implicaciones semánticas en la misma, y el conjunto de patrones que contiene en la base de conocimiento. Si el proceso termina con éxito, se obtiene al menos una sentencia SPARQL, la cual permite obtener la respuesta a dicha consulta.

PowerAqua es un sistema de preguntas y respuestas, el que toma como entrada una consulta en lenguaje

natural, y es capaz de devolver respuestas procedentes de cualquier lugar de la Web Semántica [6]. La arquitectura de PowerAqua está compuesta por tres componentes: un componente lingüístico (analiza la consulta en lenguaje natural, dando como resultado un conjunto lingüístico triple, llamado Query-Triples (QTs), que identifica las asociaciones entre el conjunto de palabras en la oración), un componente llamado PowerMap (recibe los QTs producidos por el componente lingüístico, e identifica la semántica de la palabra para poder responder a la consulta dada. Así, establece las relaciones entre los términos QTs y la semántica de las palabras), y finalmente, un componente de mezcla y de categorización de las relaciones definidas por el componente anterior (genera la respuesta final) [6].

WebQA es un sistema de búsqueda de respuestas para el español, que explota la Web como fuente de información [7]. El sistema responde preguntas que esperan una respuesta precisa y concreta, como las que comienzan con “Cuándo” o “Quién”. Este sistema utiliza el etiquetador sintáctico Freeling, para encontrar la categoría léxica de cada palabra de la pregunta. Además, utiliza patrones que permiten caracterizar los tipos de preguntas y de respuestas, generando reformulaciones de las consultas de los usuarios, que son enviadas como consultas a un motor de búsqueda. WebQA aplica tres técnicas para reformular las consultas, las que son [7]: un reformulador según patrones preestablecidos, un reformulador en base a la permutación de las dos primeras palabras, y un reformulador en base a la conjunción de palabras claves.

SWSNL está construido sobre el paradigma de la Web Semántica, y usa una ontología, con el fin de almacenar su base de conocimientos [8]. Permite que los usuarios realicen sus preguntas en lenguaje natural, cuenta con un módulo de interpretación de la consulta en lenguaje natural, donde realiza un preprocesamiento, reconocimiento de entidades y utiliza un modelo de análisis semántico estocástico para interpretar la consulta. Ese módulo realiza una interpretación semántica de la consulta y genera la consulta en SPARQL. Finalmente, el módulo de resultados de la búsqueda es el encargado de obtener los resultados [8].

Existe un gran número de trabajos recientes vinculados al procesamiento de las consultas en

lenguaje natural para la web, algunos de ellos para idiomas distintos al inglés o español [13-14], o basados en el enlazado de datos (linked data) usando lógica descriptiva [15-17].

En particular, nosotros desarrollamos un Marco Ontológico Dinámico Semántico, cuyas principales características son:

- Permite usar el lenguaje natural español para realizar consultas sobre la Web.
- Es basado en un conjunto de ontologías para el procesamiento de la consulta en lenguaje natural.
- Posee un componente de aprendizaje ontológico, que le permite adquirir nuevo conocimiento y actualizar sus ontologías. Este aspecto es el que le permite adaptarse a su entorno.
- Realiza una expansión semántica de las consultas usando el contenido de sus ontologías.
- Explota el contenido semántico sobre la Web en procesos de razonamiento automático y de aprendizaje, con el fin de optimizar los procesos de búsqueda y aprender sobre el uso y el contenido de la Web.

Algunos de los trabajos previos tienen algunas de esas características, pero ninguno de ellos tiene todas esas cualidades juntas.

Este trabajo está organizado en la siguiente forma, en la siguiente sección se describe el Marco Ontológico Dinámico, sus componentes, el proceso de interpretación de la consulta. Posteriormente, en Experimentos se muestra un ejemplo de interpretación de la consulta en el MODS, y se realiza una comparación con trabajos previos. Finalmente, se presentan las Conclusiones.

MARCO ONTOLÓGICO DINÁMICO SEMÁNTICO

El MODS es una propuesta novedosa, que permite el análisis y la realización de consultas en lenguaje natural en la Web Semántica. La consulta para el MODS, más que una petición de información, es un elemento cargado de información útil para formarse una idea del tipo de usuario y aproximarse, de manera sucesiva, a una respuesta que cada vez más satisface las necesidades del mismo. MODS tiene el desafío de formalizar e interpretar la consulta realizada

por el usuario en lenguaje natural, y refinar sus esquemas internos frente a la dinámica de la Web, para tratar futuras consultas (para eso requiere de mecanismos de adaptación).

La arquitectura general del MODS es presentada en la Figura 1. Su descripción detallada es la siguiente: la ontología de tareas modela el procesamiento de la consulta en lenguaje natural (análisis léxico-morfológico, análisis sintáctico-semántico, análisis pragmático); la ontología lingüística especifica la gramática del lenguaje español, y cuenta con una extensión de derivaciones coloquiales y con un lexicón para caracterizar al lenguaje español, que a su vez contiene un onomasticon; finalmente, la ontología interpretativa modela el conocimiento sobre el contexto específico del usuario (es una ontología de alto nivel, con especializaciones/extensiones basadas en ontologías de dominio externas al MODS). La ontología interpretativa, entre otras cosas, tiene una ontología del usuario, que describe el uso del sistema que va haciendo cada usuario, lo que permite incorporar a la consulta formal las características propias del usuario (contextualización), para intentar delimitar la respuesta de la Web. Finalmente, otro componente clave para la adaptabilidad del MODS a la dinámica de la Web y del usuario, es el componente de aprendizaje de ontologías, cuyo fin es permitir que las ontologías evolucionen a la par con la usabilidad del sistema.

Descripción de los componentes

Como se puede observar en la Figura 1, el MODS consta de cuatro ontologías, cada una ha sido conceptualizada usando la herramienta *Protégé*, la ontología del lexicón que describe los términos, la ontología lingüística que describe la gramática del español [9], la ontología interpretativa que describe al dominio donde se esté usando MODS (por ejemplo, universitario), y la ontología de tareas conceptualiza el proceso de análisis de la consulta. Pasemos a describir a cada una de las ontologías.

Ontología del lexicón

El lexicón del MODS caracteriza las palabras; en la ontología de lexicón se describe la estructura morfológica de ellas (ver Figura 2). Las palabras se agrupan en categorías (verbo, sustantivo, adverbio y adjetivo); además, se identifican por sus lemas (el lema para los sustantivos es usualmente el masculino y singular, mientras que para los verbos es el infinitivo) y lexemas (raíz de la palabra). El lexicón usa el estándar de codificación léxica EAGLES (<http://www.cs.upc.edu/~nlp/tools/parole-sp.html>).

También, el lexicón del MODS establece las relaciones semánticas entre las palabras: sinónimos (synonymy), hipónimo (hyponym), antónimos (antonymy), meronimias (meronymy) y la polisemia.

Por otro lado, el MODS cuenta con un onomasticon (ver Figura 3), el que representa los nombres

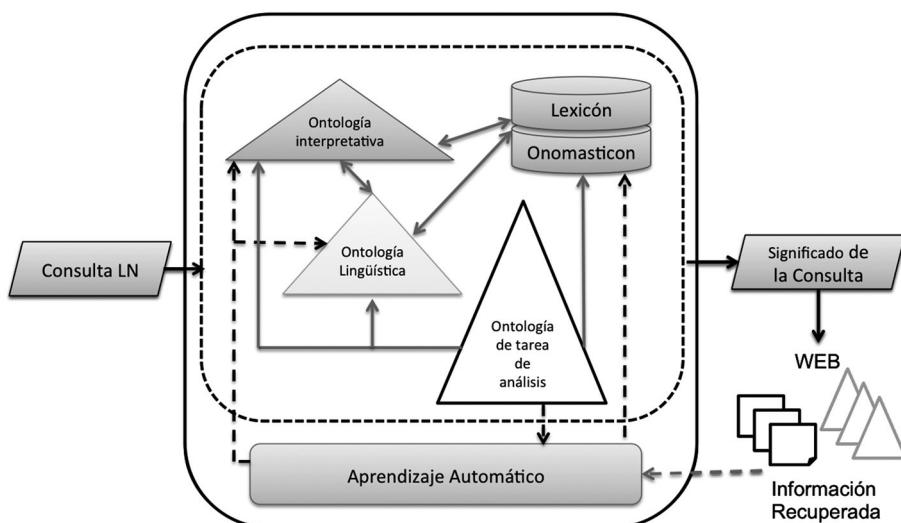


Figura 1. Marco Ontológico Dinámico Semántico.

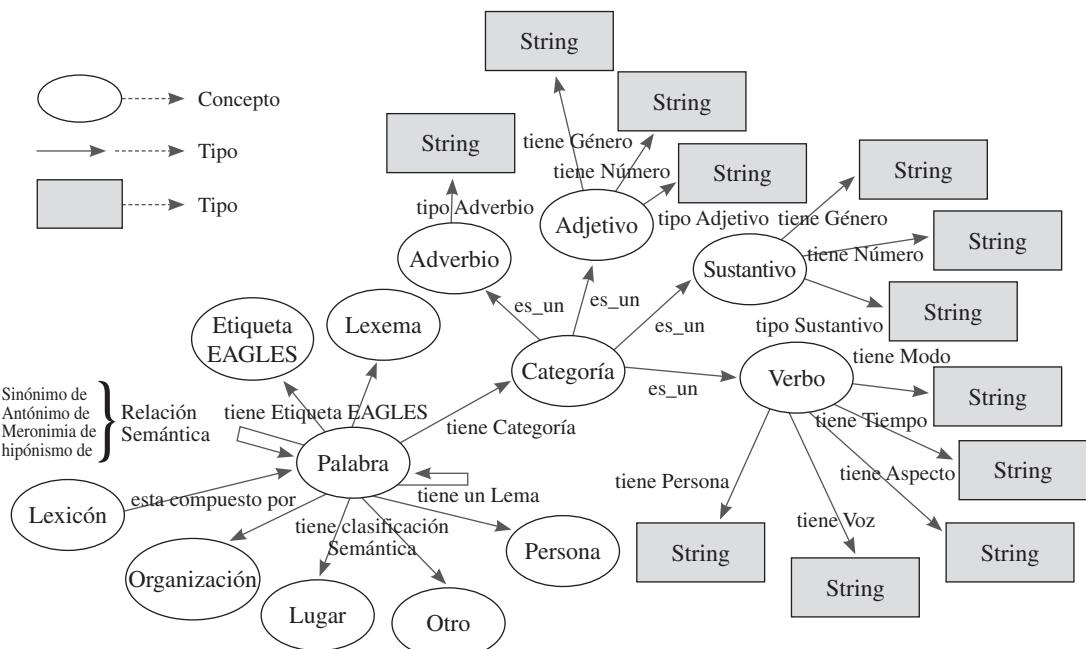


Figura 2. Esquema Conceptual del Lexicón.

propios, y está compuesto por palabras simples y compuestas de tipo sustantivo. El onomasticon usa el estándar de codificación léxica EAGLES para la caracterización semántica de cada nombre propio, definida por: organización, lugar, persona y otro.

Ontología lingüística

La ontología lingüística es la encargada de especificar la gramática del lenguaje. Su función es apoyar el procesamiento del lenguaje natural. En el caso de este trabajo, es usada para realizar

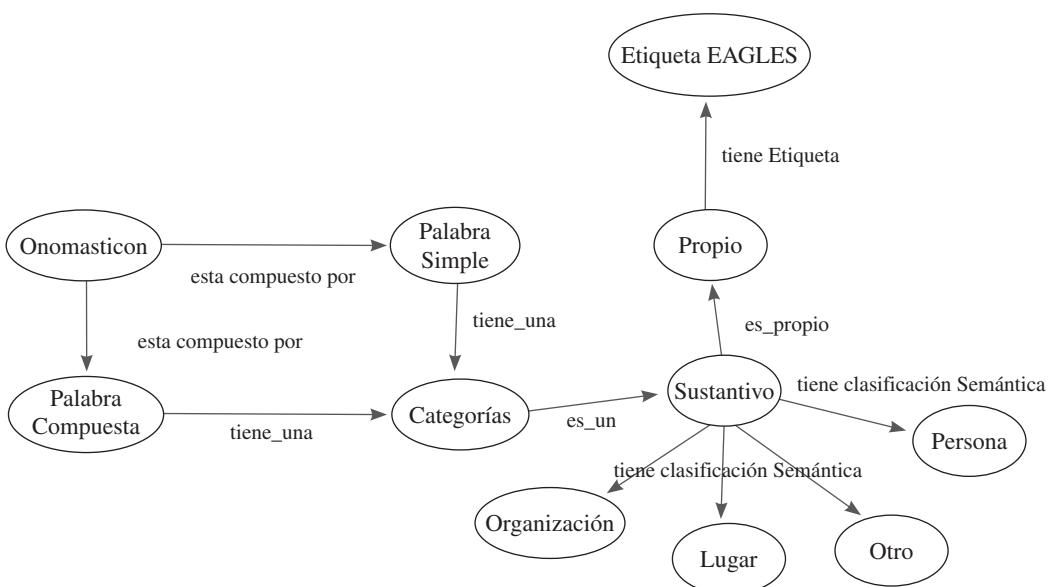


Figura 3. Esquema Conceptual del Onomasticon con las relaciones semánticas.

el análisis sintáctico y semántico del lenguaje español. Para ello, ella describe de manera general a las unidades léxicas como objetos lingüísticos en una base de datos léxica, y las relaciones entre ellas en una jerarquía conceptual (taxonomía ontológica). Por lo tanto, la ontología lingüística es una representación de conceptos lingüísticos y sus relaciones en un dominio específico, en este caso, el lenguaje español. La ontología lingüística utiliza OWL para las especificaciones de las estructuras sintácticas y semánticas. En general, en la ontología lingüística se formaliza la gramática del español utilizando el enfoque denominado *constituyente*, como se muestra en la Tabla 1. A continuación describimos cada uno de los componentes que la integran:

– Conceptos

- Categorías: los conceptos se agrupan en categorías conocidas: sustantivos, adverbios, adjetivos y verbos.
- Sustantivos: son los pronombres y nombres que denotan cosas.
- Verbos: denotan sucesos o eventos. A su vez, los verbos se clasifican en verbos transitivos, verbos copulativos y verbos intransitivos.
- Adjetivos: son los que modifican los sustantivos.
- Adverbios: son los que modifican el verbo.
- Frase: las frases se clasifican en Frase Nominal, Frase Adverbial y Frase Verbal.
- Oración: está compuesta por una Frase Nominal seguida por Frase Verbal.
- Estructura Sintáctica: cada categoría tiene una estructura sintáctica, la que se define mediante reglas, por ejemplo: oración → FraseNominal FraseVerbal
- Estructura Semántica: cada categoría tiene una estructura semántica.

– Relaciones

- Tiene: relaciona los conceptos sustantivos, verbos, adverbios y adjetivos con los Conceptos Estructura Sintáctica y Estructura Semántica. A todos esos conceptos se les debe definir tanto su estructura sintáctica como su estructura semántica.

– Axiomas

- Los axiomas son un conjunto de reglas que verifica si las sentencias son válidas, tanto sintáctica como semánticamente.

A continuación, se muestra un ejemplo del conocimiento guardado en esta ontología:

- “Un Verbo Copulativo es un verbo que iguala el sujeto con el predicado, el que tiene una estructura sintáctica como semántica”. Por lo tanto, en la ontología lingüística se define de la siguiente manera:

$\text{VerbosCopulativo} \subseteq \text{Verbo} \cap \text{tiene_EstructuraSintactica} \cap \text{tiene_EstructuraSemantica}$.

Por otro lado, para construir la estructura semántica se usan los atributos usados en la semántica ontológica [10] que se mencionan a continuación:

- Agente: el participante designado por el predicado para hacer o causar la acción. Por lo general, el sujeto del verbo transitivo es el agente. Los agentes pueden ser las organizaciones, personas, animales, etcétera.
- Paciente: participante afectado por la acción
- Tema: el participante que cambia de lugar, condición o estado, o que está en determinado estado o posición, es decir, el participante manipulado por la acción. Por lo general, el tema son los objetos directos de los verbos transitivos, sujetos en las oraciones intransitivas, o los complementos verbales.
- Instrumento: El objeto o evento utilizado para llevar a cabo la acción.
- Experimentador: el participante que se informa de algo o que experimenta un estado psicológico por lo expresado en el predicado.
- Fuente: objeto desde el que ocurre el movimiento
- Meta: objeto hacia el que el movimiento es dirigido.
- Lugar: sitio en que la acción o estado descrito por el predicado tiene lugar.
- Beneficiario: el participante que se beneficia de la acción expresada por el verbo.

Partiendo de lo anterior, un ejemplo de la estructura semántica de los verbos transitivos, para dos de sus casos, es:

- $\text{EstructuraSemántica} \subseteq \text{Agente} \cap \text{Tema} \cap \text{Beneficiario}$
- $\text{EstructuraSemántica} \subseteq \text{Agente} \cap \text{Tema}$

Y las estructuras sintácticas para esos dos casos de verbos transitivos son las siguientes:

Tabla 1. Ejemplos de la estructura sintáctica y semántica de los verbos transitivos.

Verbo	Estructura Semántica	Estructura Sintáctica
Dar	Agente \cap Tema \cap Beneficiario Ejemplo: Juan(Agente) le dio las notas(Tema) a los estudiantes(Beneficiario)	Sujeto \cap ObjetoDirecto \cap ObjetoIndirecto Ejemplo: Juan(Sujeto) le dio las notas(Objeto Directo) a los estudiantes (Objeto Indirecto)
Comer	Agente \cap Paciente Ejemplo: Pedro(Agente) comió galletas(Paciente)	Sujeto \cap ObjetoDirecto Ejemplo: Pedro(Sujeto) comió galletas(Objeto directo)
Encontrar	Como localización Agente \cap Tema \cap Lugar Ejemplo: Juan(Agente) encontró a Pedro(Tema) en Mérida(Lugar)	Como localización Sujeto \cap ObjetoDirecto \cap ObjetoIndirecto Ejemplo: Juan(Sujeto) encontró a Pedro(Objeto directo) en Mérida(Objeto Indirecto)

- EstructuraSintáctica \subseteq Sujeto \cap ObjetoDirecto \cap ObjetoIndirecto
- EstructuraSintáctica \subseteq Sujeto \cap ObjetoDirecto

A continuación, se presentan algunos ejemplos de estructuras sintácticas y semánticas para varios tipos de verbos transitivos.

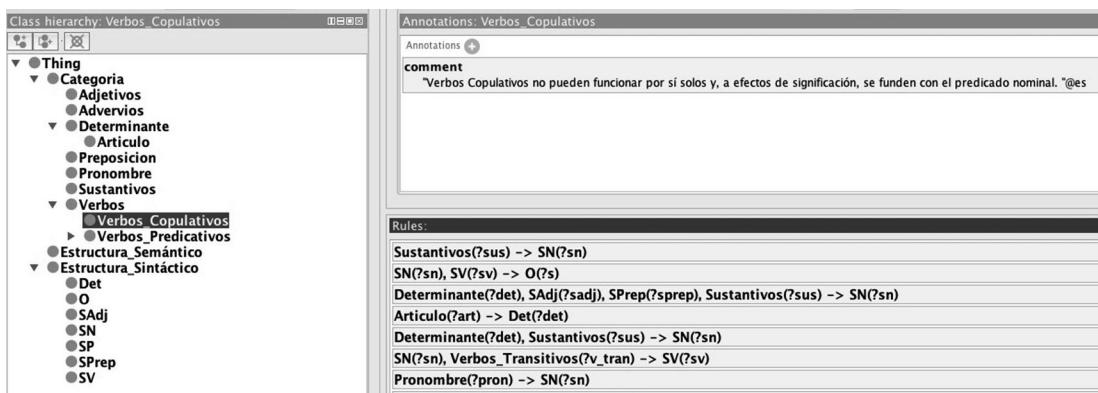
Como vemos, de esta ontología se derivan un conjunto de reglas para verificar si las consultas son válidas, tanto sintáctica como semánticamente (ver parte de ellas en la Tabla 2, descritas como expresiones regulares). En la Figura 4 se muestra parte de la ontología lingüística en *Protégé*, donde se observa la jerarquía de conceptos y algunas de las reglas.

Ontología interpretativa

La ontología interpretativa está basada en la ontología del proyecto mikrokosmo [10] para describir el contexto, y en [11] para describir el perfil del

Tabla 2. Parte de la gramática usada en la ontología lingüística.

Constituyente	Reglas
Oración -> O	O -> (SN) SV
Sintagma Nominal -> SN	SN -> { (Det) N (SP) } Pron Art N Sno
Sintagma Verbal -> SV	SP -> P SN
Sintagma Preposicional -> SP	Sno -> Np
Sintagma Nombre -> SNo	SV -> V (SN) V SP V
Determinante -> Det	SN SP
Nombre -> N	V->Vi Vt Vc
Nombre Común -> Nc	N->Nc Np
Nombre Propio -> Np	Subj->Np
Pronombre -> Pron	Ag->Subj
Preposición -> P	
Verbo intransitivo -> Vi	
Verbo transitivo -> Vt	
Verbo copulativo -> Vc	
Artículo -> Ar	
Sujeto -> Subj	
Agente -> Ag	

Figura 4. Ontología lingüística en *Protégé*.

usuario. En particular, para describir el contexto, la ontología interpretativa se compone de:

a) Conceptos

- Entidades: representa objetos físicos como abstractos (normalmente son los sustantivos, adjetivos y adverbios).
- Eventos: representa una acción (normalmente son verbos).
- Relaciones: indican las diferentes relaciones que pueden existir entre los conceptos definidos previamente, o propiedades que los mismos pueden tener.

La clasificación semántica de las Entidades y de los Eventos en el MODS se muestra en las Tablas 3 y 4, respectivamente:

Tabla 3. Clasificación Semántica de las Entidades.

Entidades	Definición
Abstractas	Pueden ser números, conjuntos, definiciones
Concretas	Objetos físicos, o que se pueden definir en algo específico (por ejemplo, el planeta Venus, ese árbol, aquella persona, organizaciones, etc.)

El concepto relación puede ser de dos tipos: relaciones para vincular entidades, entidades y eventos; o propiedades para describir atributos de los conceptos. Algunos ejemplos de las relaciones que vinculan a entidades y eventos son mostrados en la Tabla 5:

b) Relaciones

- Las relaciones entre los conceptos de esta ontología son del tipo “tiene”. Por ejemplo, una entidad tiene relaciones con otras entidades, con eventos, una entidad tiene propiedades, etc.
- La Figura 5 presenta parte del modelo de la ontología interpretativa con estos componentes. Además, un ejemplo de una instancia, para el caso de una entidad, es dado en la Figura 6. La instanciación de la ontología va ocurriendo en la medida que es usado el MODS. Específicamente, en la Figura 6 se tiene la instancia *Universidad de los Andes*, donde se puede observar que es una Universidad que se encuentra ubicada en Chile, Colombia y Venezuela, que es un objeto concreto de una organización educativa.

Tabla 4. Semántica de los Eventos.

Eventos del MODS
Comunicación
<ul style="list-style-type: none"> General (Decir, hablar) Valoración (Criticar, felicitar) Mandato (Suplicar, ordenar)
Comportamiento
<ul style="list-style-type: none"> General (Comportar) Relaciones Sociales (Acoger, casar) Fisiológica (Llorar, orinar) Vida (Hacer, matar)
Cambio
<ul style="list-style-type: none"> Destrucción Consumo (Comer, eliminar, gastar) Cuidado Personal (Lavar, cepillar) General (Cocinar, pintar) Modificación (Secar, hervir, romper) Creación (Crear, fabricar)
Mental
<ul style="list-style-type: none"> Percepción (Ver, escuchar) Sentimientos (Sentir) Sensación (Gustar, querer, doler) Cognición General (Entender, pensar) Creencia (Opinar, creer) Conocimiento (Recordar, saber)
Posesión
<ul style="list-style-type: none"> General Adeudar Deber Pertenencia Poseer Tener Carecer Transferencia Recibir Pagar Dar Cobrar

Tabla 5. Ejemplos de relaciones que vinculan a entidades y eventos.

Relaciones	Definición
Relación_binaria	Donde solo se relacionan dos conceptos. Por ejemplo. Pasajero compró boleto
EsParteDe	Son las relaciones de las partes que componen un todo. Por ejemplo el concepto apéndice es parte del concepto intestino.
Es_Sinónimo_de	Es la relación entre dos términos de significados (conceptos) similares e intercambiables en el discurso por pertenecer a la misma categoría sintáctica. Ejemplo: amplio/extenso, pelo/cabello, estimar/apreciar
Es_Polisemía_de	Es la relación entre un término que tiene varios significados, adquiridos por ampliación o restricción de su significado original, de modo que todos ellos están emparentados semánticamente.



Figura 5. Ejemplo de Ontología Interpretativa.

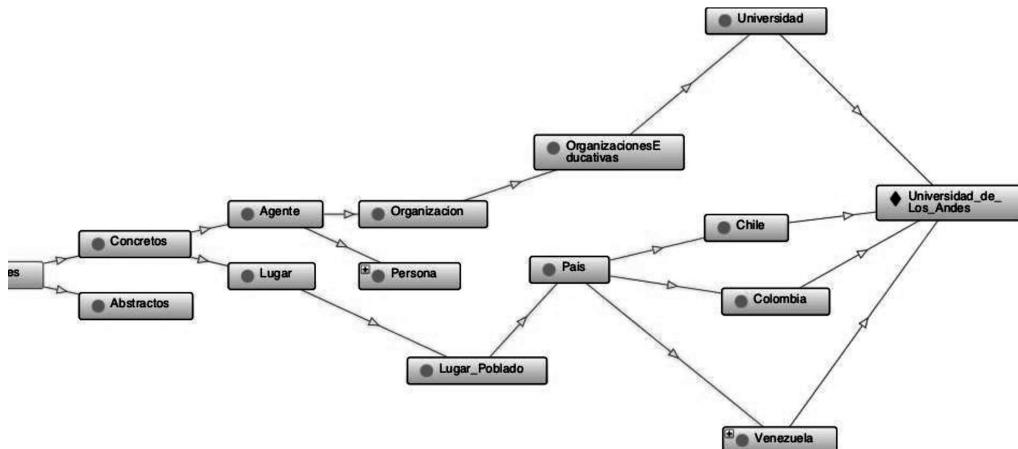


Figura 6. Ejemplo de instanciación de la Ontología Interpretativa.

Finalmente, la ontología interpretativa es completada con el perfil del usuario, como se muestra en la Figura 7 (se muestra en dos formatos diferentes). El perfil de usuario sirve para describir al usuario que hace la consulta. El usuario es descrito por sus preferencias, intereses, identidad, entre otras cosas. Además, los usuarios tienen un historial,

caracterizado por su ubicación, dispositivos usados, y tareas realizadas.

Ontologías de tareas

En cuanto a la ontología de tarea, se usa OWL como lenguaje de definición de la ontología. Los círculos de la Figura 8 representan los conceptos

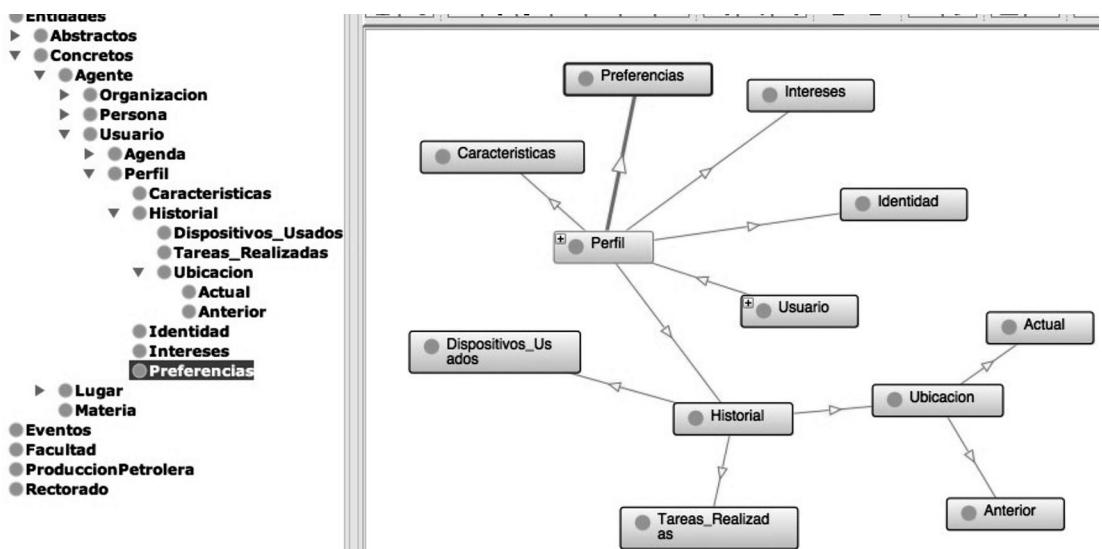


Figura 7. Modelo del perfil del usuario en la ontología interpretativa.

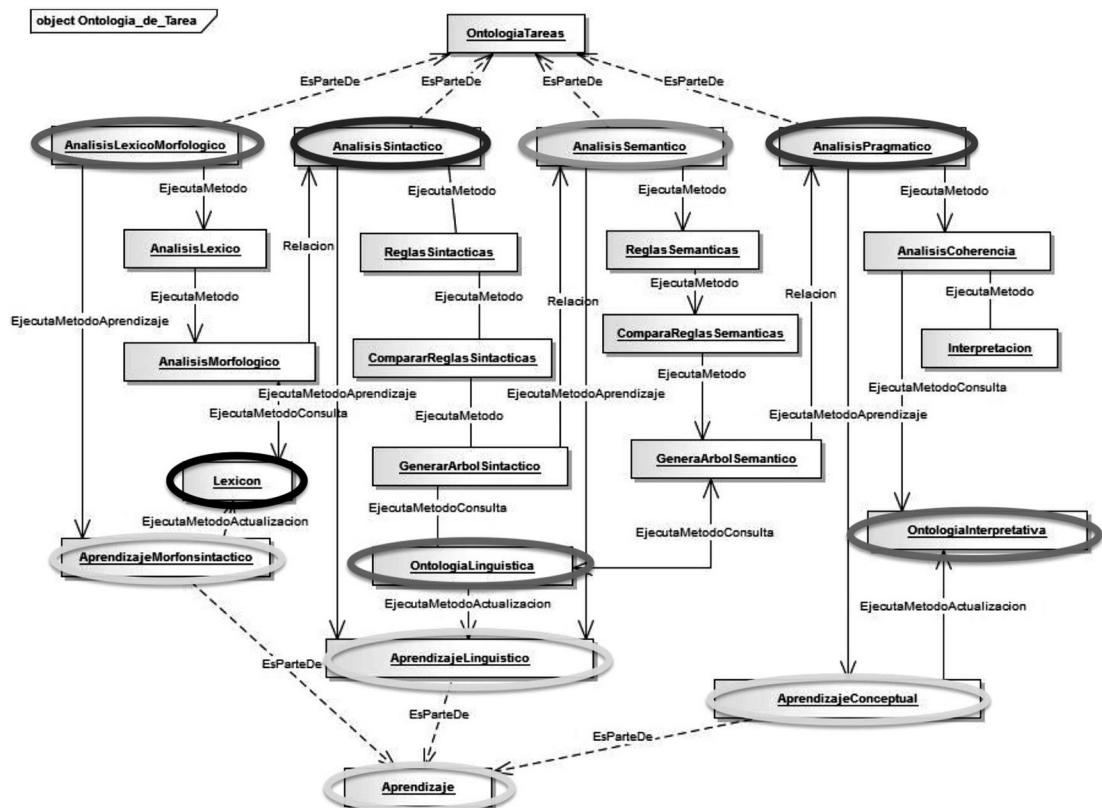


Figura 8. Ontología de Tareas (Integra las otras Ontologías de MODS y del componente de aprendizaje).

que conforman el diagrama de base taxonómico de la ontología de tarea. A continuación, se describen los elementos de esta ontología:

- Conceptos:
 - Representan las fases para el procesamiento del lenguaje natural, u otros conceptos requeridos por dicha tarea:
 - Análisis-léxico-morfológico: en esta fase se obtiene la información morfológica contenida en el lexicón de cada una de las palabras de la consulta.
 - Análisis-sintáctico: en esta fase se obtiene la estructura sintáctica, utilizando para ello una gramática (almacenada en la ontología lingüística).
 - Análisis-semántico: en esta fase se obtiene la estructura semántica (usando las reglas semánticas e información, guardadas en la ontología lingüística e interpretativa, respectivamente).
 - Análisis-pragmático: en base al contexto y perfil del usuario (definidos en la ontología interpretativa), esta fase obtiene la interpretación final de la consulta.
 - Conjunto de conceptos vinculados a las otras fuentes de conocimiento del MODS (Lexicón, Ontología lingüística, Ontología Interpretativa). Por medio de estos conceptos que invocan a esas otras fuentes de conocimiento, es que se da la integración ontológica.
 - Conjunto de conceptos que ayudan al procesamiento del lenguaje natural, que son los que se encargan de obtener las palabras, las morfologías, las reglas sintácticas y semánticas, los árboles de traducción sintácticas y semánticas, los análisis de coherencia y la interpretación de la consulta.
 - Finalmente, conceptos vinculados al componente de aprendizaje, que identifican las diferentes formas de aprendizaje que se pueden realizar en el MODS.
- EjecutaMetodoConsulta: Esta relación es la que consulta los otros niveles de conocimiento (Lexicón, Ontología Lingüística, Ontología Interpretativa) del MODS.
- EjecutaMetodoAprendizaje: Esta relación invoca el proceso de aprendizaje cuando EjecutaMetodoConsulta indica que no conoce el término, o la estructura sintáctica, o la estructura semántica, o el concepto.
- EjecutaMetodoActualizacion: Esta relación actualiza cualquier nivel de conocimiento (Ontología Lingüística, Ontología Interpretativa, lexicón) del MODS.
- Propiedades de los objetos:
 - ObjetivoMetodo: Esta propiedad es la descripción de cada método utilizado en las tareas de análisis.
 - EntradasMetodos: Son las entradas necesarias para la ejecución del método invocado.
 - SalidasMetodos: Son las salidas del método en ejecución.
- Axiomas:
 - La representación de la Ontología de Tareas se realizó con lógica descriptiva. A continuación, se muestra un ejemplo de un axioma en su base de conocimiento.

Analisis Lexico Morfologico \subseteq Tarea $\cap \forall$ Ejecuta Metodo. Analisis Lexico $\cap \forall$ Ejecuta Metodo Analisis Morfologico

Este axioma indica que para que se ejecute la tarea de AnalisisLexicoMorfologico, se deben ejecutar los métodos de Análisis Léxico y Morfológico.

Descripción detallada del MODS

Como se puede observar en la Figura 8, la ontología de tareas es la ontología que se encarga de integrar los otros tres niveles ontológicos que presenta la arquitectura (lexicón, lingüística e interpretativa). A continuación, se explica el proceso de integración en forma general, y su relación con los otros componentes del MODS:

- a) La primera tarea a realizar es el análisis léxico-morfológico, y se usa como componente de apoyo al lexicón. En esta tarea se obtiene la información morfológica de cada una de las palabras que se encuentran en la consulta, y con el onomasticon se determina si hay nombres propios

- en la consulta. El análisis léxico morfológico se divide en las siguientes sub-tareas: i) Análisis léxico, su principal función es recorrer la consulta de entrada y separarla en componentes léxicos (tokens en inglés); ii) análisis morfológico, el que se apoya en el lexicón, y consiste en determinar la forma, clase o categoría gramatical de cada palabra de la consulta.
- b) El siguiente paso es el análisis sintáctico, el que, a partir de la salida del proceso anterior, y con el fin de detectar oraciones o frases significativas para el lenguaje (en nuestro caso español), usa una gramática. El proceso de “traducción” compara las reglas que se encuentran en la ontología lingüística contra las palabras del texto de entrada. Cada regla que es instanciada por las palabras del texto agrega un elemento a la estructura, o la termina de generar. La estructura que se produce como resultado de estas instanciaciones de las reglas, es el “árbol de traducción”.
 - c) Despues, con la ontologías lingüística e interpretativa se realiza el análisis semántico, el que es el encargado de establecer qué combinaciones de significados de palabras individuales son posibles a la hora de crear un significado coherente en una oración, lo que puede reducir el número de posibles significados para cada palabra en una oración determinada. De esta forma, el analizador obtendrá la esencia de la pregunta. El análisis semántico representa uno de los elementos claves del “conocimiento” del MODS, y en función a su variedad y detalle será la riqueza del vocabulario, expresión, entendimiento, respuesta y utilidad, que el propio análisis ofrezca.
 - d) La última tarea de la ontología de tareas es el análisis pragmático o del contexto. Debido a que la semántica se ocupa del significado literal de las expresiones lingüísticas, la pragmática trata el significado adicional que adquieren las consultas en el contexto del usuario. Por lo tanto, el analizador pragmático utiliza la estructura semántica obtenida en la tarea anterior para desarrollar la interpretación final de la consulta, en función del contexto. En este nivel se analiza el discurso del usuario (es decir, los elementos lingüísticos que utiliza para comunicarse en la Web), cuál es su intención comunicativa, etc., usando para ello la metaontología interpretativa. En esta fase se

cubren aspectos como la identificación de objetos referenciados por determinados constituyentes de la frase (sintagmas nominales, pronombres, elementos elididos, etc.), el análisis de aspectos temporales, la identificación de la intención del usuario (temas y focos), para interpretar apropiadamente la consulta dentro del dominio específico vinculado al usuario. Así, el análisis pragmático realiza la interpretación final de la consulta en función del contexto y del perfil del usuario.

Otros componentes de la Figura 8 están vinculados al proceso de aprendizaje del MODS, que permiten mantener actualizado a todas las ontologías. La arquitectura de este componente es explicada en detalle en [12].

Proceso de interpretación de la consulta

Para entender el proceso de interpretación de la consulta realizado por la ontología de tarea, se realiza un ejemplo de una consulta sencilla, por ejemplo “Universidad de Los Andes”.

Como se mencionó en la sección anterior, el proceso comienza con una consulta en lenguaje natural. Seguidamente se realiza la fase de análisis léxico-morfológico, utilizando los recursos lingüísticos de las ontologías de Lexicón y Onomasticon, generando el grafo de la consulta mostrada en la Figura 9.

Como se puede observar en la Figura 9, la consulta es etiquetada usando la etiquetación léxica EAGLES, y además, es separada en tokens, donde el individuo *nil* es el fin de la consulta. También, indica que Universidad de Los Andes es un Nombre Propio.

Siguiendo con el proceso, se realiza el Análisis Sintáctico, donde se usa como recurso la ontología lingüística; este proceso recibe como entrada la salida del proceso de Análisis Léxico Morfológico, y genera el grafo de la consulta mostrada en la Figura 10.

En la Figura 10, lo que está en círculo refleja la nueva información en el grafo de consulta, que consiste en la estructura sintáctica de la consulta y qué tipo de frase es, después del razonamiento de las reglas que están definidas en la ontología lingüística.

Siguiendo con el proceso, se realiza el Análisis Semántico, recibiendo como entrada el resultado

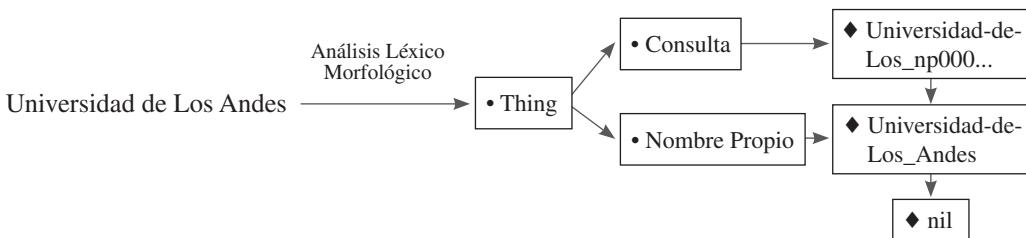


Figura 9. Grafo generado después de realizar el Análisis Léxico Morfológico.

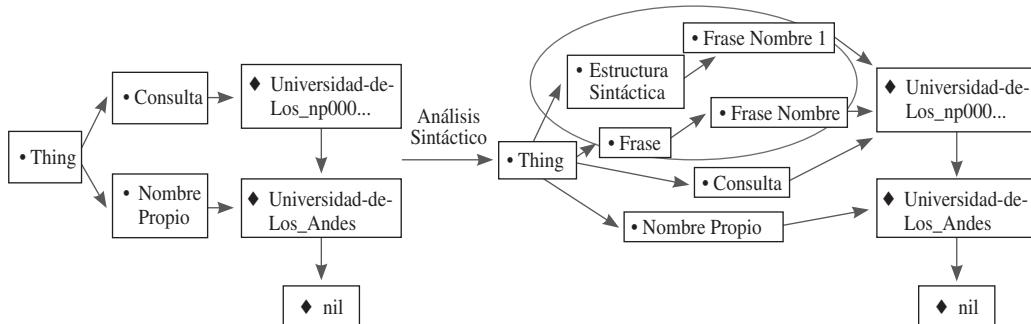


Figura 10. Grafo generado después de realizar el Análisis Sintáctico.

obtenido del Análisis Sintáctico y utilizando la ontología lingüística. Como resultado de ese proceso, se obtiene el grafo de la consulta mostrado en la Figura 11.

En la Figura 11 se puede observar lo que está en el círculo, que representa la nueva información, que consiste en la estructura semántica, la que indica que el término *Universidad de Los Andes* está actuando como agente según su rol semántico, y también define la categoría del término. Todo esto se obtiene después de haber hecho el proceso de razonamiento en la ontología lingüística.

Finalmente se realiza el proceso del Análisis Pragmático, que recibe como entrada el grafo generado del Análisis Semántico, y usa la ontología interpretativa. En el análisis pragmático se contextualiza la información extraída. Por ejemplo, para el grafo de la Figura 11, el análisis pragmático daría como resultado la Figura 12. De la Figura 12 se extrae el conocimiento de la instancia *Universidad de Los Andes*, indicando que es una Entidad de tipo Organización, específicamente Educativa, y además, que es una Universidad. La ontología del perfil de usuario podría ser usada

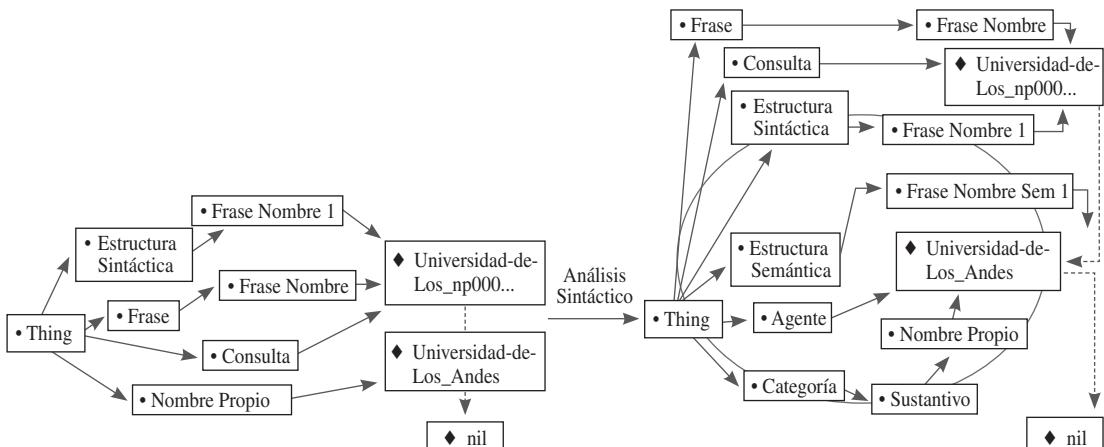


Figura 11. Grafo generado después de realizar el Análisis Semántico.

para determinar de qué país es la *Universidad de Los Andes* a quien se está refiriendo (por ejemplo, usando su ubicación).

En la Figura 12 se muestra el resultado de la interpretación de la consulta, donde nos dice que el término *Universidad de Los Andes* es una Universidad, Universidad es una Organización Educativa, etc. La información nueva esta en los rectángulos.

El resultado de la consulta es llevado a un lenguaje ontológico. Para ello, se pasa por un proceso de transformación, en nuestro caso, a una consulta Booleana, para ser enviada a la Web usando el motor de búsqueda seleccionado para tal fin, y así el usuario final obtiene la información que

desea buscar en la Web. Un ejemplo de la consulta booleana generada es mostrado al final de la siguiente sección.

Los resultados que genera la Web pasan por el componente de aprendizaje, que es un proceso interno del MODS para actualizar sus ontologías, para lograr la adaptabilidad de los componentes del MODS; este proceso se explica en [12].

EXPERIMENTOS

Para la implementación de MODS se utilizaron un conjunto de API, tal como se muestra en la Figura 13.

En general, se usa al editor de ontologías *Protégé*, para invocar a muchas de esas librerías. En particular,

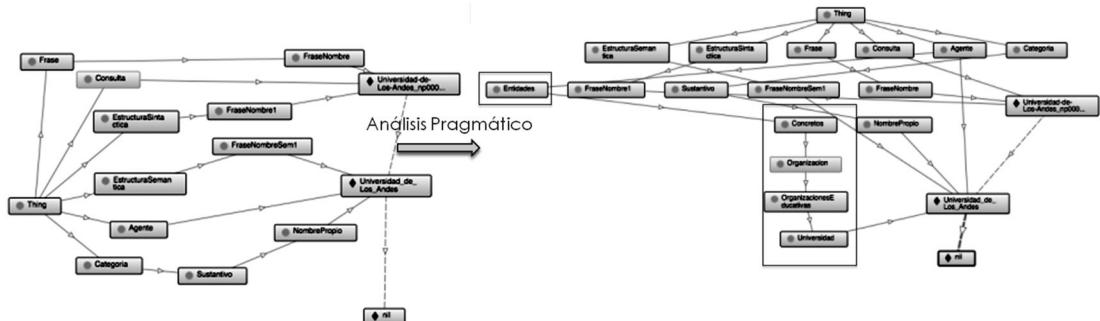


Figura 12. Grafo generado después de realizar el Análisis Pragmático.

El motor de inferencia para razonar

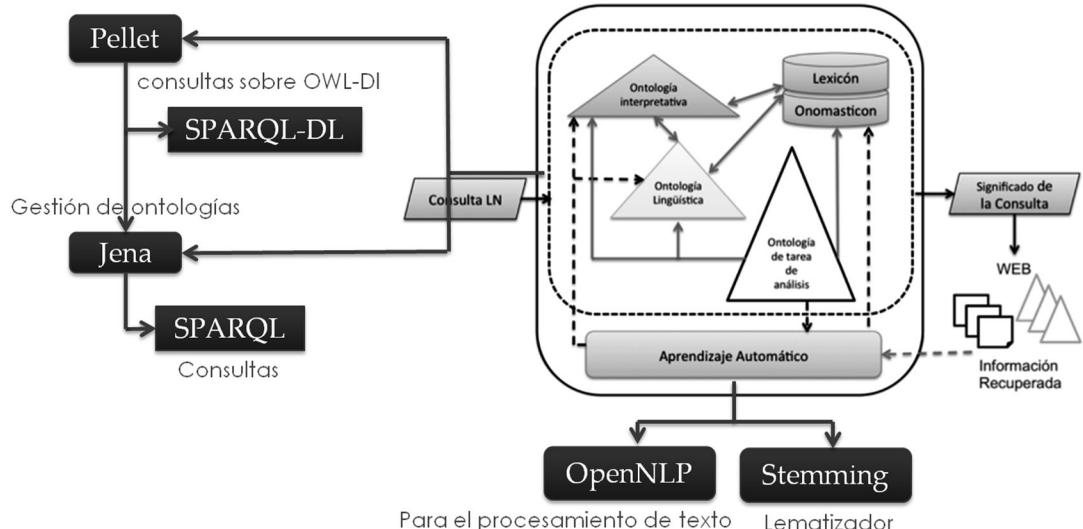


Figura 13. API utilizado en la implantación del MODS.

en la Tabla 6 se muestra que componentes del MODS usan que API:

Tabla 6. API utilizado para la implementación del MODS.

API	Uso en el MODS
Jena	Este API se usa para la manipulación de todas las ontologías del MODS, por ejemplo, para crear, leer o actualizar las ontologías y el grafo de aprendizaje (GA). También se utiliza su motor de consultas, compatible con SPARQL.
OpenNLP	Este API se usa para el procesamiento de textos, en el componente de aprendizaje del MODS.
Pellet	Este API se usa como motor de inferencia, para razonar sobre las ontologías del MODS.
OWL	Este API se usa para la manipulación de las ontologías basadas en OWL.
Stemming	Es el lematizador utilizado por el componente de aprendizaje del MODS.

Ejemplo del procesamiento de la Consulta realizado por el MODS:

En este caso se realiza una consulta al MODS, donde todos los términos son conocidos. Usaremos el mismo ejemplo: “Universidad de Los Andes”. En este caso, el MODS sigue el proceso que se muestra en la Figura 14 (para lo cual usa la ontología de tareas).

El proceso comienza con la detección de las entidades nombradas en la consulta. En el caso del ejemplo se encontró que *Universidad de Los Andes* es una entidad nombrada.

Análisis Léxico-Morfológico

Universidad de Los Andes

Termino	Propiedad	Valor
Universidad_de_Los_Andes	tieneClasificacionSemantica	organizacion
Universidad_de_Los_Andes	tieneEtiquetaEAGLES	npsfo00
Universidad_de_Los_Andes	nombre	Universidad de Los Andes
Universidad_de_Los_Andes	type	Propio
Universidad_de_Los_Andes	type	Palabra_Compuesta
Universidad_de_Los_Andes	type	NamedIndividual
Universidad_de_Los_Andes	type	Thing
Universidad_de_Los_Andes	type	Palabra_Simple
Universidad_de_Los_Andes	type	Resource
Universidad_de_Los_Andes	type	Sustantivo
Universidad_de_Los_Andes	type	Categoría
Universidad_de_Los_Andes	sameAs	Universidad_de_Los_Andes

Figura 15. Tarea de análisis léxico-morfológico.

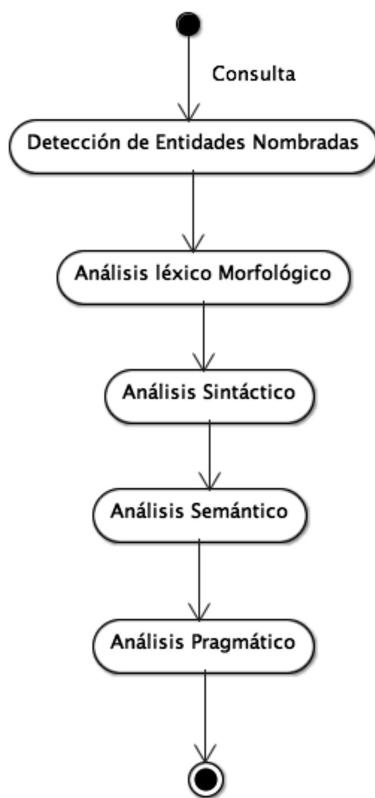
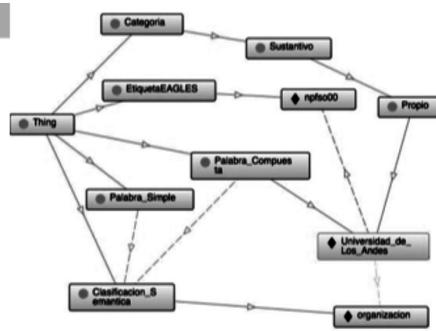


Figura 14. Proceso de la consulta sobre MODS cuando todos los términos son conocidos.

Siguiendo con el proceso, se realiza el análisis léxico-morfológico, en la Figura 15 se muestra el resultado de este proceso. Allí, básicamente se determina que la *Universidad de Los Andes* es un nombre propio, de categoría sustantivo, es una palabra compuesta, y tiene etiqueta EAGLES npsfo00; esta información es obtenida desde el Onomasticon.



El sistema forma la siguiente regla: Universidad_de_Los_andes_npfso00. El proceso sigue con el análisis sintáctico, el que consiste en generar el grafo sintáctico de la consulta. En la Figura 16 se muestra el resultado de este proceso en *Protégé*.

En este caso, se ejecutan las reglas respectivas de la ontología lingüística. Las mismas consisten, para el caso de estudio, en (ver Figura 16):

```
NombrePropio(Universidad_de_Los_andes)
-> FraseNombre
FraseNombre(Universidad_de_Los_andes) ->
FraseNominal
```

Luego, se sigue con el proceso de análisis semántico, el que considera la estructura semántica de la regla sintáctica, tal como se muestra en la Figura 17. Básicamente, para el caso de estudio, se muestra que el nombre propio “Universidad_de_Los_andes” es un sujeto, y su estructura semántica es que el sujeto es agente, aplicando las siguientes reglas:

```
NombrePropio(Universidad_de_Los_andes)
-> Sujeto
Sujeto(Universidad_de_Los_andes) -> Agente
```

Como conclusión de ese proceso, obtenemos que la *Universidad de Los Andes* es un agente.

Luego, continúa el proceso de análisis pragmático, quien se apoya en la ontología interpretativa para

indicar, además, que la *Universidad de Los Andes* es una Universidad que está ubicada en Venezuela, en la ciudad de Mérida, generando la siguiente consulta booleana (es una consulta extendida de la original):

(“Universidad de los Andes” and Mérida and Venezuela) or (ULA Mérida and Venezuela) or (“Universidad de los Andes” and “Núcleo Mérida” and Mérida and Venezuela) or (ULA and “Núcleo Mérida” and Mérida and Venezuela)

Este proceso es el que sigue el MODS cuando la consulta tiene todos los términos conocidos. En el caso que no se conoce un término de la consulta, la ontología de tarea invoca al componente de aprendizaje, el que se explica en [12].

Comparación del MODS con otros trabajos

En la Tabla 7 se presenta una comparación entre el MODS, y los trabajos presentados en la introducción más parecidos con él.

El MODS es el único cuyos recursos lingüísticos se pueden actualizar automáticamente por medio de mecanismos de aprendizaje. Además, genera la consulta en OWL (lo que facilita su uso por los mecanismos de inferencia de la web semántica de más uso actualmente), la que puede ser transformada en otros formatos para ser usadas en diversos contextos: bases de datos, ontologías, etc.

Explanation for: Universidad_de_Los_andes Type FraseNominal	
1)	Universidad_de_Los_andes Pos "0"^^int
2)	Universidad_de_Los_andes tokendeConsulta "idt001"^^string
3)	Universidad_de_Los_andes Type NombrePropio
4)	NombrePropio(?np), Pos(?np, ?p), tokendeConsulta(?np, ?t), equal(?p, 0), equal(?t, "idt001") -> FraseNombre(?np)
5)	FraseNombre(?np) -> FraseNominal(?np)
	In ALL other justifications
	In ALL other justifications
	In ALL other justifications
	In NO other justifications

Figura 16. Resultado de Análisis sintáctico usando la ontología lingüística.

Explanation for: Universidad_de_Los_andes Type Agente	
Universidad_de_Los_andes Pos "0"^^int	?
Universidad_de_Los_andes tokendeConsulta "idt001"^^string	?
Universidad_de_Los_andes Type NombrePropio	?
NombrePropio(?np), Pos(?np, ?p), tokendeConsulta(?np, ?t), equal(?p, 0), equal(?t, "idt001") -> FraseNombre(?np)	?
FraseNombre(?np), NombrePropio(?np) -> Sujeto(?np)	?
Sujeto(?np) -> Agente(?np)	?

Figura 17. Resultado de Análisis semántico usando la ontología lingüística.

Tabla 7. Comparación de los antecedentes con el MODS.

Recursos lingüísticos		Ventajas	Desventajas
MESIA	Aries: Analizador léxico morfológico. EWN: Base de datos semántica que permite obtener sinónimos.	Descompone la consulta en Sintagmas nominales, verbales, preposicionales. Análisis sintáctico para extraer los términos relevantes. Términos relevantes son expandidos por EWN.	Resultado en XML, el que no tiene semántica. No se puede realizar inferencias con los resultados. Los recursos lingüísticos son estáticos.
QUACID	Lexicón de conceptos ontológicos. Patrones de consultas en SPARQL.	Ánalisis Morfológico. Extrae las entidades de cada término del diccionario de conceptos. Asocia la consulta con los patrones ya establecidos.	Resultado en SPARQL, no se puede realizar inferencias con el resultado. Consultas en SPARQL son manuales en relación al historial de la consulta. Los recursos lingüísticos son estáticos.
PowerAgua	Datos en RDF WordNet	Obtiene los datos RDF. Valida la semántica del dato con WordNet.	Los recursos lingüísticos son estáticos. Solo es válido para consultas en inglés.
SWSNL	Ontología propia	Reconocimiento de Nombres Propios. Análisis semántico estadístico. Interpretación semántica.	Los recursos lingüísticos son estáticos.
MODS	Cuenta con un diccionario, onomástico, y ontología interpretativa propias del MODS	Realiza análisis léxico-morfológico, sintáctico, semántico y pragmático. Cuenta con una ontología de tarea para el proceso de interpretación y una ontología lingüística para el análisis sintáctico y semántico. Genera la consulta en lenguaje OWL. La consulta puede ser transformada, en booleana para el caso de recuperación de información, SQL para el caso de base de datos Relacionales, SPARQL y SPARQL-DL, en el caso de Ontologías. Cuenta con un componente de aprendizaje para la adquisición de nuevo conocimiento para la actualización de sus recursos lingüísticos.	

CONCLUSIONES

En este trabajo se presenta la especificación ontología, basada en lógica descriptiva, como la implementación, del Marco Ontológico Dinámico Semántico (MODS) para la Web Semántica.

En específico, el MODS transforma la consulta en un formato de Representación del Significado, el que es un híbrido, basado en expresiones booleanas y en OWL. Para ese proceso de transformación, utiliza los diferentes componentes que lo constituyen: las ontologías del diccionario, onomástico, tareas, lingüística e interpretativa. Además, MODS utiliza mecanismos de la minería semántica ontológica, y herramientas del Procesamiento de Lenguaje Natural, para lograr su objetivo.

Por otro lado, la ontología de tareas invoca al resto de ontologías del MODS, para el procesamiento de la

consulta en lenguaje natural. Dicha ontología utiliza los otros tres niveles de conocimiento (diccionario, lingüístico e interpretativo) del MODS, para generar la consulta del usuario en un lenguaje ontológico, la que será la que se envía a la Web. Así, la ontología de tarea es una especie de metaontología que enlaza al resto de ontologías del MODS. Cada nivel de conocimiento guarda información muy específica requerida para el procesamiento del lenguaje natural, con sus propias dinámicas y autonomía funcional, que se incorporan en el momento adecuado del proceso, según lo vaya requiriendo la ontología de tareas. Como parte de esa autonomía funcional, cada nivel de conocimiento tiene un mecanismo adaptativo, el cual forma parte del componente de aprendizaje del MODS propuesto en otro trabajo [12]. Dicho componente es el encargado de la adquisición automática de conocimiento, tanto léxico, morfológico como semántico (conceptos taxonómicos, no taxonómicos, relaciones, reglas de producción, etc.).

Algunos posibles futuros trabajos son: i) El MODS se puede extender para la interpretación de textos; ii) Cada uno de los componentes del MODS se puede convertir en servicios; iii) Algunas potenciales aplicaciones del MODS son en traductores, y en generación de resúmenes desde un grupo de documentos.

AGRADECIMIENTOS

Al proyecto CDCHT I-1237-10-02-AA, y su proyecto satélite I-1239-10-02-ED, de la Universidad de Los Andes, y al Programa de Cooperación de Postgrados (PCP) entre Venezuela y Francia, titulado “Tareas de Supervisión y Mantenimiento en Entornos Distribuidos Organizacionales”, contrato N° 2010000307. Dr. Aguilar has been partially supported by the Prometeo Project of the Ministry of Higher Education, Science, Technology and Innovation (SENESCYT) of the Republic of Ecuador.

REFERENCIAS

- [1] T. Rodríguez, E. Puerto and J. Aguilar. “Dynamic Semantic Ontological Framework for Web Semantics”. 9th WSEAS Intl. Conference on Computational Intelligence, Man-Machine Systems and Cybernetics (CIMMACS ‘10). Mérida, Venezuela. 2010.
- [2] P. Martínez y A. García “Utilizando recursos lingüísticos para mejora de la recuperación de información en la Web”. Revista Iberoamericana de Inteligencia Artificial. Vol. 6 N° 16, pp. 55-66. 2002. ISSN: 1137-3601.
- [3] J. Goñi, J. González and A. Moreno. “ARIES: A Lexical Platform for Engineering Spanish Processing Tools”. Natural Language Engineering. Vol. 3 N° 4, pp. 317-345. 1997. ISSN: 1469-8110.
- [4] A.J. Lastra-Díaz and A. García. “A new family of information content models with an experimental survey on WordNet”. Knowledge-Based Systems. Vol. 89, pp. 509-526. 2015. ISSN: 0950-7051.
- [5] O. Fernández, R. Izquierdo, S. Fernández y J. Vicedo. “Un sistema de búsqueda de respuestas basado en ontologías, implicación textual y entornos reales”. Procesamiento del lenguaje Natural. Vol. 41, pp. 47-54. 2008. ISSN: 1135-5948.
- [6] V. López, M. Fernández, N. Stieler and E. Motta “Powerqua: supporting users in querying and exploring the semantic web”. Semantic Web, Vol. 3, pp. 249-265. 2012, ISSN: 1570-0844.
- [7] D. Castelo, I. Isi, S. Martínez, G. Gara and Q. Moncecchi. “WebQA: Respuesta Automática a Preguntas”. XXXIV Conferencia Latinoamericana de Informática. 2008.
- [8] I. Habernal and M. Konopík. “SWSNL: Semantic Web Search Using Natural Language”. Expert Systems with Applications. Vol. 40, pp. 3649-3664, 2013, ISSN: 0957-4174.
- [9] A. Gelbukh y H. Calvo. “Determinación automática de roles semánticos usando preferencias de selección sobre corpus muy grandes”, Computación y Sistemas. Vol. 12 N° 1, pp. 128-150. 2008. ISSN 1405-5546.
- [10] S. Nirenburg and V. Raskin. “Ontological Semantics”. MIT Press Books, Boston, USA. ISBN: 9780262140867. 2004.
- [11] J. Aguilar. “Sistema semántico para la búsqueda inteligente de información por contexto para la Web”. Revista Ciencia e Ingeniería. Vol. 32 N° 3, pp. 141-152, 2011. ISSN 1316-7081.
- [12] T. Rodríguez and J. Aguilar. “Ontological Learning for a dynamic semantics ontological framework”. DYNA. Vol. 81, pp. 56-63 2014. ISSN: 0012-7353
- [13] H. Ishkewy and H. Harb. “ISWSE: Islamic Semantic Web Search Engine”. International Journal of Computer Applications. Vol. 112 N° 5, pp. 37-43, 2015. ISSN: 0975-8887.
- [14] K. Palaniappan and S. Vijayalakshmi. “Semantic Web Based Efficient Search Using Ontology and Mathematical Model”. International Journal of Engineering and Technology. Vol. 5 N° 6, pp. 4914-4928. 2014. ISSN: 0975-4024
- [15] A. Marginean. “GFMed: Question Answering over BioMedical Linked Data with Grammatical Framework”. 2014 Conference and Labs of the Evaluation Forum, Sheffield, UK. 2014.
- [16] D. Song, F. Schilder, C. Smiley, C. Brew, T. Zielund, H. Bretz, R. Martin, C. Dale, J. Duprey, T. Miller and J. Harrison. “TR

Discover: A Natural Language Interface for Querying and Analyzing Interlinked Datasets”. Lecture Notes in Computer Science. Vol. 9367, pp. 21-37, 2015. ISBN: 978-3-319-25010-6.

[17] R. Usbeck, A. Ngonga, L. Bühmann and C. Unger. “HAWK - Hybrid Question Answering Using Linked Data”. Vol. 9088. Lecture Notes in Computer Science, pp. 353-368, 2015. ISBN: 978-3-319-18818-8.