Procesamiento del Lenguaje Natural, Revista nº **38**, Abril de 2007

sociedad española para el Procesamiento del Lenguaje Natural

ISSN: 1135-5948

ISSN: 1135-5948
Depósito Legal: B:3941-91
Distribuye: Sociedad Española para el Procesamiento del Lenguaje Natural
Editado por la Universidad de Jaén

Procesamiento del Lenguaje Natural, Revista nº **38**, Abril de 2007

sociedad española para el Procesamiento del Lenguaje Natural

ISSN: 1135-5948

Artículos:

Tesis:

Representación formal de la estructura lógica de sitios web, y su aplicación a un navegador web multilingüe basado en diálogo

Fernando Martínez Santiago, Arturo Montejo Ráez
y Miguel Ángel García Cumbreras
Dpto. de Informática, Universidad de Jaén
Campus de las Lagunillas s/n, 23071 - Jaén
dofer@ujaen.es, amontejo@ujaen.es, magc@ujaen.es

Resumen: Un problema bien conocido de HTML es el pobre contenido semántico de sus etiquetas, dejando la tarea de interpretar los distintos elementos y secciones que conforman el sitio web al usuario. Frente a ello, iniciativas como la web semántica proponen percibir la web como una red de ontologías de manera que el significado de un sitio web sea computacionalmente accesible. Entre ambos extremos, en este trabajo se propone un formalismo denominado *Web Logic Forms* (WLF) que permite representar de manera formal cómo la información esta estructurada en un sitio web, pero sin entrar en la representación del contenido textual del sitio. De esta manera es posible que el sitio web sea presentado de una manera conveniente al usuario en otros caminos distintos al meramente visual. Es por ello que la aportación aquí propuesta no consiste en permitir realizar nuevas y complejas tareas sobre la web tal como persigue la web semántica, sino dotar de la formalidad suficiente a una página expresada en HTML para que permita al navegador u otro software conocer cómo se distribuye y estructura la información allí codificada. En esta línea en se propone un navegador web basado en diálogo apropiado para personas invidentes o para su uso en dispositivos portátiles.
Palabras clave: lógica de predicados primer orden (LPO), Web Logic Forms (WLF), Web Logic Forms Rules (WLFR), HTML, gestor de diálogo, navegador web

Abstract: HTML tags have poor semantic meaning because the final user of the web is supposed to be a human being with several skills. The user has understand the web site by means of natural language, visual features of text and images, etc. Semantic web deals to create a net of ontogies into the web by describing the meaning of the site in a more formal way. In this work, we propose a formal representation named *Web Logic Forms* (WLF) between HTML and semantic web in order to represent the logic structure of a web site. Thus, the navigator is able to present the information of the site in a more appropriate way for a given user. By example, the navigator was able to present the information without any visual object, by "reading" the information by using structural aspects of the site such as headings, sections, news, etc. In order to test WLF, we propose a web navigator based on dialog suitable for blind persons or navigation by using small portable devices such as PDAs or smart phones.
Keywords: first order logic, Web Logic Forms (WLF), Web Logic Forms Rules (WLFR), HTML, dialog manager, web navigator

1. Introducción

Que la Web ha supuesto una revolución en el modo de publicar y acceder a la información es algo ya asumido desde hace tiempo. Sin embargo, este trasiego de información dista de ser universal debido a limitaciones impuestas en los diversos elementos necesarios para que la comunicación entre el usuario y sitio web sea posible. En todo acto de comunicación se requiere un emisor, un receptor, un canal, un mensaje, un código y un contexto. En el caso de la web existen diversas restricciones sobre cada uno de estos elementos que limitan su acceso. En concreto, el código imperante en la web (HTML+lenguaje natural+gráficos..) dista de ser universal, pues en la mayoría de los casos asume un perfil determinado de receptor:

- Para poder navegar, el receptor debe es-

ISSN: 1135-5948

tar capacitado para percibir la estructura del sitio atendiendo a aspectos visuales tales como tamaño del texto, ubicación del texto dentro de la página, etc.

- Para poder comprender el mensaje, el receptor debe ser capaz de leer e interpretar el código utilizado en la redacción del mensaje, primordialmente lenguaje natural e imágenes.

Si de lo que se trata es de ampliar el tipo de receptores capacitados para interpretar el contenido del mensaje, por ejemplo cuando el receptor es un programa de ordenador, entonces es la comunidad dedicada al estudio y desarrollo de la web semántica la que se ocupa de ello[1](Berners-Lee, Hendler, y Lassila, 2001). Si, por el contrario, la limitación no la impone la capacidad cognitiva del receptor, si no la imposibilidad, por un motivo u otro, de percibir la información codificada en el sitio web, entoces es un aspecto investigado dentro del área de accesibilidad web o WAI (web accessibility iniciative) [2]. Esta iniciativa anima al diseño de sitios web que sean más fácilmente accesibles por personas con algún tipo de discapacidad. Sin embargo, son pocos los sitios que tienen en cuenta las recomendaciones más básicas en cuanto a accesibilidad.

En este trabajo se propone un enfoque original para superar la limitación que supone el "aspecto" de la mayoría de los sitios web. Este enfoque requiere añadir el grado de formalismo necesario para que un navegador pueda "conocer" cómo está la información estructurada, aunque finalmente no sepa de qué se habla allí. Para ello, se propone un formalismo denominado *Web Logic Forms* (WLF) derivado directamente a partir de HTML, y unas reglas que operan sobre WLF, denominadas WLFR (*WLF Rules*). WLF+WLFR permite dotar al navegador de la información suficiente referente a la estructura del sitio web como para mostrar tal información de la manera que resulte más adecuada al perfil de usuario. La conversión de un sitio ya existente al formalismo aquí propuesto si bien no es automática, es sencilla pues se reduce a acompañar el sitio web con un conjunto de reglas WLFR que permita al navegador interpretar correctamente las etiquetas HTML de una manera similar a como las interpretaría una persona cuando percibe la expresión visual de tales etiquetas. Por ejemplo, una de tales reglas podría indicar que textos escritos en negrita y de un determinado tamaño son titulares, o que los enlaces que se encuentran precedidos de una determinada etiqueta son secciones.

El resto del presente artículo está estructurado como sigue: En la sección 2 se repasa brevemente diversas tecnologías relacionadas con el problema abordado. En la sección 3 se presenta con detalle el formalismo propuesto para la descripción formal de la estructura lógica de un sitio web, WLF. A continuación se describe brevemente un navegador web basado en diálogo que hace uso de WLF+WLFR. Y finalmente, se discuten algunos aspectos relevantes y líneas de trabajo futuras que quedan abiertas a partir de la presente investigación.

2. Trabajo relacionado

En la figura 1 se muestran diversos códigos o lenguajes ordenados según su capacidad expresiva y el coste computacional para la manipulación automática del mensaje escrito mediante tal código. En un extremo queda la descripción de un sitio web expresado exclusivamente en lenguaje natural. Un sitio web cuyo contenido y estructura lógica sea descrita exclusivamente usando lenguaje natural tiene una gran capacidad expresiva pero es inviable computacionalmente. Próximo a este extremo se encuentra el conjunto formado por HTML, junto con todo aquello que no se corresponde con un elemento de marcado: lenguaje natural, gráficos, sonidos... Por ello, que un algorítmo pueda "compreder" la informción codificada en un sitio web es casi tan difícil como si de texto plano se tratara.

El otro extremo de la cadena quedan aquellos sitios web cuyo significado está exclusivamente codificado en algún lenguaje formal, si es que ello fuera posible. Esto aseguraría que la semántica de ese sitio web es manejable en términos computacionales, pero a costa de severas limitaciones expresivas (Levesque y Brachman,). Un equilibrio deseable entre ambos extremos lo representa RDF y OWL[3]. OWL es el acrónimo del inglés *Web Ontology Language*, un lenguaje de marcado para publicar y compartir datos usando ontologías

[1]Web semántica: http://www.w3.org/2001/sw

[2]Iniciativa para la accesibilidad de la web: http://www.w3.org/WAI

[3]RDF: http://www.w3c.org/rdf, OWL:http://www.w3c.org/owl

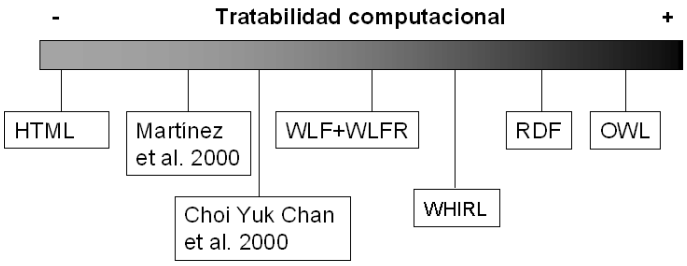

Figura 1: Idoneidad de algunos lenguajes (complementados con lenguaje natural, imágenes...) utilizados para codificar el significado de un sitio web.

en la Web. OWL, en su versión más popular, OWL-DL, es una implementación de lógica descriptiva (Baader et al., 2003) que toma la sintaxis de un modelo de marcado construido sobre RDF y codificado en XML. Así, un uso adecuado de OWL permite el razonamiento automático sobre el contenido de un sitio web, y requiere el diseño de una ontología que describa los objetos relevantes presentes en tal sitio Web y relaciones entre éstos. El problema es que esa ontología tiene que ser codificada por un experto, y esta tarea dista mucho de ser trivial.

Existen diversas propuestas para representar principalmente la estructúra lógica del sitio web, y es en este ámbito donde mejor encuadra el presente trabajo. Así, en (Martínez-Santiago, Ureña, y García-Vega, 2001) se propone una herramienta que permite derivar a partir de ciertos sitios web un documento XML con etiquetas tales como "autor", "título", "sección". Este modelo requiere describir mediante un conjunto de sencillas reglas cómo extraer tales etiquetas del sitio web. Un aspecto destacable es la naturaleza jerárquica de tales reglas de tal modo que es posible crear reglas que se aplican a cualquier página web, sólo a las páginas pertenecientes a un sitio web, a una sección de ese sitio web, o incluso a una página determinada. Por contra, es una herramienta que queda limitada por el conjunto de reglas disponible, así como por su orientación a explorar sitios de contenido periodístico por lo que le presupone una estructuración al sitio bastante rígida. En esta línea, el trabajo propuesto en (Chan y Li, 2000) que aporta algo más de flexibilidad gracias al uso de XSL como formalismo principal para representar aquellas reglas que permiten extraer información semántica del sitio web. Nótese que en ambos casos es necesario elaborar un conjunto de reglas mediante la exploración del sitio web que se desea tratar. Es en estas reglas justamente donde se codifica la semántica adicional con que se quiere enriquecer tal sitio, al modo en que, por ejemplo, OWL puede describir el contenido de un sitio web, sólo que aquí la dificultad es menor, pues no se trata de describir el contenido sino la estructura lógica del sitio.

Por otra parte, existe una amplia variedad de formalismos de representación que dan soporte a diversos *wrappers* web tal como WHIRL(Cohen, 2000). Este lenguaje, que también tiene inspiración l*o*gica, tiene una capacidad expresiva adecuada para obtener una representación formal de todo el contenido del sitio web, por lo que excede el objetivo de WLF, y su complejidad. El esfuerzo que requiere escribir y mantener un *wrapper* para uno de estos lenguajes es similar al que se requería en una web semántica. Por ello, existen esfuerzos por conseguir automatizar la escritura de *wrappers* mediante técnicas de aprendizaje automático y minería de datos, pero es difícil, con el estado de la tecnología actual, que una máquina "aprenda" el contenido semántico de una web por sí sola. Sin embargo, dado que WLF se centra en la estructura del sitio, y no en su contenido, el esfuerzo de escribir y mantener uno de tales *wrappers* es mucho menor. Por todo ello, el motivo último de WLF es describir un lenguaje computacionalmente asumible, de fácil escritura, y que aún así tenga un grado de formalismo útil para diversas tareas, como ADN, el navegador multilingüe basado en diálogo descrito en la segunda parte de este artículo.

Cuadro 1: Algunas frases y la forma lógica obtenida

Juan vuela desde Tokio hasta Nueva York	Juan_[P](x1) volar_[V](e1 x1) desde_[P](e1 x2) Tokio_[N](x2) hasta_[P](x2 x3) Nueva_York_[N](x3)
John es golpeado por una pelota	John_[N](x1) golpear_[V](e1 x2 x1) por_[P](e1 x2) pelota_[N](x2)
En vez de alubias comeré pizza	En_vez_de_[P](x2 x1) alubias_[N](x1) comer_[V](e1 _ x2) pizza_[N](x2)
El baloncesto y el tenis son grandes deportes	baloncesto_[N](x1) y_[C](x3 x1 x2) tenis_[N](x2) ser_[V](e1 x3 x4) grande_[A] (x4) deporte_[N](x4)
El profesor permitió un periodo de descanso	profesor_[N](x1) permitir_[V](e1 x1 x3) periodo_[N](x3) de_[P](x3 x2) descanso_[N](x2)

3. *WLF+WLFR: Descripción formal de la estructura lógica de un sitio web*

Web Logic Forms toma su nombre del formalismo para la representación semántica del lenguaje natural conocido como identificación de formas lógicas (Rus, 2002). La identificación de formas lógicas es un formalismo basado en lógica de predicados de primer orden (LPO) que pretende obtener una representación del lenguaje natural situada entre el nivel sintáctico y semántico partir de un texto expresado en lenguaje natural. La base de tal formalismo es la lógica de predicados de primer orden, de tal manera que a cada palabra presente en el texto se le asigna un predicado. A su vez cada predicado puede tener varios argumentos que representan la relación de ese predicado con otros elementos de la frase.

La identificación de la forma lógicas es una tarea compleja que requiere un análisis sintáctico del texto y, usualmente, un conjunto de reglas que permita interpretar el árbol sintáctico. Realmente, en el caso de WLF la tarea es más sencilla al tratarse de un lenguaje formal como es HTML, que además tiene un sintaxis sencilla y muy homogénea consituída básicamente por una secuencia de etiquetas que ocasionalmente incluyen algunos atributos y o algún texto que acompaña a la etiqueta y sobre el cual opera. En la tabla 1 se muestran algunos ejemplos de frases junto a su forma lógica equivalente.

De manera análoga se identifica la forma lógica de una página HTML. Los elementos HTML se corresponden con un predicado, cuyo primer argumento es una constante exclusiva de ese predicado, y que representará a ese elemento HTML allí donde haga falta. Más detalladamente, los pasos para obtener la forma lógica de una página HTML son los siguientes:

- Cada etiqueta HTML se representa mediante un predicado. Cada ocurrencia de esa etiqueta se identifica mediante una constante que es el primer argumento del predicado equivalente. A modo de ejemplo, de la etiqueta $< html >$ obtenemos la forma lógica *html(h1)*. El significado de cada argumento que recibe el predicado depende de la posición que éste ocupa:
  1. Constante que representa a una instancia determinada de una etiqueta HTML.
  2. Etiqueta HTML de la que depende ("none", si no depende de ninguna). De esta manera se representa la naturaleza jerárquica de HTML.
  3. Indica si se marca el inicio (*open*) o fin (*close*) de una sección.
  4. Número de etiqueta. Un número único que se corresponde con el lugar de aparación de la etiqueta dentro de la página. Realmente, este argumento es una forma alternativa de referirse a la etiqueta que representa, cuando resulta de utilidad tener en cuenta el orden relativo entre etiquetas.
- Cada atributo HTML se representa mediante, al menos, dos predicados, uno representa el atributo, y otro el valor que toma. El atributo queda identificado por la constante que representa a la etiqueta de la cual depende ese atributo, junto con el nombre del atributo.
- El texto entre etiquetas se representa con el predicado "text", cuyo identificador se

corresponde con el identificador de la etiqueta HTML que le contiene.

En la tabla 2 se muestran algunos ejemplos de código HTML con su correspondiente forma lógica.

3.1. Extracción de la estructura lógica de un sitio web

Ya que la WLF es una reescritura de HTML utilizando lógica de predicados de primer orden, el grado de formalismo de la página original y la derivada es el mismo, pero con la ventaja de que ahora contamos con las herramientas propias de la lógica para manipular ese código y obtener así una base de conocimiento con información relativa a la estructura lógica del sitio web. Así pues, para extraer información sobre aspectos estructurales del sitio web es necesario escribir reglas que identifiquen los elementos relevantes de la página: título, secciones, titulares, enlaces, etc. Nótese que, a diferencia del lenguaje natural donde la semántica de una frase queda determinada en buena medida por la sintaxis de ésta (Levin, 1993), HTML informa escasamente sobre el significado del mensaje codificado mediante su uso (esto es, cómo se organiza la información almacenada). Ésta es una diferencia primordial entre la identificación de formas lógicas y WLF. Mientras que en la mayoría de los casos, para identificar la forma lógica de una frase es suficiente con un conjunto finito de reglas (salvo en caso de ambigüedad sintáctica), en el caso del HTML esas reglas son completamente dependientes de cada página web que se desea manipular, debido a que HTML no está concebido para informar sobre la estructura lógica del sitio web que describe. Nótese que para que aplicaciones terceras puedan aprovechar convenientemente la información extraída es conveniente que tales reglas sigan alguna ontología sencilla que enumere y describa los objetos estructurales de la página y como cómo se relacionan. En la figura 2 se esquematiza el proceso. Nótese que la obtención de la forma lógica es independiente del sitio, pero no así la base de conocimiento, que es el resultado de aplicar a WLF las reglas escritas a tal efecto mediante algún demostrador de teoremas automático. En el anexo 1 se muestra el resultado final obtenido a partir de código HTML real extraído de un diario digital. Algunos hechos que típicamente se pueden preguntar a la base de conocimiento resultante son las secciones que se encuentran en la página, el título o los productos que se ofertan, si se tratara de un sitio dedicado al comercio electrónico.

4. ADN: Un navegador web gestor de diálogo basado en WLF

Presentar la información de un sitio web mediante el uso exclusivo de voz o texto, sin apoyo de formato alguno, dista de ser una tarea trivial. Existen algunos productos comerciales como JAWS (acrónimo de *Job Access With Speech*)[4], que permiten al usuario interactuar con un navegador basado en texto y leer secuencialmente la página web. Claramente, esta forma de navegar resulta pesada cuando se trata de leer o acceder a alguna sección de un sitio web comercial, que usualmente presenta una gran cantidad de información al usuario, el cual percibe visualmente los diversos componentes de tal página web, centrándose así rápidamente en aquellos aspectos de su interés (buscar una sección, leer los titulares, la descripción de un producto, etc). Ya que WLF+WLFR permite representar formalmente la estructura lógica de un sitio web, es posible que un navegador aproveche esa información para presentar la página web de una manera ordenada. Por ejemplo, sería posible que en un diario dado, diera al usuario la opción de leer los titulares o enumerar las secciones disponibles. Leer, si así lo desea el usuario, la entradilla de algún titular y posteriormente el contenido completo de la noticia, etc. En esta sección a se presenta ADN (del inglés, *A Dialog-based Navigator*). ADN permite gracias al uso de WLF+WLFR navegar de una manera eficiente sin utilizar para ello código visual alguno, tan sólo un uso controlado del lenguaje natural. Además, el hecho de que el navegador conozca la estructura del sitio web permite que la navegación mediante texto pueda realizarse en el idioma del usuario, siempre que se haya realizado previamente la localización necesaria.

A continuación se describen los dos módulos principales de que consta ADN: el gestor de contenidos web y el gestor de diálogo.

[4] JAWS: http://www.freedomscientific.com

Cuadro 2: Una porción de código HTML junto con su forma lógica equivalente

HTML	WLF
<html>	html(h1, none, open,1)
<title> Diario Digital </title>	title(h2, h1, open, 2) text(h2,"Diario Digital") title(h2, h1, closed, 3)
<body>	body(h3, h2, open, 4)
<a href="/opinion/col1.html">	a(h4,h3,open,5) attr("a",h4) fullValue("a",h4,"/opinion/col1.html")

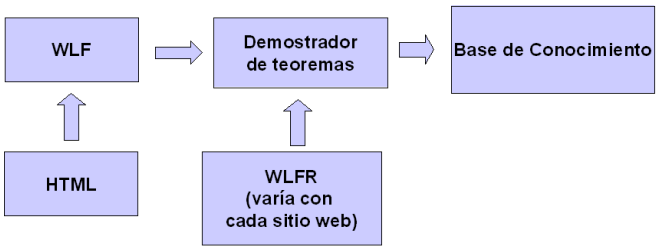

Figura 2: Esquema del proceso de extracción de la estructura lógica de un sitio web

4.1. El gestor de contenido web

Este módulo tiene capacidad para descargar páginas web, derivar la forma lógica de tal página y razonar sobre ella usando las reglas escritas a tal efecto. En cierta forma, se comporta como una base de conocimiento extraída de la web mediante el uso combinado de WLF+WLFR junto con algún demostrador de teoremas automático. En este trabajo el demostrador de teoremas usado es OTTER[5], que es un completo sistema de deducción automático basado LPO con capacidad para manejar la igualdad mediante demodulación y paramodulación y estrategias de búsqueda tales como hiperresolución o resolución binaria. Es el demostrador de teoremas basado en lógica de primer orden más extendido en la actualidad. Se ha elegido por estar bien documentado, ser sobradamente potente, y lo bastante rápido como para usarlo en tiempo real (al menos para las demostraciones aquí requeridas).

Finalmente el gestor de contenidos web también cuenta con una pequeña base de datos que permite anotar información referente al perfil de usuario, cookies, o cualquier dato que por un motivo u otro deba almacenarse.

[5]OTTER: http://wwwunix.mcs.anl.gov/AR/otter

4.2. El gestor de diálogo

El gestor de diálogo, que interactúa con la base de conocimiento en función de las ordenes que reciba del usuario. Más concretamente, el gestor de diálogo sigue un modelo basado en redes de transición aumentadas o ATNs (*Augmented Transition Networks*) (Woods, 1970),(Woods, 1973). Existen varios ejemplos en la literatura (Levy et al., 1997), (McTear, 1998), (Robinson et al., 2004), en los que el gestor de diálogo se basa en un autómata. Este es un paradigma que permite modelizar de una manera muy intuitiva aquellos diálogos de carácter imperativo, con un escenario controlado y un número relativamente pequeño de alternativas en cada momento. Si esto no se cumple, cualquier tipo de autómata necesario para modelizar un acto conversacional se vuelve excesivamente complejo, pesado, y finalmente poco manejable. En nuestro caso, la lógica de una ATN se adapta muy naturalmente al modo que usualmente navegamos, tal como se describe a continuación.

Una ATN es una red recursiva cuyas transiciones cuentan con unos registros que pueden ser leídos (operación *test*) o escritos (operación *action*) antes o después de pasar al siguiente estado. A su vez, una red recursiva es, en esencia, una automáta finito determinista donde se permite que una transición

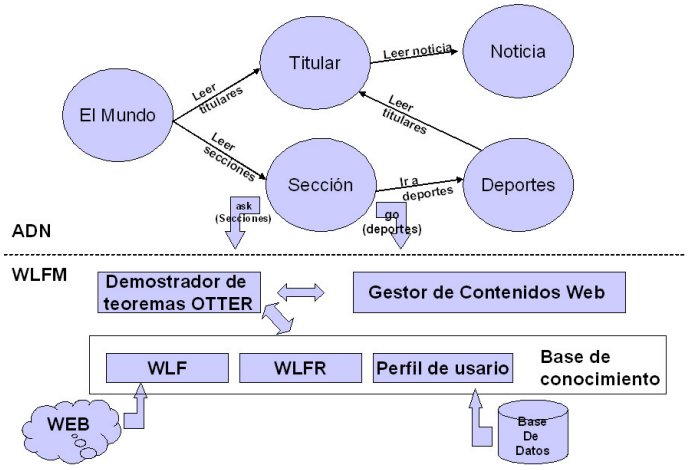

Figura 3: Arquitectura del navegador multilingüe ADN.

represente, además de un evento como es usual, un nuevo autómata. La transición se completa en el momento que el nuevo autómata llega a un estado final. De esta manera, se crea una pila de autómatas conforme se va transitando a través del ATN, de manera similar a como se apilan los sitios web en el historial de un navegador web conforme se visitan nuevos sitios. Así, cada autómata se corresponde con un sitio web, y cada estado representa una página determinada. Finalmente, una transición queda descrita por los siguientes elementos:

- Operación *test*. Precondiciones que deben cumplirse para que tal transición se lleve a cabo. Por ejemplo, que el usuario esté registrado. Este tipo de información generalmente se obtiene a partir de la base de datos contenida en el gestor de navegación.
- Eventos. Órdenes de usuario que dirigen la navegación a través del sitio. Estas ordenes se corresponden más o menos con la interacción usual con un navegador web. Se permite cierto grado de flexibilidad mediante el uso de expresiones regulares. Este enfoque, si bien es sencillo, resulta adecuado pues se trata de un diálogo sobre un dominio bien delimitado y dirigido por el navegador web. Generalemente, se tratará de respuestas a preguntas concretras de ADN, como "¿Quiere visitar la sección de nacional, internacional o deportes?".
- Operación *action*. Postcondiciones como resultado de la ejecución de la transición. Por ejemplo, si se abandona un estado que representa una página de identificación, podría almacenarse ahora el nombre de usuario y la clave facilitada.

En la figura 3 se muestra la arquitectura propuesta tomando como ejemplo un diario digital.

La mayor limitación de ADN es que sólo puede navegar sobre aquellas páginas web para las cuales se han creado las reglas WLFR pertinentes y, además, se ha diseñado el automáta que describa el modo de navegar en ese sitio web, si bien es posible crear autómatas estándar para grupos de sitios web que compartan una estructura de navegación similar. Por ejemplo, es posible crear un autómata para diarios, otro para comercio electrónico, otro para blogs, etc. Las peculiaridades de cada uno de estos sitios son ocultadas por el gestor de navegación a través de la base de conocimiento.

Actualmente, existe un primer prototipo de ADN que opera sobre dos diarios en español (El Mundo y el diario deportivo SPORT), uno inglés (The Guardian) y uno francés (Le Monde), que confirma la viabilidad de la arquitectura, si bien aun falta por probar el enfoque en sitios que requieren un grado más alto de interactividad, como un sitio de comercio electrónico, por ejemplo.

5. *Consideraciones sobre la relación entre WLF y OWL*

Ya que OWL permite describir el contenido de un sitio web, ¿por qué no usar directamente OWL?. Realmente, el hecho de describir la estructura de la página web usando lógica de predicados de primer orden o OWL

es irrelevante. OWL en su versión más equilibrada, OWL-DL, es una implementación de la lógica descriptiva que es a su vez un subconjunto de la lógica de predicados de primer orden. Así que el paso de LPO a OWL es más o menos trivial, si bien es cierto que OWL es, en cierta forma, una especialización de LPO, muy orientado a escribir ontologías y razonar sobre ellas. Es adecuado pues para representar *qué* contiene un sitio web, y no tanto *cómo* está organizado tal sitio, para lo cual en este trabajo se ha preferido la capacidad expresiva de la lógica tradicional. En cualquier caso WLF no es una alternativa a OWL, sino un procedimiento para derivar la forma lógica de un sitio web. El lenguaje usado para codificar la forma lógica para posteriormente razonar sobre ella es secundario. En definitiva, no supone más que una pequeña variación sobre el mismo tema implementar WLF sobre OWL.

6. Conclusiones y trabajo futuro

Se ha presentado un procedimiento denominado WLF que permite representar HTML mediante LPO. Ello permite, en primer lugar, dotar a la página de información referente a la estructura lógica del sitio web. Esta información realmente ya está latente en la página. El problema es que el lenguaje usado para ello (HTML+texto+imagenes+...), es extremadamente vago e impreciso, y por lo tanto intratable computacionalmente. Lo que el uso combinado WLF y WLFR posibilita es justamente tratar de manera automática la estructura lógica del sitio web, de forma similar a cómo la web semántica permite explotar el contenido de tal sitio, más allá de cómo tal contenido se muestre de cara al usuario.

Un ejemplo práctico de uso de WLF+WLFR es el navegador web conversacional multilingüe ADN, que interactúa con el usuario usando exclusivamente lenguaje natural, presentando al usuario la información de manera ordenada y conveniente. Además, dado que ADN conoce la estructura lógica del sitio es posible interactuar con el usuario en el idioma de éste, con independencia del idioma utilizado en el sitio web, si bien es cierto que la información finalmente solicitada se mostrará en el idioma original, salvo que se traduzca. ADN es un relativamente sencillo gestor de diálogo basado en ATNs. Cada ATN representa un sitio web. Los estados son páginas o tipos de páginas concretos de tal sitio, las acciones son órdenes de usuario que permiten saltar de una página a otra, o a otro sitio web.

Ya que la web no es sólo HTML, una línea de trabajo futura es ampliar la defínción de WLF a otros lenguajes comunes en la codificación de sitios web, tales como XML+XSL o lenguajes script.

Por otra parte, la mayor limitación que tiene WLF es que requiere de la escritura de las reglas WLFR mediante el análisis manual del texto HTML. Esperar que esta tarea se pudiera realizar de manera totalmente automática es casi tanto como querer que, al día de hoy, una máquina pueda comprender lenguaje natural en toda su riqueza. En cualquier caso, si es posible realizar herramientas de apoyo, que mediante algorítmos de aprendizaje automático supervisado, puedan ayudar al experto humano en el desarrollo de las reglas WLFR.

En lo relativo al navegador ADN, actualmente se está evaluando con buenos resultados la eficacia del navegador en sitios que requieran un alto grado de interacción, tal como la realización de un pedido. Un segundo aspecto más ambicioso es dotar al gestor de diálogo de cierta capacidad de comprensión de lenguaje natural y planificación. Una comprensión más elaborada del lenguaje natural permitiría diseñar planes para dar respuesta a solicitudes de usuario que conlleven realizar más de una acción a partir de una única sentencia, tales como “navega a la sección de deportes de El País”,que requiere navegar al diario y luego a la sección solicitada, o “lista los productos de la sección de electrodomésticos de El Corte Inglés”.

7. Agradecimientos

Este trabajo ha sido financiado parcialmente mediante el proyecto TIMOM (TIN2006-15265-C06-03), del Ministerio de Ciencia y Tecnología, y el proyecto de investigación de la Universidad de Jaén con código RFC/PP2006/Id_514.

Bibliografía

Baader, Franz, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, y Peter F. Patel-Schneider, editores. 2003. *The Description Logic Handbook: Theory, Implementation, and Applications.* Cambridge University Press.

Berners-Lee, Tim, James Hendler, y Ora Lassila. 2001. The Semantic Web. A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities. *The Scientific American*, Mayo.

Chan, Jessica Choi Yuk y Qing Li. 2000. WebReader: A Mechanism for Automating the Search and Collecting Information from the World Wide Web. En *WISE*, volumen 2, páginas 20–47.

Cohen, William W. 2000. WHIRL: A word-based information representation language. *Artif. Intell.*, 118(1-2):163–196.

Levesque, Hector J. y Ronald J. Brachman. *Readings in Knowledge Representation.*

Levin, Beth. 1993. *English Verb Classes and Alternations: A Preliminary Investigation.* University of Chicago Press, Chicago, IL.

Levy, D., R. Catizone, B. Battacharia, A. Krotov, y Y. Wilks. 1997. Converse: A conversational companion. En *Proceedings of the First International Workshop on Human-Computer Conversation*, páginas 27–34, Bellagio, Italia.

Martínez-Santiago, Fernando, Alfonso Ureña, y Manuel García-Vega. 2001. WWW como fuente de recursos lingüísticos. *Sociedad Española para el Procesamiento del Lenguaje Natural*, 27:41–70.

McTear, M. 1998. Modelling spoken dialogues with state transition diagrams: experiences with the cslu. En *ICSLP.*

Robinson, Kerry, D. Horowitz, E. Bobadilla, M. Lascelles, y A. Suarez. 2004. Modelling spoken dialogues with state transition diagrams: experiences with the cslu. En *SIGDIAL.*

Rus, Vasile. 2002. *Logic Form For WordNet Glosses and Application to Question Answering.* Ph.D. tesis, Computer Science Department, School of Engineering, Southern Methodist University, Dallas, Texas.

Woods, W. A., 1973. *Natural Language Processing. Algorithmics Press*, capítulo An experimental parsing system for transition network grammars, páginas 111–154. Rustin, R., New York.

Woods, W.A. 1970. Transition network grammars for natural language analysis. En *CACM*, volumen 13, páginas 591–606.

Anexo I. Ejemplo de WLF+WLFR sobre código HTML

HTML

Villepin autoriza el toque de queda en los lugares azotados por la violencia callejera

WLF

div(h1,none,open,1)
attr("class",h1)
fullValue("class",h1),"tituno")
a(h2,h1,open,2)
attr("href",h2)
fullValue("href",h2,"/elmundo/2005/11/07/sociedad/1131392990.html")
attr("class",h2)
fullValue("class",h2,"tituno")
text(h2,"Villepin autoriza el toque de queda en los lugares azotados por la violencia callejera")
a(h2,h1,close,3)
div(h1,none,close,4)

WLFR

Obtener el titular:
all x1 x2 x3 div(x1,none,open,x3) & fullValue("class",x1)="tituno" & a(x2,x1,open) & text(x2)→diario.titular.texto(x2)

Obtener la URL de la noticia:
all x1 x2 x3 div(x1,none,open,x3) & fullValue("class",x1)="tituno" & a(x2,x1,open) & fullValue("href",x2) →diario.titular.url(x2)

Algunas preguntas a la BC

¿Cuál es el titular?
ask: exists x diario.titular.texto(x)
→ diario.titular.text("Villepin autoriza el toque de queda en los lugares azotados por la violencia callejera")

¿Cuál es la URL de la noticia?
ask: exists x diario.titular.noticia(x)
→ diario.titular.url("/elmundo/2005/11/07/sociedad/1131392990.html")