

# Un sistema de búsqueda de respuestas basado en ontologías, implicación textual y entornos reales \*

## *An User-centred Ontology- and entailment-based Question Answering System*

Óscar Ferrández, Rubén Izquierdo, Sergio Ferrández y José Luis Vicedo

Dept. de Lenguajes y Sistemas Informáticos (Universidad de Alicante)

Carretera San Vicente s/n 03690 Alicante España

{ofe, ruben, sferrandez, vicedo}@dlsi.ua.es

**Resumen:** Este artículo presenta un sistema de Búsqueda de Respuestas basado en ontologías, implicación textual y requerimientos de usuario. Se propone una metodología para la construcción de una base de conocimiento de usuario que nos permite asociar preguntas en lenguaje natural con una representación formal de datos. El núcleo de nuestra estrategia se basa en la implicación textual, la cual permite detectar implicaciones entre preguntas y la base de conocimiento. El sistema ha sido desarrollado para el español y sobre el dominio de cine obteniendo unos resultados prometedores para su utilización en entornos reales.

**Palabras clave:** Búsqueda de Respuestas, Implicación textual, Interfaces de Lenguaje Natural, Modelado de Ontologías

**Abstract:** This paper presents an user-centred ontology- and entailment-based Question Answering system. A methodology is proposed in order to carry out the construction of the user knowledge database. This knowledge database allows us to fill the gap between natural language expressions and formal expressions such as database queries. The core of the system relies on an entailment engine capable of deducting inferences between queries and the knowledge database. The system has been developed for Spanish, covering the cinema domain and obtaining very promising results within real environments.

**Keywords:** Question Answering, Textual Entailment, Natural Language Interfaces, Ontology Modelling

### 1. *Introducción*

La Búsqueda de Respuestas (BR) surge ante la necesidad de recuperar información concreta solicitada por usuarios a partir de preguntas en lenguaje natural. En el caso de sistemas de BR basados en ontologías, los datos donde las respuestas deben ser localizadas poseen una estructura definida en una ontología. Una ontología determina una representación de conceptos y sus relaciones en un dominio específico. Las ontologías juegan un papel esencial en la llamada web semántica<sup>1</sup>,

posibilitando el intercambio de conocimiento. Sin embargo, para la correcta explotación del conocimiento, se necesitan herramientas que comuniquen el lenguaje natural utilizado por los usuarios con la representación lógica de las ontologías.

Con el objetivo de solucionar esta tarea, en este artículo presentamos un sistema de BR basado en ontologías (el cual denominamos QACID) que se compone de dos núcleos principales: una base de conocimiento generada a partir de pruebas realizadas con usuarios reales; 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 soliciten. A cada agrupación se le asocia manualmente una sentencia

\* Esta investigación ha sido parcialmente financiada bajo los proyectos QALL-ME, dentro del Sexto Programa Marco de Investigación de la Unión Europea con referencia FP6-IST-033860, y el Gobierno de España proyecto CICyT número TIN2006-15265-C06-01.

<sup>1</sup>La Web Semántica es una Web extendida y dotada de mayor significado, apoyada en lenguajes universales, que van a permitir que los usuarios puedan encontrar respuestas a sus preguntas de forma más

rápida y sencilla gracias a una información mejor definida.

SPARQL<sup>2</sup> la cual permite el acceso a la información requerida por el usuario. Esta base de conocimiento modela la interacción de los usuarios con el sistema.

Una vez desarrollado el módulo de implicación textual, las preguntas realizadas al sistema en lenguaje natural son procesadas por este módulo, el cual infiere deducciones semánticas entre dichas preguntas y las preguntas agrupadas anteriormente con el objetivo de asociar a una nueva pregunta su sentencia SPARQL correspondiente.

El resto del artículo está estructurado de la siguiente manera: la sección 2 presenta el estado de la cuestión. La sección 3 muestra una descripción detallada del sistema. La sección 4 describe la creación de la base de conocimiento del sistema generada a partir de la interacción con usuarios. A continuación, se presenta el módulo de implicación textual y la sección 6 describe la evaluación y resultados obtenidos. Finalmente, la sección 7 presenta las conclusiones y trabajos futuros.

## 2. Estado de la Cuestión

Las interfaces en lenguaje natural sobre bases de datos han sido extensamente estudiadas (Androutsopoulos, 1996; Copestake y Jones, 1990; Chan y Lim, 2003; Popescu, Etzioni, y Kautz, 2003; Filipe y Mamede, 2000; Minock, 2005). Este tipo de herramientas permite a los usuarios formular sus peticiones sobre bases de conocimiento mediante consultas en lenguaje natural. Existen dos tipos de interfaces en lenguaje natural en función de la capacidad de procesar consultas informales de usuarios:

1. Interfaces en lenguaje natural completas: son sistemas que procesan preguntas en lenguaje natural sin restricciones. Nuestra aproximación se enmarca dentro de esta categoría.
2. Interfaces en lenguaje natural restringidas: comprenden sistemas que procesan preguntas formuladas en un lenguaje controlado. Los usuarios deben aprender previamente dicho lenguaje para poder interrogar la base de conocimiento.

Podemos encontrar un ejemplo típico en (Androutsopoulos, Ritchie, y Thanisch,

<sup>2</sup>SPARQL es un lenguaje estándar de consulta para la recuperación de información desde datos RDF, ([www.w3.org/TR/rdf-sparql-query](http://www.w3.org/TR/rdf-sparql-query)).

1993), el cual procesa la pregunta en lenguaje natural, para transformarla en una forma lógica intermedia, que será más tarde transformada en SQL. Otras aproximaciones (Zelle y Mooney, 1996; Thompson y Mooney, 1999; Zhang y Yu, 2001) hacen uso de métodos de aprendizaje automáticos para transformar preguntas informales en representaciones lógicas estructuradas. Estos sistemas son entrenados usando datos de dominios específicos, proporcionados por Money<sup>3</sup>, y obtienen unos resultados cercanos al 90 %, pero necesitan una gran cantidad de datos para realizar un entrenamiento eficiente. Otros trabajos (Rodrigo et al., 2005; Kang et al., 2004) intentan dividir la pregunta en palabras clave y componer una consulta formal a partir de estas palabras. Finalmente, resaltamos dos sistemas (Popescu, Etzioni, y Kautz, 2003; Wang et al., 2007), los cuales transforman las preguntas en lenguaje natural en consultas estructuradas en SPARQL.

El procesamiento de preguntas en lenguaje natural es normalmente complejo y ambiguo. Por esta razón, muchos sistemas trabajan únicamente sobre preguntas formuladas usando un lenguaje limitado, con restricciones léxicas y gramaticales. Sin embargo, hacer uso de estos sistemas conlleva que el usuario aprenda estos lenguajes restringidos y su sintaxis. El sistema presentado en (Bernstein et al., 2005) hace uso del “*Attempto Controlled English*” para formular consultas. Siguiendo en esta línea, (Popescu, Etzioni, y Kautz, 2003) define la noción de consultas semánticamente tratables, intentando saber a priori los requerimientos de la pregunta para generar la consulta SQL. Sin embargo, en esta aproximación las preguntas que contienen palabras desconocidas no son semánticamente tratables y no pueden ser procesadas.

Los métodos basados en patrones son también usados para resolver tareas relacionadas con interfaces en lenguaje natural. El sistema (Lopez et al., 2007) procesa las preguntas y las clasifica en 23 categorías. Si se consigue clasificar la pregunta de entrada en una de estas 23 categorías, el sistema será capaz de procesarla correctamente. Sin embargo, la limitada cobertura de los patrones conlleva que muchas preguntas no puedan resolverse. Por contra, aunque QACID está basado también en el uso de patrones, evita este proble-

<sup>3</sup>[www.cs.utexas.edu/users/ml/nldata.html](http://www.cs.utexas.edu/users/ml/nldata.html).

ma usando un módulo de implicación textual, que permite obtener relaciones entre patrón y pregunta en la mayoría de los casos. Para finalizar, el sistema (Bernstein y Kaufmann, 2006) ayuda al usuario a construir la consulta evitando la ambigüedad por medio de un motor de búsqueda en lenguaje natural. Sin embargo, de nuevo la capacidad de este sistema está limitada por las restricciones del lenguaje ofrecido.

### 3. Descripción General

Esta sección presenta nuestro sistema de BR basado en ontologías y en la experiencia con usuarios reales: QACID (acrónimo en inglés, *Question Answering on Cinema Domain*). La estructura de QACID se sostiene bajo cuatro componentes principales: la ontología, los datos estructurados, la base de conocimiento de usuario y el módulo de implicación textual.

#### La ontología

Un sistema de BR basado en ontologías procesa información formalmente estructurada sobre un dominio específico determinado por la ontología. En nuestro caso, hemos usado OWL<sup>4</sup> para diseñar una ontología sobre el dominio turístico (trabajo realizado bajo el proyecto de investigación QALL-ME<sup>5</sup>), siendo las clases y relaciones referentes al subdominio *Cinema* las utilizadas en QACID.

#### Los datos

La ontología ha sido poblada con información sobre el dominio de cine provista por la empresa LaNetro<sup>6</sup>, almacenando los datos en formato RDF<sup>7</sup>. Los datos RDF son usados como base de datos donde las respuestas son extraídas por medio de consultas SPARQL.

#### Base de Conocimiento de Usuario

Con el objetivo de conocer las diferentes y múltiples maneras con las que se pueden solicitar información sobre el dominio *Cinema*, decidimos adquirir dicho conocimiento a partir de pruebas realizadas con usuarios reales. Para esto, se solicitó a un grupo de personas que demandaran datos sobre el dominio

*Cinema*, generando así un conjunto de preguntas representativas del dominio.

Seguidamente, las preguntas fueron analizadas y agrupadas automáticamente en función de la información que solicitan, constituyendo de esta forma agrupaciones de preguntas. A cada agrupación se le asoció manualmente una sentencia SPARQL, la cual permite el acceso a la información requerida por el usuario. Como resultado final se obtuvo un conjunto de pares pregunta–consulta SPARQL, que forma la base de conocimiento de usuario del sistema QACID.

#### Módulo de Implicación Textual

El módulo de implicación textual constituye el eje principal de nuestra estrategia de BR. Este módulo implementa técnicas de implicación textual con el objetivo de inferir deducciones semánticas entre preguntas de entrada y las agrupaciones de la base de conocimiento de usuario previamente obtenidas. Este proceso permite asociar consultas SPARQL a las preguntas de entrada y así recuperar las respuestas desde los datos RDF.

En la figura 1 se muestra la arquitectura general de QACID. Las siguientes subsecciones detallan el proceso completo de BR el cual se compone de dos fases principales.

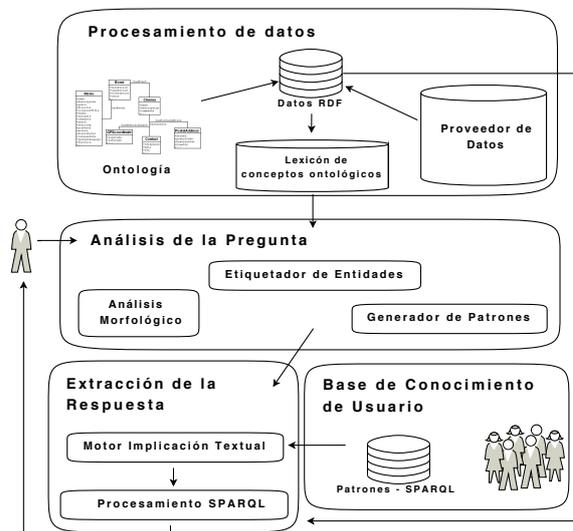


Figura 1: Arquitectura del sistema QACID.

### 3.1. Análisis de la Pregunta

Este módulo se encarga de procesar la pregunta de entrada con el objetivo de obtener una representación formal de la misma (siguiendo el formato de los patrones de la base de conocimiento de usuario).

<sup>4</sup>OWL (del inglés, *Ontology Web Language*) es un lenguaje de marcado para publicar y compartir datos usando ontologías, [www.w3.org/TR/owl-features/](http://www.w3.org/TR/owl-features/).

<sup>5</sup>Descrito en [qallme.itc.it/](http://qallme.itc.it/).

<sup>6</sup>[www.lanetro.com](http://www.lanetro.com).

<sup>7</sup>RDF (del inglés, *Resource Description Framework*) es un modelo de metadatos que propone un método general para el modelado de información, [/www.w3.org/RDF](http://www.w3.org/RDF).

Las preguntas se analizan morfológicamente<sup>8</sup> y se detectan y etiquetan las entidades de la pregunta. Para realizar su etiquetado, QACID aplica técnicas de emparejamiento difuso<sup>9</sup> entre las palabras que contiene la pregunta y un lexicón generado a partir de los datos almacenados en la ontología. Dicho lexicón se obtiene automáticamente e incluye las instancias de la ontología con sus respectivas clases ontológicas (por ejemplo, “*Casino Royale*”  $\Leftrightarrow$  [MOVIE]; “*Alicante*”  $\Leftrightarrow$  [DESTINATION]; “*Cinesa Panoramis*”  $\Leftrightarrow$  [CINEMA]).

La salida de este módulo es una pregunta en lenguaje natural etiquetada con información morfológica y con conceptos de la ontología (ver tercera fila del cuadro 1).

### 3.2. Extracción de la Respuesta

La pregunta es procesada por el módulo de implicación textual con el objetivo de determinar las implicaciones semánticas entre la misma y el conjunto de patrones que contiene la base de conocimiento de usuario. Si este proceso se realiza con éxito, se obtiene una sentencia SPARQL genérica (ver la fila cuarta del cuadro 1) que permite obtener la respuesta a la pregunta. Seguidamente, antes de realizar la consulta a los datos RDF, la sentencia SPARQL se instancia con los datos originales de la pregunta del usuario. En el cuadro 1 se muestra un ejemplo del proceso completo.

<b>Pregunta de entrada</b>
<i>¿Dónde puedo ver Casino Royale?</i>
<b>Análisis morfológico</b>
<i>¿</i> [Fia] <i>Dónde</i> [dónde PT000000] <i>puedo</i> [poder VMIP1S0] <i>ver</i> [ver VMN0000] <i>Casino Royale</i> [casino royale NP00000] <i>?</i> [Fit]
<b>Etiquetado de conceptos de la ontología</b>
<i>¿Dónde puedo ver</i> [MOVIE]?
<b>SPARQL por Implicación Textual</b>
SELECT DISTINCT ?nameCinema WHERE { ?movie name [MOVIE]. ?event hasEvent ?movie. ?event isInSite ?cinema. ?cinema name ?nameCinema }
<b>SPARQL final</b>
SELECT DISTINCT ?nameCinema WHERE {?movie name “ <b>Casino Royale</b> ”.?event hasEvent ?movie.?event isInSite ?cinema.?cinema name ?nameCinema }

Cuadro 1: Ejemplo de proceso de BR.

Las secciones siguientes explican en deta-

<sup>8</sup>Para esta tarea utilizamos *Freeling toolkit*, disponible en [garraf.epsevg.upc.es/freeling/](http://garraf.epsevg.upc.es/freeling/).

<sup>9</sup>Concretamente *secondStrings library*, disponible en [secondstring.sourceforge.net](http://secondstring.sourceforge.net).

lle la creación de la base de conocimiento de usuario y el módulo de implicación textual.

## 4. Base de Conocimiento de Usuario

El principal objetivo de la base de conocimiento es tener una muestra representativa de preguntas que reflejen los intereses y necesidades de los usuarios sobre el dominio. El proceso de construcción de dicha base de conocimiento incluye tres pasos:

1. Generar un conjunto de preguntas significativas de acuerdo al dominio de la ontología. Para esto se seleccionan 50 usuarios de diferente edad, género y nacionalidad. Se les muestra la ontología junto a una lista de entidades reales extraídas de nuestros datos para que generen preguntas sobre cualquier dato de interés. Mediante este proceso se generaron un total de 500 preguntas.

2. Detectar y etiquetar las entidades que aparecen en el conjunto de preguntas. Para esto, se aplica el Anotador de Entidades a las 500 preguntas, reemplazando las entidades detectadas por su concepto correspondiente en la ontología. Por ejemplo, la pregunta “*¿Dónde puedo ver Saw 3?*” se transforma en “*¿Dónde puedo ver [MOVIE]?*”. Como se ha mencionado anteriormente, el Anotador de Entidades está implementado con una técnica de emparejamiento difuso de cadenas, de modo que se intenta emparejar subcadenas en lenguaje natural libre con entidades en nuestro lexicón. Una vez que se completa el proceso, se eliminan aquellas preguntas repetidas y obtenemos un conjunto de 348 preguntas diferentes.

3. Las preguntas anotadas se agrupan manualmente de acuerdo a su equivalencia semántica. Dos preguntas son semánticamente equivalentes cuando ambas solicitan la misma información, y ambas contienen los mismos conceptos ontológicos. Por ejemplo, las preguntas:

“¿Cuál es el número de teléfono del cine [CINEMA]?”

“¿Cuál es el teléfono de contacto del cine [CINEMA]?”

pertenecen al mismo cluster semántico. Se crean un total de 54 agrupaciones semánticamente distintas. Una vez creadas estas agrupaciones, se asocia una consulta SPARQL a cada uno, la cual permitirá obtener la respuesta para cualquiera de las preguntas de la agrupación.

Considerando una agrupación concreta, disponemos de varias preguntas solicitando la misma información, y es muy sencillo extraer conocimiento sobre la forma de requerir esta información. Teniendo en cuenta esto, desarrollamos un recurso llamado *caracterización de atributos de la ontología*. Esta caracterización consiste en saber los diferentes modos en que los usuarios preguntan sobre un atributo concreto. Por ejemplo, considerando las preguntas pertenecientes a la agrupación que solicita información sobre el número de teléfono de un cine, encontramos tres modos distintos que los usuarios han utilizado para preguntar sobre este atributo: *número de teléfono*, *número de contacto* y *número telefónico*. Este conocimiento es muy útil para que el módulo de implicación textual detecte paráfrasis entre los atributos que aparecen en las preguntas.

En resumen, la base de conocimiento está compuesta por 54 agrupaciones. Cada una contiene una media de 6.44 preguntas equivalentes, la consulta SPARQL correspondiente y la información sobre la *caracterización de los atributos de la ontología*.

A pesar de que la generación de la base de conocimiento se ha construido a partir de preguntas realizadas espontáneamente por los usuarios, la mayoría de atributos y relaciones de la ontología (88%) quedan cubiertos por el conjunto final de agrupaciones (por ejemplo, no hay preguntas sobre el e-mail, fax o coordenadas GPS de un cine). Esta alta cobertura demuestra que la metodología de construcción de la base de conocimiento es robusta y apropiada, y que aquellos atributos que quedan fuera son con mucha probabilidad de poco interés para los usuarios.

## 5. Módulo de Implicación Textual

El sistema de implicación textual establece inferencias léxico-semánticas entre una pregunta y el conjunto de patrones predefinidos en la base de conocimiento (ver sección 4). Se consideran relaciones unidireccionales entre dos preguntas, siguiendo la metodología propuesta en (Glickman, 2005) para

relaciones de implicación textual. El sistema utilizado es una extensión del presentado en (Ferrández et al., 2007) adaptando e incorporando nuevas inferencias relevantes para el nuevo paradigma en el que se enfoca.

### 5.1. Inferencias Léxicas

Se componen de un conjunto de medidas<sup>10</sup> léxicas, basadas en las coocurrencias de las palabras y el contexto en que aparecen. Para su cálculo, las preguntas son tratadas como bolsas de palabras (del inglés, *bag-of-words*), lo cual es simple a la vez que preciso, es más rápido computacionalmente y consigue resultados competitivos comparados con otras aproximaciones (ver (Giampiccolo et al., 2007)). Las medidas son:

- Algoritmo Smith–Waterman
- Emparejamiento entre subcadenas consecutivas
- Distancia de Jaro
- Distancia Euclídea
- Coeficiente de similitud de Jaccard
- Emparejamiento entre términos interrogativos

### 5.2. Inferencias basadas en la Ontología

Corresponden con conocimiento directamente derivado de la ontología.

**Restricción sobre los conceptos:** tanto las preguntas como los patrones son etiquetados con conceptos de la ontología. Consecuentemente, se establece una restricción por la que todos los pares de preguntas de las que se puede deducir una relación de implicación deben contener los mismos conceptos, tanto en número como en tipo.

**Inferencia basada en atributos ontológicos:** apoyándose en el conocimiento adquirido sobre las diferentes formas que los usuarios han utilizado para referirse a los atributos de la ontología (i.e. la *caracterización de los atributos de la ontología*, ver sección 4), el sistema implementa una inferencia sobre el atributo o atributos ontológicos que solicitan las preguntas que se realizan al sistema. El procedimiento utiliza la *caracterización de los atributos de la ontología* para detectar la presencia de atributos (normalmen-

<sup>10</sup>Para algunas de ellas se ha usado su implementación de *SimMetrics library* <http://www.dcs.shef.ac.uk/~sam/simmetrics.html>.

te éstos son la información requerida). Aquellos patrones que contengan atributos equivalentes a atributos de la pregunta de entrada serán puntuados positivamente en la detección de implicación textual. Dos atributos son equivalentes si están expresados de la misma manera o usando alguna de sus paráfrasis almacenadas en la *caracterización de los atributos de la ontología*. El peso final obtenido es:

$$Att_{sim} = \frac{\sum_{a_i \in Pg, a_j \in Pt} Eql(a_i, a_j)}{|Pg|} \quad (1)$$

donde  $Pg$  y  $Pt$  contienen los atributos de la pregunta de entrada y el patrón que se está procesando, y  $Eql(a_i, a_j)$  toma el valor:

$$Eql(a_i, a_j) = \begin{cases} 1 & a_i = a_j \text{ o parafrasis,} \\ 0 & \text{otro caso.} \end{cases} \quad (2)$$

Por lo tanto, dos o más atributos pertenecientes a la misma pregunta tienen la misma importancia, a la vez que patrones que no contengan atributos equivalentes serán considerados menos relevantes durante la implicación textual.

Por último, cada inferencia obtiene un factor de similitud entre cero y uno, por lo tanto el coeficiente de implicación final es la suma de todos los factores entre el número de inferencias consideradas. Para decidir las implicaciones, se establece un umbral empírico sobre un conjunto de preguntas de entrenamiento, por lo que preguntas nuevas que obtengan un coeficiente de implicación superior al umbral serán consideradas como deducciones de implicación textual correctas. La siguiente sección detalla la fase de entrenamiento del sistema así como los resultados obtenidos durante la evaluación del mismo.

## 6. Evaluación y resultados

Para comprobar la efectividad de QACID, hemos desarrollado un marco de evaluación sobre el dominio de cine. Debido a que los resultados del sistema están influenciados directamente por el módulo de implicación textual, nos hemos centrado en evaluar la capacidad de éste en la detección correcta de implicaciones entre preguntas.

### 6.1. Marco de Evaluación

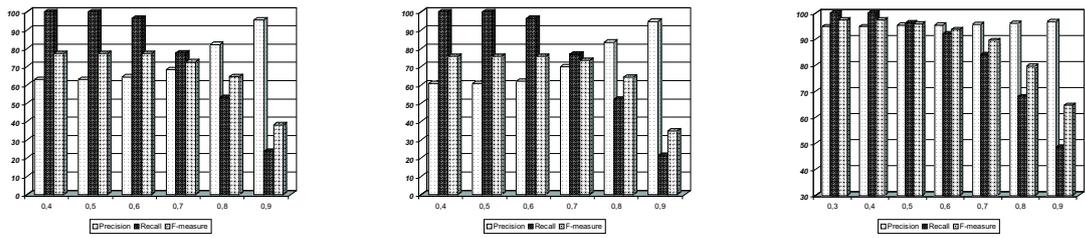
Esta evaluación verificará cómo de representativa y útil es nuestra base de conocimiento a la hora de realizar deducciones, y cómo de preciso es el módulo de implicación textual. Para ello, 10 nuevos usuarios son requeridos para generar una nueva pregunta para cada cluster usando las entidades almacenadas en nuestro lexicón. En total 450 nuevas preguntas que son divididas en dos grupos: el conjunto de entrenamiento y el conjunto de test. El primero grupo se usa para ajustar el umbral de decisión del módulo de implicación textual, de modo que cuando el valor de similitud devuelto por dicho módulo es menor que el umbral establecido, consideramos la pregunta como incierta<sup>11</sup>. El conjunto de test se usa como un conjunto ciego para la evaluación final del sistema. Estos conjuntos comprenden 378 preguntas (7 usuarios) y 162 (3 usuarios) respectivamente. Destacar también que no se eliminan las preguntas repetidas de estos conjuntos, y todas las preguntas pertenecen al dominio de cine (esta evaluación no tiene en cuenta preguntas fuera de dominio).

### 6.2. Análisis de Resultados

La figura 2 muestra la evolución de la precisión, cobertura y medida F dependiendo del umbral de decisión establecido. Cada gráfico corresponde a los experimentos llevados a cabo teniendo en cuenta las inferencias descritas en los apartados 5.1 y 5.2:

- Baseline Léxico (BL): implementa todas las inferencias léxicas (ver sección 5.1) y establece nuestro sistema base para medir la mejora introducida con las inferencias basadas en la ontología.
- BL+Restricción de Conceptos (BL+RC): añade al baseline léxico la restricción referente a la correspondencia entre conceptos de la ontología.
- BL+RC+Inferencia basada en Atributos (BL+RC+IbA): implementa todas las inferencias léxicas y ontológicas, incluyendo aquella basada en la paráfrasis de atributos de la ontología.

<sup>11</sup>Aunque en el cálculo de precisión, cobertura y F, las preguntas inciertas son tratadas como erróneas, cabe destacar que el análisis de las mismas nos ayudará a orientar correctamente el enriquecimiento de conocimiento en el módulo de implicación textual.



(a) *Baseline Léxico (BL)*. (b) *BL+Restricción de Conceptos* (c) *BL+RC+Inferencia basada en Atributos (BL+RC+IbA)*.

Figura 2: *Precisión, cobertura y F-medida para cada inferencia sobre el entrenamiento.*

Como se puede observar, el sistema obtiene buenos resultados para todos los umbrales debido al hecho de que casi todas las nuevas preguntas poseen un patrón correcto asociado en la base de datos. Esto demuestra la eficiencia en la construcción de la base de conocimiento. Teniendo en cuenta el conjunto de preguntas de entrenamiento, el umbral de decisión que obtiene mejor precisión sin comprometer la cobertura es 0.6 para los experimentos BL y BL+RC, y 0.5 para el experimento BL+RC+IbA.

El cuadro 2 muestra los resultados sobre el conjunto de test. Aunque los valores de precisión, cobertura y medida F son ligeramente menores que en el caso del conjunto de entrenamiento el comportamiento del sistema es similar en ambos casos.

Exp.	Test			
	umbral	Prec.	Cob.	F
BL	0.6	57.61	93.21	71.21
BL+RC	0.6	76.64	84.57	80.41
BL+RC+IbA	0.5	<b>89.24</b>	<b>97.53</b>	<b>93.2</b>

Cuadro 2: Resultados sobre el test.

Como era de esperar, el experimento BL obtiene los resultados más bajos, pero se mejoran sustancialmente incluyendo el conocimiento basado en la ontología. La restricción basada en la correspondencia de conceptos (BL+RC) mejora la medida F en 12.91%, mientras que la restricción de alineamiento entre atributos (BL+RC+IbA) produce una mejora de 30.88%.

De esta forma se demuestra la correcta aplicación de: (1) las medidas léxicas sin hacer uso de información ontológica, y (2) el conocimiento semántico adicional extraído de la ontología.

### 7. Conclusión y trabajos futuros

La principal aportación de este artículo es el desarrollo de una metodología para la creación de un sistema de BR sobre dominios restringidos. Dicha metodología hace uso de una ontología que modela el dominio y de una base de conocimiento creada a partir de las necesidades e intereses de los usuarios. Además, el sistema utiliza un módulo de implicación textual para deducir inferencias semánticas entre una nueva pregunta y el conjunto de patrones que almacena en su base de conocimiento.

Los resultados obtenidos muestran una alta precisión del sistema, y en concreto del módulo de implicación textual. Además, estos valores han sido obtenidos sin el uso de complejos recursos semánticos que pudieran comprometer la eficiencia del sistema.

La metodología propuesta se podría extrapolar a otros dominios u otros idiomas, se necesitaría una ontología que modelara el nuevo dominio y un conjunto de usuarios para crear la base de conocimiento en el idioma requerido.

Finalmente, como trabajo futuro se plantea la extensión del sistema mediante la detección de expresiones temporales y espaciales. Preguntas como: “¿Cuál es el cine más cercano donde puedo ver [MOVIE]?” o “¿Dónde puedo ver [MOVIE] mañana?” entrarían dentro de esta línea. Además, aunque la detección de la respuesta esperada es en mayor o menor medida considerada por nuestra *caracterización de atributos*, también se plantea añadir un módulo específico que incorpore este conocimiento al módulo de implicación textual. Por ejemplo, en la pregunta “Dime dónde fue rodada la película [MOVIE]” esta información sería de gran valor ya que en ningún momento se menciona el

atributo “*estudio*”. Sin embargo si se formulara como “*Dime en qué estudio fue rodada la película [MOVIE]*”, nuestra *caracterización de atributos* aportaría el conocimiento suficiente para el módulo de implicación textual.

## Bibliografía

2005. *The Semantic Web - ISWC 2005, 4th International Semantic Web Conference, ISWC 2005, Galway, Ireland, November 6-10, 2005, Proceedings*, volumen 3729 de *Lecture Notes in Computer Science*. Springer.
- Androutsopoulos, I., G. Ritchie, y P. Thanisch. 1993. An Efficient and Portable Natural Language Query Interface for Relational Databases. In: *6th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*, páginas 327–320.
- Androutsopoulos, Ion. 1996. A Principled Framework for Constructing Natural Language Interfaces to Temporal Databases. *CoRR*, cmp-lg/9609004.
- Bernstein, Abraham y Esther Kaufmann. 2006. GINO - A Guided Input Natural Language Ontology Editor. En *International Semantic Web Conference*, volumen 4273 de *Lecture Notes in Computer Science*, páginas 144–157. Springer.
- Bernstein, Abraham, Esther Kaufmann, Anne Göhring, y Christoph Kiefer. 2005. Querying Ontologies: A Controlled English Interface for End-Users. En *International Semantic Web Conference (DBL, 2005)*, páginas 112–126.
- Chan, Hock Chuan y John Lim. 2003. A Review of Experiments on Natural Language Interfaces. En *Advanced Topics in Database Research, Vol. 2*. páginas 55–71.
- Copestake, A. y K.S. Jones. 1990. Natural Language Interfaces to Databases. *Knowledge Engineering Review*, 5(4):225–249.
- Ferrández, Óscar, Daniel Micol, Rafael Muñoz, y Manuel Palomar. 2007. A Perspective-Based Approach for Solving Textual Entailment Recognition. En *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, páginas 66–71, Prague. Association for Computational Linguistics.
- Filipe, Porfírio P. y Nuno J. Mamede. 2000. Databases and Natural Language Interfaces. En Carlos Delgado Esperanza Marcos, y José Manuel Marqués Corral, editores, *JISBD*, páginas 321–332. Universidad de Valladolid, Departamento de Informática.
- Giampiccolo, Danilo, Bernardo Magnini, Ido Dagan, y Bill Dolan. 2007. The Third PASCAL Recognizing Textual Entailment Challenge. En *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, páginas 1–9, Prague. Association for Computational Linguistics.
- Glickman, Oren. 2005. *Applied Textual Entailment*. Ph.D. tesis, Bar Ilan University.
- Kang, In-Su, Seung-Hoon Na, Jong-Hyeok Lee, y Gijoo Yang. 2004. Lightweight Natural Language Database Interfaces. En *NLDB*, volumen 3136 de *Lecture Notes in Computer Science*, páginas 76–88. Springer.
- Lopez, Vanessa, Victoria S. Uren, Enrico Motta, y Michele Pasin. 2007. AquaLog: An ontology-driven question answering system for organizational semantic intranets. *J. Web Sem.*, 5(2):72–105.
- Minock, Michael. 2005. A Phrasal Approach to Natural Language Interfaces over Databases. En *NLDB*, volumen 3513 de *Lecture Notes in Computer Science*, páginas 333–336. Springer.
- Popescu, Ana-Maria, Oren Etzioni, y Henry A. Kautz. 2003. Towards a theory of natural language interfaces to databases. En *Intelligent User Interfaces*, páginas 149–157. ACM.
- Rodrigo, Luis, V. Richard Benjamins, Jesús Contreras, Diego Patón, D. Navarro, R. Salla, Mercedes Blázquez, P. Tena, y I. Martos. 2005. A Semantic Search Engine for the International Relation Sector. En *International Semantic Web Conference (DBL, 2005)*, páginas 1002–1015.
- Thompson, Cynthia A. y Raymond J. Mooney. 1999. Automatic Construction of Semantic Lexicons for Learning Natural Language Interfaces. En *AAAI/IAAI*, páginas 487–493.
- Wang, Chong, Miao Xiong, Qi Zhou, y Yong Yu. 2007. PANTO: A Portable Natural Language Interface to Ontologies. En *ESWC*, volumen 4519 de *Lecture Notes in Computer Science*, páginas 473–487. Springer.
- Zelle, John M. y Raymond J. Mooney. 1996. Learning to Parse Database Queries Using Inductive Logic Programming. En *AAAI/IAAI, Vol. 2*, páginas 1050–1055.
- Zhang, Lei y Yong Yu. 2001. Learning to Generate CGs from Domain Specific Sentences. En *ICCS*, volumen 2120 de *Lecture Notes in Computer Science*, páginas 44–57. Springer.