Una Arquitectura de un Sistema de Búsqueda de Respuestas

Alejandra Carolina Cardoso[1], Agustina Bini[2], M. Alicia Pérez Abelleira[3]

[1,2,3]Facultad de Ingeniería e IESIING

Universidad Católica de Salta

Campo Castañares, Salta A4400

[1]acardoso@ucasal.net, [2]aperez@ucasal.net, [3]agubini@hotmail.com

Resumen: *Dada la gran cantidad de información no estructurada disponible en todo tipo de organizaciones, la minería de textos va creciendo en importancia, y en particular el tipo de sistemas que pretenden responder a preguntas expresadas en lenguaje natural, los sistemas de búsqueda de respuestas. Este trabajo describe la arquitectura de un sistema con esas características que responde a preguntas de usuarios cuyas respuestas están en un corpus de más de ocho mil documentos que contienen resoluciones rectorales.*

Palabras Claves: *búsqueda de respuestas, minería de textos, UIMA.*

Abstract: *All kinds of organizations store large amounts of unstructured information. Thus text mining keeps growing in importance, and in particular ever greater attention is being given to the kind of systems who answer user questions in natural language, namely question answering systems. This paper describes the architecture of a one such system that finds answers to user questions in a corpus of more than eight thousand documents corresponding to university academic and administrative resolutions.*

Keywords:*question answering systems, text mining, UIMA.*

INTRODUCCIÓN

La búsqueda de respuestas (BR) tiene como objetivo dar respuestas en lenguaje natural a preguntas también en lenguaje natural. Extendiendo esta definición, puede considerarse como un proceso interactivo entre un usuario y el sistema en el que el segundo pretende comprender la necesidad de información del primero expresada como una consulta en lenguaje natural, recuperar datos o información relevantes para responderla, priorizar entre éstos y finalmente presentar las respuestas relevantes al usuario (García Cumbreras, 2010).

Aunque el problema de BR ha sido estudiado desde hace más de diez años, continúa siendo un desafío que incorpora varias tareas del ámbito de la minería de textos, del procesamiento del lenguaje natural y otras técnicas para poder (a) comprender adecuadamente las necesidades de información de la pregunta, (b) obtener una lista de respuestas candidatas a partir de los documentos, y (c) filtrarlas en base a evidencia que justifique que cada una de esas respuestas es la correcta.

En este trabajo presentamos una primera aproximación a un sistema de BR que puede contestar preguntas factoides sobre un corpus de más de 8000 documentos que contienen 9 años de resoluciones rectorales de la Universidad Católica de Salta en distintos formatos (Word, texto plano, PDF). Este trabajo es continuación de una línea de investigación en minería de textos que ha desarrollado una arquitectura para búsqueda semántica (Pérez & Cardoso, 2011) (Pérez & Cardoso, 2013). Tras una introducción a los sistemas de búsqueda

de respuestas en la Sección 2, la sección 3 describe dicha arquitectura de búsqueda semántica.

El sistema de BR propuesto está formado de los siguientes componentes: análisis de la pregunta, que incluye su categorización y la construcción de la consulta; recuperación de documentos en base a la consulta; y extracción de las respuestas candidatas y presentación al usuario. Estos componentes se describen a partir de la Sección 4. El artículo concluye evaluando el desarrollo actual de este sistema y enunciando algunas conclusiones.

SISTEMAS DE BÚSQUEDA DE RESPUESTAS

Ya en los años 70 Wendy Lehnert (Lehnert, 1977) propuso la primera aproximación a un sistema de BR y sus características ideales, según las cuales el sistema debía entender la pregunta del usuario (comprensión del lenguaje natural), buscar la respuesta en una base documental (búsqueda de conocimiento), y componer la respuesta y mostrarla al usuario (generación de lenguaje natural). Esta estructura de tres componentes se ha preservado en la mayoría de los sistemas actuales.

Los sistemas de BR comenzaron a desarrollarse en el ámbito de la inteligencia artificial y después en el de los sistemas de recuperación de la información. Así se consiguieron avances importantes en la fase de recuperación de documentos o pasajes susceptibles de contener la respuesta correcta, pero no tanto en la comprensión de las preguntas y extracción de las respuestas. Los desarrollos posteriores incluyeron diversos recursos lingüísticos procedentes del ámbito del procesamiento del lenguaje natural, para ayudar a la comprensión de los textos y de las preguntas. La cantidad y complejidad de los recursos lingüísticos utilizados ha sido variable, incluyendo etiquetadores POS (part-of-speech), analizadores sintácticos, extractores de entidades con nombre (NER), diccionarios, recursos externos como bases de datos léxico semánticas y ontologías, y hasta técnicas de análisis semántico y contextual. Algunas técnicas, especialmente las correspondientes a los primeros niveles del análisis de lenguaje natural, es decir los niveles léxico y sintáctico, han sido efectivas. Sin embargo, construir recursos más sofisticados es una tarea compleja y no necesariamente llega a mejores resultados, con lo cual a menudo no se justifica la mejora de rendimiento con el esfuerzo empleado en su aplicación y con los tiempos de ejecución empleados en el proceso completo (García Cumbreras, 2010)

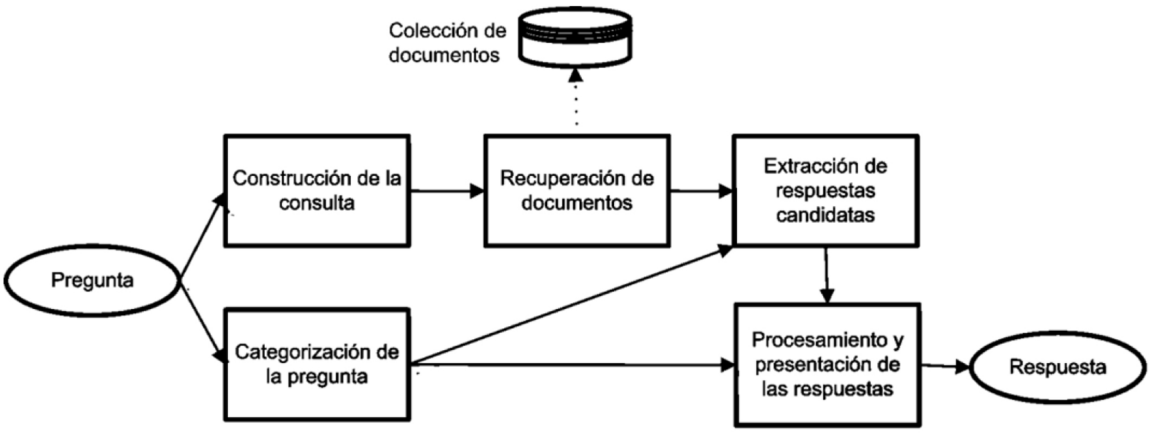

Figura 1. Arquitectura básica para la búsqueda de respuestas [3].

(Webber & Webb, 2010) (Montes-y-Gómez, Villaseñor-Pineda, & López-López, 2008).

Un sistema típico de búsqueda de respuestas supone una serie de procesos que comienzan tomando la pregunta del usuario como entrada y terminan respondiendo con una respuesta o una lista de respuestas priorizadas, con indicaciones de la fuente de la información. Pasca (Pasca, 2007) caracterizó el paradigma de-facto para la búsqueda de respuestas como: recuperar documentos potencialmente relevantes, extraer las respuestas potenciales, devolver las respuestas con mayor ranking. Para procesar las preguntas suelen realizarse dos pasos: identificar qué tipo de información se busca (categorización de la pregunta), y elaborar la consulta que ubique en la colección de documentos fragmentos de texto con posibilidades de contener la respuesta. La Figura 1 muestra una arquitectura genérica para un sistema de BR con los componentes mencionados.

ANTECEDENTES

La línea de trabajo en la que se encuadra este proyecto ha estado centrada en el uso de técnicas de minería de texto para la explotación de información no estructurada. Una aplicación de gestión de la información no estructurada (UIM por sus siglas en inglés) típicamente es un sistema que analiza grandes volúmenes de información no estructurada con el fin de descubrir, organizar y entregar conocimiento relevante al usuario final. La información no estructurada puede ser mensajes de correo electrónico, páginas web o documentos generados con una variedad de procesadores de texto, como en el caso de las resoluciones rectorales de nuestra universidad. Estas aplicaciones utilizan para el análisis una variedad de tecnologías en las áreas del procesamiento del lenguaje natural, recuperación de la información, aprendizaje automático, ontologías y hasta razonamiento automático.

El resultado del análisis generalmente es información estructurada que se hace accesible al usuario mediante aplicaciones adecuadas. Un ejemplo puede ser la generación de un índice de búsqueda y la utilización de un buscador que facilita el acceso a documentos de texto por tema, ordenados según su relevancia a los términos o conceptos de la consulta del usuario (Pérez & Cardoso, 2011). Los sistemas de búsqueda de respuestas son una instancia de este tipo de aplicaciones.

Conceptualmente las aplicaciones de gestión de

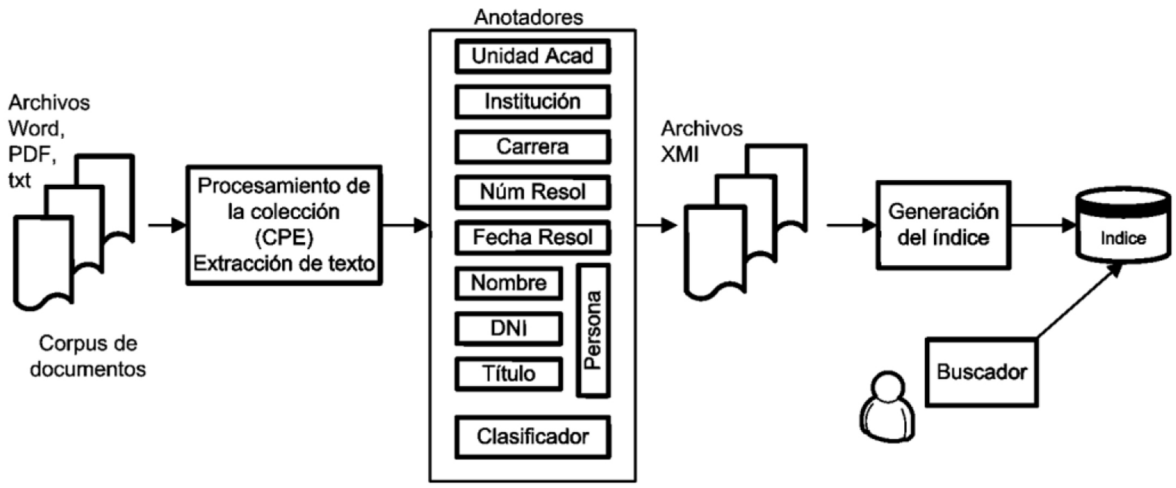

Figura 2. Arquitectura del sistema de gestión de información no estructurada.

información no estructurada suelen organizarse en dos fases. En la fase de análisis se recogen y analizan colecciones de documentos y los resultados se almacenan en algún lenguaje o depósito intermedio. La fase de entrega hace accesible al usuario el resultado del análisis, y posiblemente el documento original completo mediante una interfaz apropiada. La Figura 2 muestra la aplicación de este esquema a nuestro dominio (Pérez & Cardoso, 2011), en el que partimos de más de 8000 resoluciones rectorales en archivos de texto de distinto tipo: Word, PDF, texto plano. Previo al análisis, se procede a la extracción del texto de cada archivo utilizando herramientas de software libre (poi.apache.org y tm-extractors). El texto se normaliza eliminando acentos para facilitar los procesos de búsqueda y equiparación de cadenas. También se divide en partes la resolución extrayendo el encabezado (texto que contiene el número y la fecha de la resolución) y el cuerpo con la mayor parte de la información, y descartando en lo posible el texto "de forma".

La fase de análisis incluye tokenización y detección de entidades en documentos individuales tales como personas, fechas, organizaciones, unidades académicas y metadatos de la resolución (fecha y número). Además con la ayuda de un clasificador aprendido automáticamente del corpus de resoluciones se anota cada documento con una categoría (Pérez & Cardoso, 2011). Existen 21 categorías que fueron obtenidas del personal especializado en la elaboración de resoluciones. Algunos ejemplos son: designación de planta docente, convenio de pasantías, convenio de colaboración, o llamado a concurso docente.

El resultado de la fase de análisis es un conjunto de archivos en formato XMI. Estos archivos contienen,

```
<?xml version="1.0" encoding="UTF-8" ?>

<xmi:XMI ... xmi:version="2.0">

...

  <cas:Sofa xmi:id="1" sofaNum="2" sofaID="encabezado" mimeType="text" sofaString="ResoluciOn N 470/08 En el Campo Castanares, sito en la Ciudad de Salta, Capital de la Provincia del mismo nombre, Republica Argentina, sede de la Universidad Catolica de Salta, a los veintisiete dias el mes de mayo del ano dos mil ocho:" />

  <cas:Sofa xmi:id="13" sofaNum="1" sofaID="_InitialView" mimeType="text" sofaString="la presentacion efectuada por las autoridades de la Escuela Universitaria de Educacion Fisica, dependiente de la Facultad de Artes y Ciencias en virtud de la cual se propone las modificaciones de designaciones docentes Planta Transitoria, para la carrera Licenciatura en Educacion Fisica…" />

...

  <examples:SourceDocumentInformation xmi:id="25" sofa="13" begin="0" end="0" uri="file:/D:/UIMA/docs/RES.%20%20Nº0470-08.txt" offsetInSource="0" documentSize="2280" />

  <ucs:UA xmi:id="48" sofa="13" begin="177" end="208" confidence="30.0" componentId="de.julielab.jules.lingpipegazetteer.GazetteerAnnotator" specificType="UnidadAcademica" />

  <ucs:Carrera xmi:id="72" sofa="13" begin="398" end="430" confidence="0.0" componentId="de.julielab.jules.lingpipegazetteer.GazetteerAnnotator" specificType="Carrera" />

  <ucs:NumeroResol xmi:id="33" sofa="1" begin="0" end="19" nroResol="RESOLUCION N 470/08" numero="470" anio="2008" />

  <ucs:FechaResol xmi:id="40" sofa="1" begin="196" end="248" anio="2008" mes="MAYO" dia="27" fechaResolCompleta="VEINTISIETE DE MAYO DE DOS MIL OCHO" />

  <ucs:Clase xmi:id="28" sofa="1" begin="0" end="0" valor=" DesigDocPlanta" />

...

</xmi:XMI>
```

Figura 3. Ejemplo de texto anotado.

además de las partes relevantes del texto original, metadatos en forma de anotaciones correspondientes a las entidades existentes y a la categoría de documentos. Estos archivos serán procesados para construir el índice de un motor de búsqueda que contiene los tokens (en nuestro caso, las palabras que aparecen en el texto) y las entidades y categorías extraídas automáticamente.

En la fase de entrega existe una interfaz para hacer búsquedas en el índice. El usuario puede buscar documentos que contengan combinaciones booleanas de entidades, categorías y tokens mediante un motor de búsqueda semántica.

Las dos fases están desarrolladas sobre UIMA (Unstructured Information Management Architecture), una arquitectura basada en componentes para construir sistemas de procesamiento de información no estructurada (Ferrucci & Lally, 2004). En UIMA, el componente que contiene la lógica del análisis se llama anotador. Cada anotador realiza una tarea específica de extracción de información de un documento y genera como resultado anotaciones, que son añadidas a una estructura de datos denominada CAS (common analysis structure). A su vez, esas anotaciones pueden ser utilizadas por otros anotadores. Los anotadores pueden ser agrupados en anotadores agregados. La mayoría de nuestros anotadores realizan reconocimiento de entidades con nombre (NER), a saber: personas, unidades académicas, carreras, instituciones; además hay otros que extraen fechas, número y año de las resoluciones. Para detectar entidades correspondientes a personas se agregan otras (nombres propios, DNIs y títulos) obtenidas por los anotadores correspondientes. Un último anotador asigna la categoría de documento utilizando el modelo aprendido automáticamente. Inicialmente los anotadores para NER fueron codificados a mano; luego fueron reemplazados o complementados con modelos aprendidos automáticamente usando campos aleatorios condicionales (CRFs) (Pérez & Cardoso, 2013).

El trabajo descrito en este artículo construye sobre esta arquitectura, añadiendo, en la fase de entrega, un sistema de búsqueda de respuestas mediante el cual el corpus anotado de documentos sirve para contestar preguntas en lenguaje natural.

FORMATO DE LA INFORMACIÓN ESTRUCTURADA

Un importante principio de arquitectura en UIMA es que todos los analizadores operan sobre una estructura de datos estándar, el CAS, que incluye el texto y las anotaciones. El CAS es el principal repositorio de información estructurada y utiliza como lenguaje el estándar XMI (XML Metadata Interchange) (OMG, 2007). XMI proporciona un formato en XML para el intercambio de modelos UML, lo que además hace posible la interoperabilidad con otras aplicaciones. En XMI el texto del documento se incluye (opcionalmente) en uno o más SofAs (Subject of Analysis) del CAS como atributo sofaString. Las anotaciones tienen prefijos particulares en el espacio de nombres de XMI, como por ejemplo <ucs:Carrera>. El espacio de nombres queda definido al declarar un sistema de tipos como parte del dominio en UIMA. La Figura 3 muestra un ejemplo del resultado del proceso de análisis. Dos SofAs recogen el encabezado, del que sólo se extrae la fecha y el número de resolución, y el cuerpo de la misma. La primera anotación del tipo SourceDocument Information guarda metadatos sobre el archivo, para poder recuperarlo posteriormente, por ejemplo, para mostrarlo al usuario como resultado de una búsqueda. A continuación aparecen varias anotaciones, algunas de ellas con varios campos. La anotación Clase contiene la categoría DesigPlanta asignada a esta resolución por el modelo aprendido.

CATEGORIZACIÓN DE LAS PREGUNTAS

Para una computadora, el método más sencillo de interpretar una pregunta hecha en lenguaje natural sería eliminar las denominadas palabras vacías o stopwords, incluyendo quién, cuál o dónde, y convertir el resto de la pregunta en una consulta booleana. Sin embargo, al hacer esto se está desperdiciando información de gran utilidad para reducir el alcance de la pregunta. Por ejemplo, eliminando la palabra "cuándo" de la frase "¿Cuándo murió Güemes?" puede llevar a recuperar respuestas sobre cómo y dónde, en lugar de recuperar solo respuestas con la fecha solicitada. Por ello es que en los sistemas de BR se definen categorías de preguntas. La categorización de la pregunta tiene como objetivo asociar a la misma una categoría que indique el tipo de información que se busca, por ejemplo, el nombre de una persona que cumpla una serie de características, una fecha, una definición, etc. En general las entidades con nombre identificadas en la colección de documentos son candidatos típicos como objetivo de una pregunta. Por tanto, estas categorías asociadas a las preguntas proporcionan restricciones semánticas a la hora de extraer las respuestas candidatas.

Durante el análisis de la pregunta se ubican identificadores del tipo de pregunta, que después pueden compararse con patrones típicos para cada una de las categorías definidas. De hecho, en la literatura queda claro que determinar el dominio de la pregunta y caracterizar el tipo de la respuesta buscada son pasos esenciales en los sistemas de búsqueda de respuestas. No obstante existen una variedad de enfoques que varían en el número de categorías determinadas, la estructura (plana o jerárquica) de la clasificación y la elección de las categorías propiamente dichas (Amaral, Laurent, Martins, Mendes, & Pinto, 2004) (Webber & Webb, 2010).

Como se indicó en la Seccion 1, el sistema descrito en este trabajo tiene como objetivo contestar preguntas factoides sobre un corpus de más de 8000 documentos que contienen resoluciones rectorales de la Universidad Católica de Salta. Las preguntas factoides requieren normalmente un hecho sencillo como respuesta, por ejemplo una fecha, el nombre de alguien o algo, o una cantidad. En nuestro caso, para establecer las posibles categorías de preguntas nos guiamos por el conjunto de entidades con nombre y otros elementos que los anotadores son capaces de detectar en el corpus, y que en general son el foco de las respuestas esperadas. Es en base al tipo de anotaciones que se estableció la clasificación. Para cada una de las anotaciones se determinó el patrón de comienzo de la pregunta, que en general incluye un pronombre interrogativo, ya que por el momento se trata en todos los casos de preguntas factoides.

Una vez establecida la clasificación, para la categorización de una pregunta dada en la literatura suele utilizarse uno de estos dos enfoques: reglas construidas manualmente, por ejemplo, "si la pregunta comienza con Quién, el tipo de la pregunta es PERSONA"; y clasificadores aprendidos automáticamente que predicen la P(TipoDePregunta | Pregunta) donde la pregunta se representa como un conjunto de atributos. Estos clasificadores son aprendidos en base a conjuntos de preguntas categorizadas manualmente. En nuestro caso, hicimos un análisis exhaustivo de ejemplos de preguntas significativas en nuestro corpus, y dada la limitada variedad de las mismas el primer enfoque es suficiente, evitando además la construcción del conjunto de preguntas necesario para entrenar un clasificador.

En base a lo expuesto, la Tabla 1 muestra las categorías de preguntas relevantes en nuestro corpus particular, organizadas en base al tipo de respuesta esperada.

Cada una de las categorías de preguntas lleva asociados patrones típicos de la pregunta, como se explica a continuación.

RESPUESTA ESPERADA	PREGUNTA
PERSONA	¿QUIÉN... ¿A QUIÉN... ¿QUINES...
INSTITUCIÓN	¿A QUÉ... ¿QUÉ INSTITUCIÓN...
	¿QUÉ EMPRESA...
UNIDAD ACADÉMICA	¿QUÉ FACULTAD...
	¿QUÉ UNIDAD ACADÉMICA...
	¿EN QUÉ UNIDAD ACADÉMICA...
	¿QUÉ ESCUELA... ¿EN QUÉ ESCUELA...
	¿QUÉ SEDE... ¿EN QUÉ SEDE...
	¿QUÉ DELEGACIÓN...
	¿EN QUÉ DELEGACIÓN...
CARRERA	¿QUÉ CARRERA.. ¿EN QUÉ CARRERA...
FECHA	¿CUÁNDO... ¿EN QUÉ MES...
	¿EN QUÉ AÑO... ¿EN QUÉ FECHA...
RESOLUCIÓN	¿EN QUÉ RESOLUCIÓN...
	¿CUÁL ES EL NÚMERO DE LA RESOLUCIÓN EN QUE...
TÍTULO	¿QUÉ TÍTULO...

Tabla 1. Clasificación de las preguntas según el tipo de respuesta esperada.

CONSTRUCCIÓN DE LA CONSULTA

Después de haber determinado el tipo de pregunta, es necesario generar una consulta que recupere documentos del índice, que deberán contener una o más anotaciones correspondientes al tipo de la pregunta, y los conceptos relevantes de la pregunta del usuario, y todo ello dentro de una ventana de texto de longitud limitada. En esta sección se describe la construcción de las consultas, explicando previamente el lenguaje utilizado para expresar las mismas.

FORMATO DE LAS CONSULTAS

Para construir las consultas hemos adoptado el lenguaje de consultas XML Fragments (Chu-Carroll, Prager, Czuba, Ferrucci, & Duboue, 2006) debido a su expresividad y a la disponibilidad de un motor de búsquedas que acepta consultas en este lenguaje. El motor acepta en las consultas la sintaxis de las anotaciones de UIMA y además texto libre y operadores estándar como AND, OR, NOT para combinar los fragmentos que componen las consultas.

Una consulta en el lenguaje XML Fragments consta de una estructura XML sub-especificada que combina consultas de palabras con consultas de información anotada. Esto permite buscar conceptos más específicos, e incluso relaciones entre objetos (por ejemplo, “la persona y la unidad académica deben aparecer en la misma frase” o “una persona que pertenece a cierta unidad académica” o “una unidad académica vinculada a una institución” porque aparecen cerca en el documento).

Las operaciones que se pueden realizar con XML Fragments incluyen (Chu-Carroll, Prager, Czuba, Ferrucci, & Duboue, 2006):

- Conceptualización, que generaliza un literal (string) a un concepto del sistema de tipos. Por ejemplo, la consulta “institución” devuelve documentos en que aparezca esa palabra, mientras que <Institución>.<Institución/> busca ocurrencias de la anotación institución, aunque la palabra “institución” misma no aparezca en el documento.

- Restricción: restringe las ocurrencias de etiquetas XML indicando qué palabras deben aparecer. Por ejemplo <Institución> ingenier </Institución> devuelve ocurrencias de la institución Consejo Profesional de Ingenieros mientras que <UnidadAcadémica> ingenier </UnidadAcadémica> devuelve documentos sobre la Facultad de Ingeniería.

- Relación: una anotación representa la relación entre los términos de la consulta. Ej. <FechaResol año=2007 mes=Septiembre><FechaResol> encuentra las resoluciones de septiembre del 2007 (y no simplemente resoluciones de 2007 en que aparezca “septiembre”).

CONSTRUCCIÓN DE LAS CONSULTAS

El presente trabajo describe el prototipo inicial del sistema de BR, en que solo se han conside-

rado preguntas que tienen como respuesta una persona y en particular que comienzan con el pronombre interrogativo “Quién”. En esta sección se describe el proceso de armado de la consulta en el lenguaje XML Fragments a partir de la consulta en lenguaje natural del usuario. Por comenzar con “Quién”, como se indicó en la Sección 4, se espera que la respuesta sea una persona, en particular un fragmento de algún documento que haya sido anotado como persona. Por tanto, parte de la consulta será la expresión <Persona>.</Persona>, que indica que el texto devuelto debe contener una anotación de tipo persona.

Para construir la consulta es necesario analizar la pregunta. Hay diversos enfoques más o menos complejos de “comprensión” de la pregunta, pero para nuestro caso un enfoque sencillo, basado en características léxico-sintácticas es suficiente. Este análisis sintáctico realizado se denomina chunking. Podría considerarse una variante superficial de un análisis sintáctico completo (parsing o deep parsing), y ahorra tiempo de procesamiento al mismo tiempo que se centra en los elementos claves de la frase, en este caso la consulta, como son el verbo y los sintagmas nominales, ignorando otras partes de la frase, lo cual es suficiente para nuestra tarea. No necesitamos un análisis sintáctico completo, sino lo suficiente para obtener las partes claves de la pregunta (Ingersoll, Morton, & Farris, 2013).

En el procesado de la pregunta se procedió a anotar la misma de dos formas diferentes:

(a) utilizando el etiquetador POS de FreeLing, una colección de herramientas de código abierto para el análisis del lenguaje (http://nlp.lsi.upc.edu/freeling/).

(b) utilizando los anotadores desarrollados o aprendidos automáticamente en UIMA (sección 3) que detectan Personas, Títulos, Instituciones, Unidades Académicas, Carreras y Fechas. Frecuentemente las preguntas tienen referencias a entidades con nombre, tales como nombres propios, organizaciones, lugares, fechas etc. Es de utilidad que estas entidades sean gestionadas como un solo token, en lugar de como varias palabras.

Una vez anotada la pregunta, se utilizan patrones para extraer los componentes relevantes de la misma. En el caso de las preguntas “Quién” el foco es el sintagma nominal que aparece después del verbo. De forma heurística detectamos en la pregunta componentes que podríamos llamar grupos nominales que aparecen después del verbo, afines a los sintagmas nominales pero sin la garantía de realizar un análisis sintáctico completo. Cada uno de estos grupos nominales puede ser un nombre común, un nombre común seguido de un adjetivo, un nombre propio. Además, si alguno de estos grupos va separado de otro nombre por una sola palabra, en general una preposición, se anexa éste al grupo. Para cada uno de estos grupos nominales gn, se añade a la consulta la cadena <>”gn”</>

A estos términos obtenidos mediante la aplicación del patrón se añaden los devueltos por los anotadores, que ya están en el lenguaje XML Fragments. Las Tablas 2 y 3 presentan dos ejemplos.

En el prototipo actual es preciso que la pregunta se formule de manera correcta sintáctica y ortográficamente para poder obtener los resultados correctos. Por otro lado, este enfoque da pie a dificultades. En el segundo ejemplo si el evaluador es una mujer, no se obtendrá respuesta, ya que el texto del documento buscado probablemente incluya la palabra “evaluadora” en lugar de “evaluador”. Un uso adecuado de la lematización puede resolver algunos de estos problemas.

RECUPERACIÓN DE DOCUMENTOS RELEVANTES

La indexación y recuperación de documentos está basada en el paquete SemanticSearch 2.1 (Apache UIMA Development Community, 2014) el cual añade a UIMA un motor de búsqueda semántica.

La búsqueda semántica es una búsqueda en que se especifica la intención semántica de la consulta, no solo los términos que se desean encontrar. Para ello se incluyen en la consulta entidades con nombre, anotadas como tal, o hasta sus atributos.

Pregunta: ¿Quién fue designado decano de la Facultad de Ingeniería e Informática en el 2008?
Tipo de pregunta: ¿Quién. **Respuesta esperada:** Persona.
Anotaciones: <UA> Facultad de Ingrniería e Informática</Ua>
Etiquetas proporcionadas por Freeling: rue ---- > VERBO designado ----> VERBO decano ----> NOMBRE_COMÚN de ----> OTRO la ----> otro Facultad_de_Ingeniería ----> NOMBRE_PROPIO e ----> OTRO Informática ----> NOMBRE_PROPIO en ----> OTRO el ----> OTRO 2008 ----> NUMERAL
Consulta resultante en XML Fragments: +<Persona> , </Persona> + <>decano</> +<UA>Facultad de Ingeniería e Informática</UA> +<FechaResol_anio="2008"></FechaResol>

Tabla 2. Construcción de la consulta: Ejemplo 1.

Por ejemplo, se podría especificar que se está buscando una carrera de “ingeniería” (<Carrera>ingeniería</Carrera>), y tal consulta no devolvería resultados sobre otros usos de la palabra ingeniería.

Nuestra implementación de búsqueda semántica usando SemanticSearch incluye un CAS Consumer que llena un índice con el contenido del documento y con las anotaciones añadidas por los anotadores ya descriptos implementados en UIMA. Las facilidades de búsqueda incluyen, además de la búsqueda de términos en el contenido del documento, la búsqueda por anotaciones. Ambas se realizan mediante el lenguaje XML Fragments ya descrito. Este paquete es software libre aportado por IBM y para acceder al índice existe una API que puede ser utilizada en una aplicación en Java.

Pregunta: ¿Quién ha sido el evaluador externo de proyectos del Consejo de Investigación?
Tipo de pregunta: ¿Quién. **Respuesta esperada:** Persona.
Anotaciones: <UA>Consejo de Investigaciones</Ua>
Etiquetas proporcionadas por Freeling: ha ---- > VERBO sido ----> VERBO evaluador ----> NOMBRE_COMÚN externo ----> ADJETIVO de ----> otro proyectos ----> NOMBRE_COMÚN de ----> OTRO el ----> OTRO Consejo_de_Investigaciones ----> NOMBRE PROPIO
Consulta resultante en XML Fragments: +<Persona> , </Persona> + <>evaluador externo de proyectos</>+<UA>Consejo de Investigaciones</UA>

Tabla 3. Construcción de la consulta: Ejemplo 2.

EXTRACCIÓN Y PRESENTACIÓN DE RESPUESTAS

La consulta al índice devuelve una lista de d archivos XMI candidatos a contener la respuesta buscada. Estos documentos contienen al menos una anotación del tipo de la respuesta, además de las anotaciones y palabras clave detectadas a la hora de armar la consulta. Los d documentos están ordenados en un ranking en base a factores tales como la capacidad de los términos de distinguir un documento de otros (ej. un término que aparece en la mayoría de los documentos contribuye menos a la puntuación, que otro que aparece solo en un conjunto pequeño de documentos), el número de ocurrencias de los términos buscados en el documento, o la proximidad entre los mismos. La métrica utilizada no está documentada en Semantic Search. En general la respuesta buscada, si existe, aparece en los primeros documentos del ranking, y en los primeros experimentos con nuestro corpus para preguntas”Quién” el valor d=15 es más que suficiente.

A continuación se procesa esta colección de documentos para extraer hasta un máximo de r respuestas candidatas (con valor r=5 adecuado para nuestro corpus).

Cada documento se divide previamente en

fragmentos. Un fragmento en nuestro corpus es la secuencia de caracteres más larga entre caracteres punto o punto y coma. Para esto utilizamos la herramienta OpenNLP Sentence Detector que detecta oraciones o fragmentos de textos en inglés, adaptada a nuestro caso. Dadas las características de nuestro corpus, en que en general las oraciones son largas y a menudo con varias oraciones subordinadas, los fragmentos suelen ser largos.

En cada documento se ubica una anotación del tipo de respuesta buscada, y alrededor de ésta se va a centrar el análisis. Se busca una ventana de c que incluya dicha anotación y además las anotaciones y palabras claves que componen la consulta, y se devuelve el fragmento o fragmentos de texto que incluyen todos estos elementos. En los experimentos un valor de c=150 obtiene resultados adecuados.

En nuestro corpus es frecuente que aparezca la misma respuesta más de una vez en el documento (por ejemplo como parte de los considerandos de la resolución y como parte de las decisiones resueltas). Por ello de cada documento solo se devuelve una respuesta.

La anotación del tipo buscado en el contexto del fragmento que la incluye constituye una respuesta. En ese fragmento la anotación respuesta aparece resaltada en un color, y los términos de la consulta aparecen en negrita. En los documentos de nuestro corpus los fragmentos suelen ser largos, y el frag-

Figura 4. Resultados de la consulta del ejemplo 1.

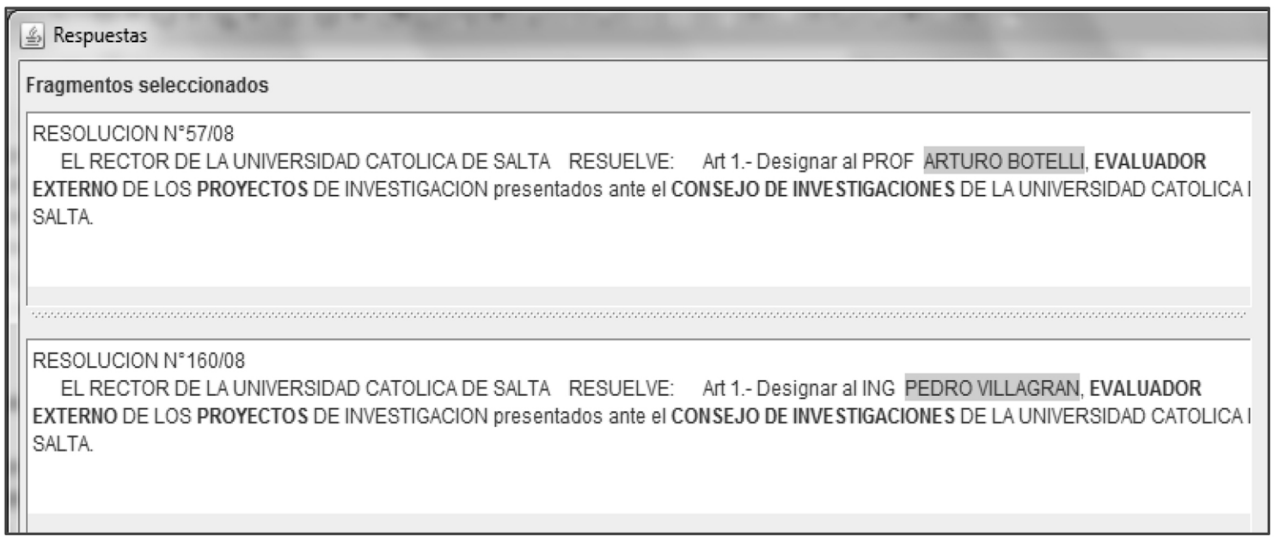

Figura 5. Resultados de la consulta del ejemplo 2.

mento en que aparece la anotación que sirve de respuesta provee suficiente contexto para dar confianza al usuario y que éste decida si la respuesta obtenida es adecuada para la consulta realizada. La sección siguiente justifica la elección de fragmentos de esta manera como resultado a mostrar al usuario.

Finalmente se devuelven las primeras r respuestas obtenidas (las anotaciones en su contexto), que corresponden a r documentos diferentes. El orden de las respuestas es el mismo en que aparecieron sus respectivos documentos. Por cada respuesta se devuelve el fragmento en que está incluida.

Las Figuras 4 y 5 muestran los resultados de las consultas de las Tablas 2 y 3 respectivamente.

CONSIDERACIONES SOBRE LA PRESENTACIÓN DE RESPUESTAS

Lin et al (Lin, y otros, 2003) exploraron los diversos tipos de interfaces para los sistemas de BR, partiendo del paradigma foco-más-contexto. Dado que la mayoría de los sistemas extraen las respuestas de documentos de texto, el texto que rodea a la respuesta sirve como proveedor natural de contexto. Los autores realizaron un experimento para decidir cuánto contexto debería devolver el sistema, en particular:

- La respuesta exacta.
- Respuesta-en-oración: la respuesta exacta con la frase de la que se extrajo.
- Respuesta-en-párrafo: la respuesta exacta con el párrafo del que se extrajo, en el que se resalta la frase que contiene la respuesta.
- Respuesta-en-documento: la respuesta exacta con el documento completo del que se extrajo, en el que se resalta la frase que contiene la respuesta.

El estudio mostró que los usuarios prefirieron la opción respuesta-en-párrafo, mientras que la opción respuesta exacta fue la menos preferida. En particular, los pronombres causaron problemas, especialmente en la opción respuesta-en-oración al tomar las frases con pronombres fuera de contexto. Además se evaluó la percepción de confiabilidad de las respuestas: los participantes necesitaron en promedio al menos un párrafo para formarse un juicio en cuanto a la confiabilidad de la respuesta.

EVALUACIÓN Y TRABAJO FUTURO

Dado que el sistema de BR está en etapa preliminar no tenemos un mecanismo formal de evaluación. No obstante en nuestros experimentso utilizamos como punto de referencia para la evaluación el buscador semántico descrito en la Sección 3, que consta de una interfaz tipo formulario para realizar consultas en el lenguaje XML Fragments y devuelve los documentos relevantes. Para evaluar la construcción de las consultas a partir de las preguntas en lenguaje natural se comparan los documentos obtenidos por el sistema BR con los del buscador semántico. Para evaluar si las respuestas obtenidas (la respuesta resaltada dentro de su contexto) son adecuadas, realizamos inspección ocular para verificar que las respuestas obtenidas son correctas en base a los contenidos del corpus. Los resultados preliminares, para preguntas de tipo “Quién” son satisfactorios. A partir de estos resultados y la arquitectura básica descripta se están extendiendo los tipos de preguntas que se pueden contestar, y las características de las respuestas (por ejemplo, en el caso en que la respuesta no sea solo una entidad sino un conjunto de ellas).

CONCLUSIONES

Los sistemas de búsqueda de respuestas son una de las áreas de investigación más activas en la minería de textos. En este trabajo hemos presentado una arquitectura preliminar para un sistema de búsqueda de respuestas en un corpus de más

de 8000 documentos que contienen resoluciones rectorales, que por el momento responde solamente a preguntas factoides sencillas, pero que está sirviendo de plataforma para atacar tanto preguntas más complejas como tipos de respuestas también más complejos.

AGRADECIMIENTOS

Este trabajo ha sido financiado en parte por el Consejo de Investigaciones de la Universidad Católica de Salta (Resol Rect 839/13).

REFERENCIAS

Amaral, C., Laurent, D., Martins, A., Mendes, A., and Pinto, C., "Design and Implementation of a Semantic Search Engine for Portuguese", Proceedings of 4th International Conference on Language Resources and Evaluation (LREC 2004), (2004).

Apache UIMA Development Community, "UIMA Tutorial and Developers' Guides, version 2.6.0", (May de 2014). Obtenido de https://uima.apache.org/d/uimaj-2.6.0/index.html

Chu-Carroll, J., Prager, J., Czuba, K., Ferrucci, D., and Duboue, P., "Semantic Search via XML Fragments: a High-Precision Approach to IR", Proceedings of the 29th Annual international ACM SIGIR Conference on Research and Development in information Retrieval. New York, (2006).

Ferrucci, D., and Lally, A., "Building an example application with the Unstructured Information Management Architecture", IBM Systems Journal; 45, 3, (2004).

García Cumbreras, M. Á., "BRUJA: Un Sistema de Búsqueda de Respuestas Multilingüe", Colección de Monografías de la Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN) 9, (2010).

Ingersoll, G., Morton, T., and Farris, A., "Taming Text", Shelter Island: Manning, (2013).

Lehnert, W., "Human and computational question-answering", Cognitive Science; 1, 47-73, (1977).

Lin, J., Quan, D., Sinha, V., Bakshi, K., Huunh, D., Katz, B., and Karger, D., "The Role of Context in Question Answering Systems", CHI '03: Extended Abstracts on Human Factors in Computing Systems, Fort Lauderdale, Florida, 1006-1007,. (2003).

Montes-y-Gómez, M., Villaseñor-Pineda, L., and López-López, A., "Mexican Experience in Spanish Question Answering", Computación y Sistemas; 12, 1, (2008).

OMG, "XML Metadata Interchange (XMI), v 2.1.1", (2007).

Pasca, M., "Lightweight Web-Based Fact Repositories for Textual Question Answering", CIKM'07, Lisboa, (2007).

Pérez, A., y Cardoso, A. C., "Categorización automática de documentos", Simposio Argentino de Inteligencia Artificial, 40 JAIIO, Córdoba, (2011).

Pérez, A., y Cardoso, A. C., "Extracción de entidades con nombre", Simposio Argentino de Inteligencia Artificial, 42 JAIIO, Córdoba. (2013).

Webber, B., and Webb, N., "Question Answering", en A. Clark, C. Fox, and S. Lappin (Edits.), The Handbook of Computational Linguistics and Natural Language Processing, Wiley-Blackwell, 630- 654, (2010).

Fecha de Presentación: 15/05/15
Fecha de Aprobación: 08/06/15