

WebQA: Respuesta Automática a Preguntas

Daniel Castelo, Jorge Isi, Sebastián Martínez, Diego Garat, and Guillermo Moncecchi

Instituto de Computación, Facultad de Ingeniería, Universidad de la República,
Uruguay

Resumen En este trabajo se presenta WebQA, un sistema de *Búsqueda de Respuestas* para el español, orientado a responder preguntas que esperan una respuesta precisa y concreta, tales como las comenzadas en «*Cuándo*», o «*Quién*». WebQA utiliza como repositorio de información a la World Wide Web y asume como hipótesis que la información que allí existe se encuentra en forma redundante: es posible encontrar varias veces la respuesta a una misma pregunta, aunque con diferentes formulaciones. Para responder, WebQA clasifica la pregunta, realiza distintas consultas, y genera un ranking de respuestas combinando los documentos encontrados en la Web. Los resultados son promisorios: el rendimiento es superior al de varios sistemas de respuestas similares con los que se lo comparó. Además, como su funcionamiento se basa en *patrones*, es posible extenderlo para ampliar el número de preguntas a las que responde correctamente.

1. Introducción

La Web es el mayor repositorio de información existente y está en continuo crecimiento. En este escenario la necesidad de sistemas eficientes y precisos de búsqueda de información es cada vez mayor. Los sistemas más utilizados actualmente son los *motores de búsqueda*, como *Google* o *Yahoo*, sistemas de *Recuperación de Información*. Su funcionamiento se basa en el ingreso, por parte del usuario, de un conjunto de palabras clave para recuperar documentos de la Web que las contengan, utilizando diferentes criterios para destacar los resultados más relevantes entre los obtenidos. En este tipo de sistemas, es el usuario quien debe expresar su necesidad de información en forma de palabras clave y luego buscar entre los resultados presentados por el buscador la información deseada.

Esta necesidad de información podría expresarse más naturalmente para el usuario utilizando preguntas formuladas en lenguaje natural. Los sistemas de Búsqueda de Respuestas son herramientas capaces de obtener respuestas concretas a necesidades de información muy precisas a partir de una pregunta y un análisis de documentos escritos en lenguaje natural[1]. Tratan de resolver el doble problema de aceptar este tipo de preguntas, para identificar, localizar, extraer y presentar al usuario única y exclusivamente aquella información que desea conocer.

Cuando se realiza la búsqueda, es importante considerar la fuente de información a utilizar: si estamos buscando en un corpus de documentos, la respuesta estará probablemente en un documento específico del conjunto, y será necesario realizar un análisis profundo para obtenerla. Si la búsqueda se hace en la Web, alguna de las premisas cambian: es mucho más grande que cualquier otro corpus, por lo que, dada una pregunta, es muy probable que se encuentre allí respondida, incluso varias veces, con diferentes variantes. Por tanto, existe en este caso un elemento sumamente importante que podría mejorar la búsqueda: la alta redundancia en la información.

Las investigaciones de los sistemas de búsqueda de respuestas se vienen desarrollando rápidamente, principalmente debido a la creciente demanda y la organización de conferencias para su evaluación, como son las TREC[11] y las CLEF[3]. Sin embargo, estas investigaciones son básicamente para el inglés, y es poco lo que se ha hecho en español. Además gran parte de los sistemas trabajan sobre corpus acotados de documentos y no sobre la Web. Dentro de los sistemas en inglés podría destacarse a WebClopedia[8], AnswerBus[12] y AskMSR[2], mientras que en español podría mencionarse el sistema de la Universidad de Alicante [7] y los sistemas del INAOE de México de Alejandro del Castillo [5] y de M. Perez Coutiño[4], entre muchos otros.

Este trabajo profundiza lo que se ha hecho para el español en el área, en base a adaptar y mejorar técnicas existentes para otros idiomas.

El resultado de la investigación concluye en el desarrollo de WebQA, un sistema de búsqueda de respuestas en español que explota la Web como fuente de información. Dada una pregunta en lenguaje natural, el sistema la analiza y retorna de la Web una lista de fragmentos relevantes. A su vez, siempre que la pregunta requiera una respuesta concreta, el sistema la identifica y la resalta en los fragmentos retornados. El sistema tiene la característica de tener cierto grado de escalabilidad, permitiendo agregar información sobre nuevos tipos de preguntas detectados.

2. El Sistema WebQA

WebQA es un sistema de búsqueda de respuestas en el cual tanto las preguntas como las respuestas están expresadas en español. La fuente de información de WebQA es la Web, y el objetivo es utilizarla asumiendo que: (a) la respuesta a la pregunta formulada estará presente en la Web; y (b) la respuesta aparecerá varias veces, posiblemente formulada de distintas formas. La redundancia de este corpus dio resultados previos aceptables sin necesidad de utilizar recursos léxicos importantes.

El tipo de preguntas que se optó por responder son las llamadas *fácticas*, aquellas que requieren una única respuesta correcta y concreta y además tienen un pronombre interrogativo claro («cuándo», «dónde», «quién»). Por ejemplo, en la pregunta «¿En qué ciudad está la Torre Eiffel?», la respuesta esperada es «París». Otros tipos de pregunta no son soportados (al menos explícitamente); por ejemplo, las preguntas causales o de recopilación de información.

WebQA cumple con las etapas clásicas de este tipo de sistemas[7]: análisis de la pregunta, recuperación de documentos, selección de pasajes relevantes, extracción y presentación de la respuesta. Para obtener documentos de la Web se recurre a motores de búsqueda utilizando reformulaciones de la pregunta, se analizan los fragmentos retornados por el buscador (*snippets*) y se ponderan según la probabilidad de contener la respuesta. A partir de este análisis, se aplican técnicas para extraer la respuesta concreta a la pregunta. Al usuario se le presentan como posibles respuestas la lista de fragmentos de texto ordenados, con la respuesta concreta marcada. En las secciones siguientes se describen más en detalle cada una de estas etapas.

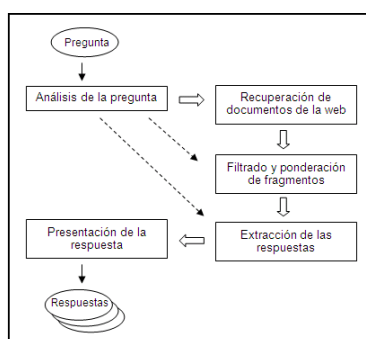


Figura 1. Etapas de WebQA.

2.1. Análisis de la Pregunta

En esta etapa inicial se busca obtener información importante de la pregunta que formula el usuario (como por ejemplo el tipo de respuesta esperado) que luego será utilizada en las etapas sucesivas con el objetivo final de retornar una respuesta correcta. Esto es, dada una pregunta, se determina de qué tipo es (cuándo, dónde, quién, etc.) y qué se pretende como respuesta (fecha, lugar, nombre propio). Se utiliza un etiquetador sintáctico (Freeling 1.4 [6]) para encontrar la categoría léxica de cada palabra de la pregunta. Esta información sirve para luego acotar los fragmentos a analizar y ubicar la respuesta exacta.

Es necesario también identificar las palabras vacías —artículos, conjunciones— que no serán de utilidad para las búsquedas por no aportar significado y ser demasiado frecuentes.

Identificar el tipo de pregunta y el tipo de respuesta requerido no es un problema trivial: para el español existen muchísimas maneras de formular cada tipo de pregunta. Si se quiere saber dónde está el Cabo Polonio (un pequeño balneario en la costa de Rocha, en Uruguay), posibles formulaciones válidas para consultar son entre otras: «¿Dónde está ubicado Cabo Polonio?», «¿En qué

lugar está ubicado Cabo Polonio?», «¿En qué sitio está ubicado Cabo Polonio?» o «¿Cuál es el lugar dónde está ubicado Cabo Polonio?»

La técnica utilizada en WebQA para contemplar esta situación consiste en establecer *patrones* que permiten determinar el tipo de la pregunta y de la respuesta esperada [9,8]. Cada patrón se compone de una secuencia de palabras y etiquetas gramaticales, conectadas por operadores sencillos para disyunciones.

Dado un tipo de pregunta, no siempre se puede inferir directamente cuál es el tipo de respuesta esperado. Por ejemplo, para las preguntas que empiezan con «*Cuándo*», directamente se desprende que el tipo de respuesta esperado es una **fecha**; por el contrario, las preguntas que empiezan con «*Quién*» pueden esperar como respuesta el nombre de una persona («¿*Quién fue el primer hombre en pisar la Luna?*») o su descripción («¿*Quién fue Neil Armstrong?*»).

En WebQA hay actualmente definidos veinticinco patrones, de los cuales diez corresponden a las preguntas del tipo **cuándo**, cinco del tipo **dónde**, cinco para el tipo **cuántos**, tres del tipo **quién**, y por último dos que no están incluidas en ninguno de estos casos, en los que se espera un nombre común como respuesta («¿*Qué deporte predomina en Uruguay?*»). Esta lista de patrones es extensible, de modo de soportar otras formulaciones. Para el caso de que no coincida la pregunta con los patrones existentes se tiene una categoría especial «*Indefinida*» para la cual se presentan los fragmentos recuperados ordenados por su probabilidad de contener la respuesta, pero no se realiza la etapa de extracción. En la Tabla 1 se muestran algunos ejemplos.

Cuadro 1. Ejemplo de formulación de patrones.

Formulación	Tipo pregunta	Tipo respuesta
Dónde («¿ <i>Dónde está ubicado Cabo Polonio?</i> »)	Dónde	Lugar
Quién (es fue era) <Nombre Propio> («¿ <i>Quién es Bill Gates?</i> »)	Quién	Descripción
Quién («¿ <i>Quién descubrió América?</i> »)	Quién	Persona

2.2. Recuperación de Documentos de la Web

En esta etapa se intenta obtener de la Web una lista de documentos que podrían contener la respuesta buscada. Basándose exclusivamente en la manipulación de las palabras de la pregunta —y sin utilizar ningún otro recurso léxico, como realizan otros sistemas de este tipo [8]— se construyen *reformulaciones* que son enviadas como consultas a un motor de búsqueda. Existen diferentes maneras de reformular una pregunta, pero hay reformulaciones que parecen *a*

priori mejores que otras, en lo que tiene que ver con la expectativa de obtener un documento con la respuesta exacta a la pregunta; sin embargo, cuanto mejor es en ese sentido la reformulación, menor cantidad de documentos se obtienen como respuesta.

El sistema webQA aplica tres técnicas para formular las consultas, las que se detallan a continuación:

Reformulador según Patrones Preestablecidos. Dada una pregunta, en el momento de definir cuáles documentos son interesantes para recuperar de la Web, por tener altas posibilidades de contener la respuesta, algunas interrogantes planteadas fueron: ¿Cómo debería estar formulada la respuesta? ¿Qué relación existe entre la manera en que está formulada la pregunta y la manera en que se formula la respuesta? ¿Cómo se puede aprovechar esto?

Analizando distintos ejemplos de preguntas y respuestas, se llegó a la conclusión de que la formulación de la respuesta se puede lograr a partir de la forma declarativa de la pregunta, combinándola con palabras que refieren a lo que se espera como respuesta —por ejemplo, «*año*», «*día*», «*país*», etc.—. Esto permite definir una serie de patrones asociados a un conjunto de reformulaciones. Estos patrones, junto a los que se utilizan para determinar el tipo de pregunta (2.1), permiten la escalabilidad del sistema frente a nuevos tipos de pregunta o dominios de información.

Para determinar los patrones se identifican tres partes de la pregunta: el verbo —a veces junto con un verbo auxiliar—, su contexto izquierdo y su contexto derecho. Para la pregunta que hemos utilizado como ejemplo, los patrones se construyen a partir de **departamento**, **está ubicado**, y **Cabo Polonio**. Para saber si un patrón es aplicable a una pregunta, se comparan sus palabras iniciales (en el ejemplo EN QUE). En la Tabla 2 se muestran ejemplos de reformulaciones.

Cuadro 2. Ejemplos de patrones para reformulación de la pregunta, donde: X = contexto izquierdo al verbo de la pregunta; V = verbo de la pregunta; Y = contexto derecho al verbo de la pregunta.

Patrón	Ejemplo
EN QUE	«¿En qué departamento está ubicado Cabo Polonio?»
X + Y V	departamento + Cabo Polonio está ubicado
X + V Y	departamento + está ubicado Cabo Polonio
X + V + Y	departamento + está ubicado + Cabo Polonio
CUANTOS	¿Cuántos días tiene un año bisiesto?
X V Y	días tiene un año bisiesto
X + Y V	días + un año bisiesto tiene
X + V Y	días+ tiene un año bisiesto
X + V + Y	días + tiene + un año bisiesto

Reformulador en base a Permutación de Dos Primeras Palabras.

Esta técnica está basada en el trabajo de Luis Villaseñor[10]. Consiste en tomar la pregunta, quitando el pronombre interrogativo o palabras iniciales que determinan su tipo (quién, dónde, en qué año, a qué hora, etc.), y realizarle ciertas permutaciones predefinidas a sus dos primeras palabras. Estas dos palabras generalmente se componen del verbo, y un pronombre o un verbo auxiliar, de modo que manipulándolas se escribe la pregunta en sus distintas formas declarativas. Las reformulaciones realizadas sobre las palabras restantes son: utilizar todas las palabras, quitar la primera palabra, quitar la primera y segunda palabra, mover la primera palabra al final, mover la primera y segunda palabra al final.

Volviendo al ejemplo, para la pregunta «¿Dónde está ubicado Cabo Polonio?», quitando el pronombre interrogativo queda «está ubicado Cabo Polonio», por lo tanto las dos palabras a manipular son el verbo auxiliar **está** y el verbo principal **ubicado**. Las reformulaciones obtenidas son: “**está ubicado** Cabo Polonio”, “**ubicado** Cabo Polonio”, “Cabo Polonio”, “**ubicado** Cabo Polonio **está**” y “Cabo Polonio **está ubicado**”.

Reformulador en base a Conjunción de Palabras Clave.

Por la forma en que están generados, si alguno de los reformuladores anteriores retorna un documento, es de esperar que tal documento contenga la respuesta correcta. Pero para ciertas preguntas, en especial las largas, se obtienen pocos o ningún resultado. En estos casos, se utiliza además un reformulador que seguramente traiga documentos, aunque con menos seguridad de encontrar en ellos la respuesta correcta.

El reformulador genera dos consultas, que se nombrarán como máxima y mínima. La reformulación máxima se compone de la conjunción de todas las palabras de la pregunta, sin tomar en cuenta las muy frecuentes. La reformulación mínima se compone únicamente del verbo, nombres propios y fechas que ocurran en la pregunta.

2.3. Filtrado y Ponderación de Fragmentos

Una vez obtenidos los documentos de la Web, es necesario filtrarlos, y medir su probabilidad de contener la respuesta correcta. Como el análisis de los documentos completo implica un alto tiempo de procesamiento, se optó por analizar solamente los fragmentos retornados por el motor de búsqueda.

Primero se filtran los fragmentos, descartando a aquellos que no tiene ningún *token* del tipo de respuesta esperado. Luego, para ponderar al resto de los fragmentos, se trabaja sobre la hipótesis de que cuantos mas tokens de la pregunta contenga, mayor es la probabilidad de que contenga la respuesta. Además se analiza cuán cerca están estos tokens entre ellos, cuánto del token con el tipo de respuesta esperado y cuál fue la reformulación por la que se obtuvo el fragmento. El algoritmo implementado está basado en el trabajo de José Luis Vicedo González [7]. Se utiliza una ventana de análisis de tamaño fijo que se desplaza por los tokens del fragmento. La *mejor ventana* es aquella que contenga más tokens de la pregunta, siendo al menos uno del tipo esperado por la respuesta. En la Tabla 3 se muestra la fórmula para el cálculo del puntaje de los fragmentos.

Cuadro 3. Fórmula para el cálculo del puntaje del fragmento

Sea:

- P = Puntaje final del fragmento
- F = Cantidad de tokens clave de la pregunta en el fragmento
- V = Cantidad de tokens clave de la pregunta en la mejor ventana
- C_1 y C_2 = Constantes. En las pruebas realizadas, $C_1=1$ y $C_2=0.5$
- R = Puntaje de la reformulación de la pregunta con la cual se obtuvo el fragmento.

La fórmula es: $P = (C_1 \times V + C_2 \times (F - V)) \times R$

2.4. Extracción de la Respuesta Mediante Análisis de Ngramas

Una vez determinados los documentos más relevantes, es necesario buscar dentro de ellos la respuesta concreta a la pregunta. Para esto, se asume que tal respuesta se repetirá en la mayoría de los fragmentos recuperados: no se busca sólo en el primer documento, sino que se intenta sumar la información aportada por cada uno de los que aparecieron en los primeros lugares. La técnica implementada, una variante del análisis de n-gramas de AskMSR[2], se divide en tres etapas:

Recolección y Ponderación de Ngramas. En esta primera instancia, se analizan los fragmentos de modo de recolectar y ponderar todos los unigramas, bigramas y trigramas candidatos a ser respuesta a la pregunta. La suposición fundamental es que, de todos los tokens con el tipo esperado, el que sea la respuesta a la pregunta se repite más veces. A su vez, para la ponderación, se tienen en cuenta otros factores, como por ejemplo si el token está dentro de la ventana del fragmento y la ponderación del fragmento del cual fue obtenido.

Solapamiento de NGramas. La intención aquí es solapar el primer ngrama (el de mayor puntaje), el cual ya es la respuesta a la pregunta, con los que le siguen en puntaje, de forma de generar respuestas más largas y más completas. Por ejemplo si el primer ngrama es A, y a continuación le siguen en puntaje los ngramas AB y BC, la respuesta a la pregunta será ABC. No se solaparán todos los ngramas obtenidos, sino solamente los de mayor puntaje.

Reponderación de Fragmentos según Ngrama Respuesta. Luego del análisis de n-gramas se vuelve a realizar otro ranking de los fragmentos en base a si contienen (total o parcialmente) la respuesta obtenida. Con esto, los fragmentos que contengan el n-grama completo quedarán en primer lugar.

2.5. Presentación de la Respuesta

Como etapa final del proceso, se presentan al usuario, como posibles respuestas los fragmentos resultantes ponderados y ordenados, destacando en cada uno de ellos la respuesta “exacta” (el ngrama de mayor puntaje.)

3. Resultados Obtenidos

Las medidas de evaluación utilizadas son la precisión (proporción de respuestas correctas del total de realizadas) de la respuesta extraída (marcada en los fragmentos) y la precisión y el Mean Reciprocal Rank (MRR)[5] de las respuestas obtenidas según el ranking de fragmentos. Para el cálculo de la precisión, en el primer caso se considera a una respuesta como correcta si está marcada correctamente en el fragmento, y en el segundo si se encuentra en los fragmentos presentados al usuario.

El MRR¹ califica a cada pregunta asignándole el inverso de la posición del primer fragmento retornado por el sistema que contiene la respuesta esperada. La importancia del MRR es que pondera, además de la aparición de la respuesta correcta, la posición en el ranking en la que aparece.

Con el propósito de analizar el rendimiento para cada tipo de pregunta se utiliza un conjunto de 150 preguntas creado con antelación al desarrollo del sistema y evaluado manualmente. En la Tabla 4 puede verse que las preguntas que se responden mejor son las de tipo Cuándo, Dónde y Quién y los peores resultados se dan en la pregunta de tipo Cuánto. El principal problema es que el principio de redundancia es más difícil de aplicar en estos casos: las cantidades están expresadas en distintas unidades (metros, pies, millas) y además muchas veces no existen datos exactos sobre determinados fenómenos. Por lo tanto el sistema no puede alimentarse en el análisis de n-gramas de la respuesta que más se repite en los fragmentos recuperados.

Cuadro 4. Rendimiento del sistema por tipo de pregunta.

Tipo	MRR	Prec. Frag.	Prec. Ext. Respuesta
Cuál	0,4487	0,6923	0,4615
Cuándo	0,5833	0,7333	0,6896
Cuántos	0,4902	0,5294	0,2220
Dónde	0,6595	0,7429	0,6563
Indefinido	0,3125	0,3750	0,1666
Quién	0,6383	0,7568	0,6857
Total	0,5776	0,6929	0,5906

¹ $MMR = \frac{\sum_{i=1}^n \frac{1}{pos(i)}}{n}$ donde n corresponde al número de preguntas de prueba y pos indica la posición de la primera respuesta correcta para la pregunta i

3.1. Comparación con Otros Sistemas

Las principales dificultades para comparar a WebQA con otros sistemas son que la mayoría están contruidos para el idioma inglés y que sólo un grupo reducido utiliza la Web como repositorio de información. Cualquier comparación con un sistema que no tenga estas características tiene un factor de incertidumbre importante, ya sea porque debe traducirse la pregunta de un idioma a otro o ignorarse la diferencia de que los repositorios de información sean distintos.

Entre otras evaluaciones, se comparó a WebQA con el sistema de Alejandro del Castillo [5], que es el que mas se asemeja a WebQA ya que es para el idioma español y utiliza la Web como repositorio de información, y con los resultados de dos sistemas que el mismo autor cita en su trabajo de maestría del año 2005 (uno del INAOE de Méjico y otro de la Universidad de Alicante), que utilizan el corpus de documentos proporcionado por la CLEF. Para la evaluación se utilizó el conjunto de preguntas de las CLEF 2003 para el idioma español, en base a que no se encontraron evaluaciones mas recientes para estos sistemas.

Cuadro 5. Resultados comparativos de sistemas BR en Español.

Sistema	MRR	Precisión
WebQA	0,4845	0,5442
INAOE (Pérez)	0,3958	0,4250
INAOE (Del Castillo)	0,3580	0,4150
Alicante	0,3075	0,4000

En esta comparación WebQA presenta un rendimiento significativamente superior con un MRR-Precisión de (0,484, 0,544) contra los valores (0,396,0,425) que obtiene el sistema siguiente y (0,358, 0,415) que obtiene el sistema de Alejandro del Castillo. La mayor *performance* de WebQA en la comparación puede deberse a que los buscadores mejoraron significativamente su rendimiento entre el año 2007 y 2005, y a que la Web tiene mayor información y por ende mayor redundancia, que es uno de los pilares en los que se basan ambos sistemas.

4. Conclusiones

Este trabajo mostró que se pueden obtener muy buenos resultados explotando la Web. Para el 70 % del conjunto de preguntas de prueba utilizado, la respuesta correcta se encontró dentro de los tres primeros fragmentos presentados, un resultado mejor que los sistemas con los cuales se comparo WebQA. Esto hace pensar que la hipótesis de que la alta redundancia de información en la Web permite salvar el inconveniente de su baja confiabilidad como fuente de información, a la vez que hace viable utilizar recursos “simples” (sin apelar a análisis profundo de los textos) para obtener buenos resultados.

WebQA es un sistema que responde bien a preguntas cuya respuesta es fáctica y precisa, por lo que no se lo puede considerar un sistema de BR “genérico”. Dentro de los tipos de pregunta que el sistema tiene mejor rendimiento se encuentran las que empiezan con los pronombres **Cuándo**, **Dónde** y **Quién**. Las preguntas de tipo **Cuál** son más difíciles de responder para WebQA porque tienen más reformulaciones posibles y, sobre todo, distintos tipos de respuesta esperada. La complejidad en este tipo de formulaciones podría abordarse en la etapa de análisis de la pregunta, agregando tantas reformulaciones como sean necesarias en los patrones. Para las de tipo de pregunta **Cuántos** se tiene el problema de que no siempre se tienen cifras exactas, o unidades iguales. Esto hace que el análisis de ngramas no sea útil en estos casos.

Muchas de las respuestas incorrectas dadas por WebQA se deben a que falla el supuesto de la redundancia de información (preguntas muy específicas, cuya respuesta no aparece, o aparece muy poco en la Web). La posibilidad de analizar el documento entero o realizar análisis más profundos, podrían, en nuestra opinión, mejorar el rendimiento.

Referencias

1. Sistemas de question answering, Abril 2007.
<http://question-answering.galeon.com/Question-Answer.html>.
2. Eric Brill, Susan Dumais, and Michele Banko. An analysis of the askmsr question-answering system. In *In Proceedings of the ACL-02 conference on Empirical methods in natural language processing*, Morristown, NJ, USA, June 2002.
3. CLEF. Cross language evaluation forum (clef), Abril 2007.
<http://www.clef-campaign.org/>.
4. M. Perez Coutiño. The use of lexical context in question and answering for spanish. In *Text REtrieval Conference*, 2004.
5. Alejandro Del Castillo Escobedo. Búsqueda de respuestas mediante redundancia en la web. Master’s thesis, INAOE: Instituto Nacional de Astrofísica, Óptica y Electrónica, Puebla, Mexico, Febrero 2005.
6. Freeling. Freeling, Abril 2007.
<http://garraf.epsevg.upc.es/freeling>.
7. José Luis Vicedo González. Recuperación de información de alta precisión: Los sistemas de búsqueda de respuestas. Alicante, España, Julio 2003.
8. E. Hovy, L. Gerber, U. Hermjakob, M. Junk, and C. Lin. Question answering in webclopedia. In *Proceedings of The Ninth Text REtrieval Conference (TREC 9)*, Maryland, Estados Unidos, Noviembre 2000.
9. L. Kosseim, L. Plamondon, and L.J. Guillemette. Answer formulation for question-answering. In *Proceedings of The Sixteenth Canadian Conference on Artificial Intelligence (AI’2003)*, Halifax, Canada, Junio 2003.
10. Luis Villaseñor Pineda, Manuel Montes y Gómez, and Alejandro del Castillo. Búsqueda de respuestas basada en redundancia: un estudio para el español y el portugués. Puebla, Mexico, Noviembre 2004.
11. TREC. Text retrieval conference (trec), Abril 2007.
<http://trec.nist.gov/>.
12. Zhiping Zheng. Answerbus question answering system. In *Proceeding of HLT Human Language Technology Conference (HLT 2002)*, San Diego, Estados Unidos, Marzo 2002.