Recuperación de fragmentos útiles de texto a partir de páginas web. Resultados experimentales y perspectivas futuras

Carlos G. Figuerola, José Luis Alonso Berrocal, y Angel F. Zazo Rodríguez

Grupo de Investigación REINA, Universidad de Salamanca
c/ Francisco de Vitoria, 6-16, 37008 Salamanca
{figue, berrocal, afzazo}@usal.es

Resumen. Todos aceptamos que el web es el mayor repositorio de información disponible, y por ello son especialmente importantes las herramientas y las técnicas que permiten acceder a dicha información. Uno de los problemas con los que debemos enfrentarnos es la diversidad de usos posibles; esta diversidad de usos requiere enfoques especializados. En este trabajo se describe uno de los usos de la información que se encuentra en el web, los problemas que plantea la obtención de información útil para esos usos, así como los enfoques adoptados en diversos experimentos realizados tedentes a evaluar las posibilidades de diversas técnicas aplicables.

Palabras clave: recuperación de información, recuperación web, fragmentación de texto, conversión a texto plano

1 Introducción

Todos aceptamos que el web es el mayor repositorio de información disponible, y por ello son especialmente importantes las herramientas y las técnicas que permiten acceder a dicha información. Buscadores de diverso tipo, directorios manuales o automáticos, tecnologías basadas en agentes, *web mining*, son algunas de las técnicas que intentan ayudarnos en la utilización de la información disponible, de una u otra forma, en el web. Uno de los problemas con los que debemos enfrentarnos es la diversidad de usos posibles; esta diversidad de usos requiere enfoques especializados, toda vez que parece que las soluciones genéricas resultan en muchas ocasiones poco satisfactorias.

Desde esta perspectiva, una experiencia interesante es la auspiciada por el Cross Lingual European Forum (CLEF) [1], en el sentido de modelar un determinado uso de información que se encuentra en el web y plantear experimentos tendentes a medir los resultados de aplicar diferentes técnicas. Una de las ventajas de la metodología empleada por CLEF es que el entorno en el que se han de desarrollar los experimentos es homogéneo para todos esos experimentos, y las técnicas de medición y evaluación de resultados están bastante contrastadas.

En concreto, uno de los escenarios planteados es el de un supuesto usuario que requiere información sobre un tema (*topic*) determinado, sobre el cual debe escribir un trabajo o artículo más o menos académico. El usuario no requiere páginas web (o documentos PDF, etc.) sino información útil para sus fines; esta información útil toma la forma de bloques o fragmentos de texto directamente utilizables en la redacción del artículo, en el mejor de los casos susceptibles de ser copiados y pegados en el procesador de texto. En el modelo planteado, el usuario dispone de los resultados de unas cuantas búsquedas en un buscador estándar (Google), así como de unos cuantos documentos considerados como *fuentes conocidas* sobre el tema en cuestión [2]. El escenario completo consta de varios temas o *topics* (alrededor de 60, en la actualidad) en varias lenguas.

El planteamiento tiene el interés adicional de que se trata de una simulación muy cercana a la realidad, en el sentido de que es preciso resolver no sólo la implementación de modelos teóricos, sino la resolución de todos los problemas que aparecen en una situación real: páginas HTML mal formadas, etiquetas maliciosas (*web spam*) [3], multiplicidad de lenguas, errores tipográficos, conversión a texto, detección de lenguas, etc. Muchos de estos aspectos, aunque no tienen el impacto conceptual ni teórico de otros, están lejos de estar resueltos adecuadamente y no son en absoluto triviales, en el sentido de que influyen de forma notable en los resultados que se obtienen en situaciones reales.

Este trabajo trata de mostrar algunos de los enfoques aplicados en la resolución de una tarea como la descrita, de sus limitaciones y de las posibles vías de superar tales limitaciones.

2 Preparación de la colección documental

2.1 Planteamiento general

Nuestra aproximación consiste en considerar, para cada *topic*, el conjunto de documentos recuperados por Google como la colección de documentos con la que trabajar. Puesto que la tarea exige obtener bloques o fragmentos de texto, estos documentos han de fragmentarse o descomponerse en trozos, cada uno de los cuales será considerado como un documento independiente.

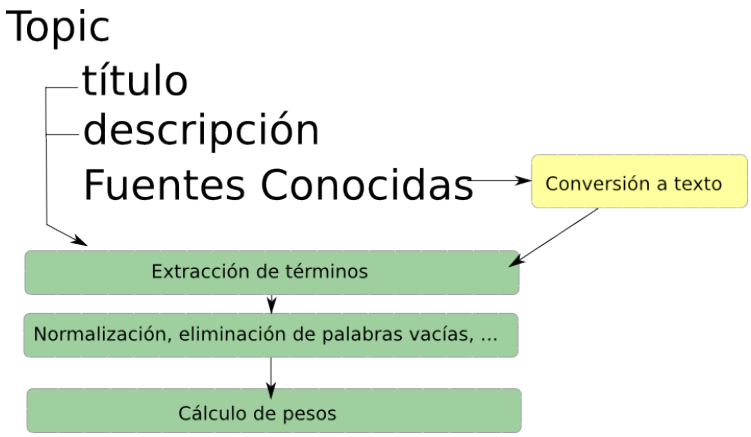

Fig. 1. De *topic* a consulta

Como consulta, para cada *topic*, podemos utilizar la descripción que tenemos para cada uno de ellos. Adicionalmente, dicha consulta puede ser enriquecida con más términos, provenientes de las *fuentes conocidas* para ese *topic*. También podemos usar los anclas disponibles que apuntan a los documentos recuperados por Google.

De esta forma, la tarea puede abordarse como un problema clásico de recuperación, y aplicar en consecuencia, técnicas convencionales.

Así, la colección estará formada por los *snippets* provenientes de los documentos recuperados por Google para cada *topic*. Para cada uno de estos *topics*, se han efectuado una o más búsquedas en Google, y se han tomado para cada una de esas búsquedas los n primeros documentos recuperados. Esto implica un número variable de documentos por *topic*.

Hemos considerado o valorado igual todas las búsquedas en Google para un mismo *topic*. Así que, para cada uno de los documentos recuperados por Google deberíamos obtener el

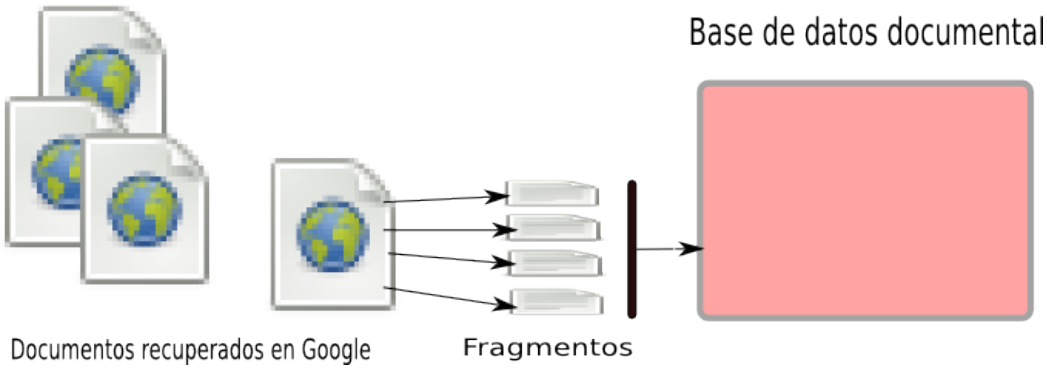

Fig. 2. El espacio de búsqueda

documento original, convertirlo a texto, descomponerlo en fragmentos, obtener los términos de cada fragmento y calcular sus pesos.

2.2 Conversión a texto

En líneas generales la conversión a texto plano puede acometerse mediante herramientas más o menos convencionales. Sin embargo, el uso de codificaciones dispares de caracteres introduce algunas dificultades. Para idiomas que usan caracteres no contenidos en el ASCII estándar, la codificacion y decodificación de tales caracteres es una fuente de problemas. Simplemente la detección del sistema de codificación es, en muchos casos, problemática [4]. Como ejemplo, hemos utilizado el Universal Encoding Library Detector (conocido generalmente como *chardet*), un módulo para Python basado en las librerías para detección de codificaciones utilizadas por Mozilla [5].

2.3 Fracionamiento de los documentos

Varias técnicas pueden ser utilizadas para fragmentar los documentos y obtener pasajes de texto más o menos cortos [6]. En líneas generales, unas están basadas en el tamaño de los fragmentos, el cual, a su vez, puede estimarse en *bytes* (o en caracteres) o en palabras. Otras están orientadas a la separación en frases o párrafos. El primer tipo de técnicas produce, obviamente, fragmentos o trozos más homogéneos en tamaño, pero a menudo carentes de sentido alguno, ya que el punto por el que se parte es ciego. Las otras técnicas tienden a producir fragmentos de muy diferente tamaño. Además, su aplicación no siempre es fácil; en muchos casos la conversión de documentos HTML a texto plano hace desaparecer la separación entre párrafos, las diferencias entre saltos de línea blandos y duros; y también elimina u oculta elementos estructurales, como las tablas.

Un enfoque simple, como la elección de un carácter ortográfico, como el punto, para fragmentar el texto, tiende a producir fragmentos demasiado cortos, y, por tanto, poco útiles para los objetivos de esta tarea. En nuestro caso, hemos adoptado un enfoque mixto. Después de varias pruebas, encontramos que un tamaño adecuado para cada fragmento era alrededor de 1500 caracteres, pero como buscábamos fragmentos que tuvieran sentido informativo, nuestro fragmentador busca el carácter ortográfico elegido (el punto) más cercano a los 1500 caracteres, y parte el texto por ahí.

2.4 Otras operaciones de análisis léxico

Adicionalmente. se efectuaron otras operaciones: conversión a minúsculas, eliminación de acentos, eliminación de palabras vacías (mediante una larga lista de palabras vacías para todos los idiomas contemplados), y aplicación de un sencillo pero eficaz *s-stemmer* [7] .

Cada fragmento obtenido tras estas operaciones fue considerado un documento independiente. De los documentos así obtenidos se extrajeron los términos, los cuales fueron pesados con un esquema de peso ATU (slope=0.2) [8], aplicando el bien conocido modelo vectorial.

3 Formación de las consultas

De alguna manera, nuestro objetivo es resolver la tarea utlizando técnicas de recuperación convencionales, o al menos ya conocidas. A partir de la colección de documentos formada con los fragmentos obtenidos, debemos seleccionar aquéllos que son más útiles para cada *topic*. La clave de nuestro enfoque está en componer consultas que puedan conseguir una selección adecuada. Para componer dichas consultas disponemos de varias fuentes de información; en primer lugar, tenemos *topics* con un título corto y una breve descripción. Además, tenemos, para cada *topic*, unos cuantos documentos denominados *fuentes conocidas*, a texto completo. También tenemos las consultas hechas a Google para obtener información sobre cada *topic*.

De esta manera, podemos utilizar los *topics* (tanto los títulos como las descripciones) como núcleo o base de cada consulta, y enriquecer o aumentar las consultas con términos provenientes de las *fuentes conocidas*. Las *fuentes conocidas* son documentos completos, algunos de ellos muy largos, que pueden contener muchos términos. Podemos preguntarnos si tal vez esto introducirá demasiado ruido en la consulta. Una posible solución es pesar los términos provenientes de las *fuentes conocidas* de una forma diferente a los que provienen del título y la descripción de cada *topic*.

Además, es también posible considerar diferentes estructuras o campos en las *fuentes conocidas* dado que en su mayor parte son páginas HTML (title, body, headings, meta tags, etc.). Experimentos anteriores en ediciones anteriores de CLEF [9] mostraron la importancia para la recuperación de algunos de esos campos, así como el escaso interés de otros. El campo más interesante, en este sentido, es el ancla de los *backlinks* [10]. Sin embargo, dado que tenemos un conjunto muy reducido de documentos, no tenemos muchos *backlinks* con los que trabajar; no obstante, parecen especialmente importantes aquéllos que, desde las *fuentes conocidas* apuntan a alguno de los documentos recuperados por Google.

Así, hemos utilizado en las consultas los términos de títulos y descripciones de los *topics*, más los términos de los anclas mencionadas antes. A esto hemos añadido los términos de las *fuentes conocidas* pero pesados de diferente forma. En ediciones anteriores de WebCLEF hemos trabajado en el uso de diferentes fuentes de información en la recuperación, y en cómo mezclar o fusionar esas fuentes [11]. En esta ocasión hemos elegido modificar los pesos de los términos operando sobre la frecuencia de éstos en cada documento. El esquema de peso elegido también para las consultas es ATU (slope=0.2), razón por la cual el peso es directamente proporcional a la frecuencia del término en el documento; así, hemos establecido un coeficiente por el cual multiplicar dicha frecuencia..

4 Resultados preliminares

Diversas pruebas llevadas a cabo varían en función de este coeficiente: una de ellas mantiene la frecuencia original, por lo que los términos provenientes de las *fuentes conocidas* son pesados igual que los de los *topics*. Otras pesan los términos de las fuentes conocidas reduciendo el peso en diversas medidas; y una prueba adicional no utiliza en absoluto los términos de las *fuentes conocidas*.

Lamentablemente, a la hora de redactar estas líneas carecemos aún de resultados oficiales, acordes con la metodología de evaluación CLEF. Sin embargo, una evaluación oficiosa, basada en una prospección manual de los *snippets* recuperados nos permite avanzar algunos de los problemas que habrá que abordar, así como áreas de trabajo futuro.

Los resultados ofociosos de estas pruebas muestran poca diferencia entre ellas. Parece que usar los términos de las fuentes conocidas es más útil que no usarlos. Pero debemos tener en cuenta que varios *topics* (casi la mitad de ellos) no producen ningún resultado útil. El problema principal parece estar en la naturaleza de los bloques o fragmentos de texto recuperados. Muchos de ellos, aún estando relacionados de alguna forma con el tema o *topic* deseado no contienen información útil para los supuestos del escenario diseñado para las pruebas o experimentos; por ejemplo, muchos de éstos simples referencias bibliográficas, o enlaces a otros documentos. Apuntan a otros documentos tal vez útiles, pero ellos mismos no contienen información directamente utilizable para los fines del supuesto trabajado.

Es importante señalar que, en este escenario concreto, tal vez el esquema de peso elegido no sea el más adecuado. En efecto, en el modo de calcular el peso o importancia de cada término en un documento dado se suele aplicar algún normalizador que permita obviar las diferencias de tamaño (en número de términos) etre los documentos; en nuestro caso ya hemos comentado que aplicamos el *Pivoted document length normalization*, que es aceptado como uno de los más eficaces. Sin embargo, en este escenario concreto puede ser interesante primar los bloques de texto más grandes, siempre dentro de ciertos límites. Ya hemos comentado que el sistema utilizado para fragmentar las páginas o documentos produce bloques de tamaños dispares, aunque con un límite máximo en torno a los 1500 caracteres. Algunos de los bloques producidos acaban siendo tan pequeños que resultan poco útiles y, en general, los bloques que, manteniendo coherencia en su contenido, son de tamaño mayor resultan ser los más interesantes. Cabe preguntarse, por este motivo, si otro esquema de normalización de pesos, o incluso la no normalización podrían mejorar los resultados.

Otro problema importante es la gran cantidad de información duplicada o casi duplicada que podemos encontrar sobre un mismo tema; la dificultad mayor estriba en que no se trata de duplicados exactos. El mismo texto con otros añadidos de otras fuentes, el mismo texto pero en páginas con cabeceras, pies, menúes, barras laterales, etc. diferentes, que producen bloques de texto que no son idénticos considerados como cadenas de caracteres, pero sí duplicados desde el punto de vista de la información contenida. Hemos utilizado el coeficiente de Dice como medida para comparar fragmentos y detectar duplicados aproximados. Comop ejemplo, si consideramos duplicados aproximados los que arrojan un coeficiente superior a 0.7 y aplicamos este umbral a los fragmentos o bloques recuperados para cada topic, encontramos que el 11.08 % son duplicados.

En este sentido, parece que la conversión de documentos HTML a texto plano mediante herramientas convencionales tiene evidentes limitaciones. En efecto, este tipo de documentos presenta en muchos casos una estructura visual [12] que la conversión convencional es incapaz de respetar. Este es el caso de los bloques en que muchas plantillas dividen las páginas web generadas por diferentes aplicaciones; algunos de estos bloques, por ejemplo, tienen sólo interés navegacional y no contienen, obviamente, información relevante para nuestros fines; otros contienen noticias de copyright, de contacto o responsabilidad de la página, o incluso publicidad, u otros elementos poco o nada útiles para nuestro caso: calendarios, votaciones, formularios de logins, etc.

La figura 3 muestra una página que es *fuente conocida* de uno de los *topics*, en la que se han marcado las áreas que contienen información relevante. El ejemplo es siginificativo, porque los artículos de la *wikipedia* son con frecuencia *fuente conocida*. La misma página, otro lado, convertida a texto plano de forma convencional, produce un fichero de 6155 caracteres, de los cuales sólo 1177 (menos del 20 %) corresponden a información relevante.

Una conversión que identificara este tipo de elementos [13] permitiría descartar partes considerables de las páginas, que producen en nuestro sistema bloques de texto no útiles. Algunas pruebas informales dan idea de las dimensiones de este problema: aplicamos un enfoque simple (incluso burdo), filtrando y eliminando fragmentos basándose en una heurística simple: bloques con demasiadas lineas en blanco, con líneas demasiado cortas, con pocas palabras en relación al tamaño del fragmento, etc.. Así, de 639 215 fragmentos obtenidos de los documentos conseguimos eliminar 165 442 (=25.88 %).

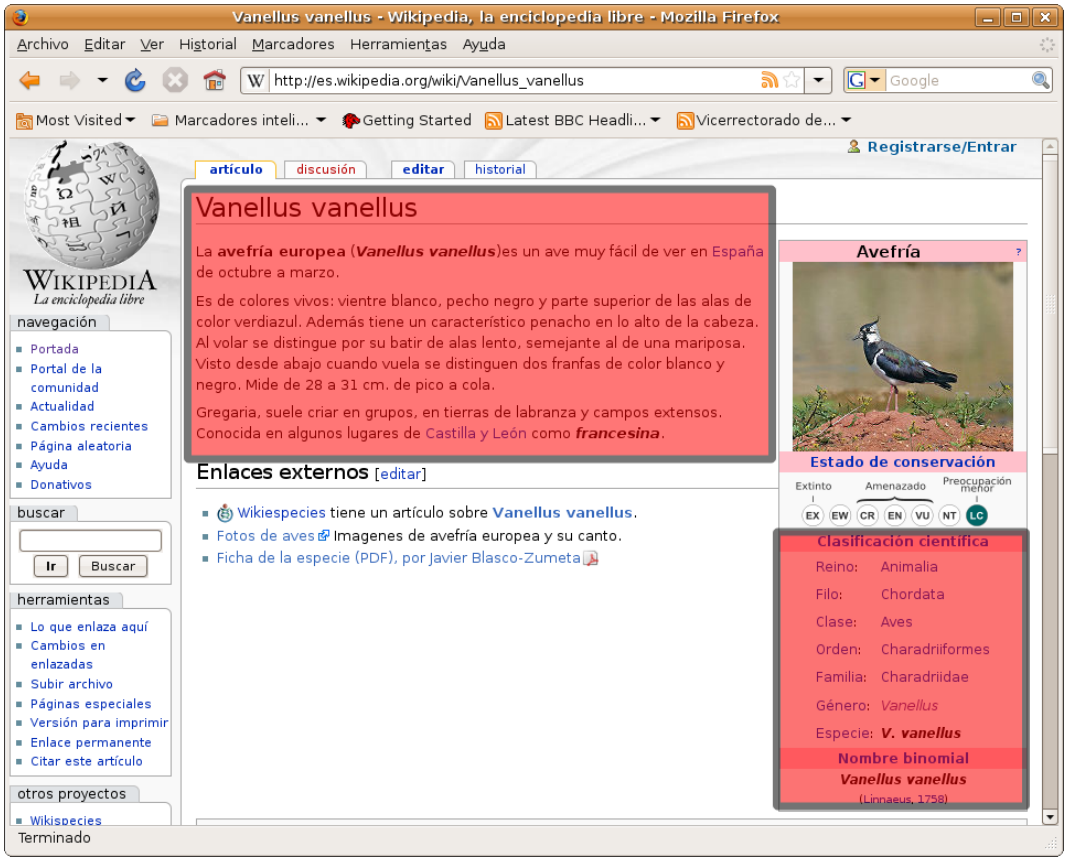

Fig. 3. Sólo las áreas marcadas son relevantes

5 Conclusiones

Hemos descrito nuestra enfoque y nuestros experimentos en la forma de abordar una situación real que se plantea cuando se necesita obtener información del web. Los primeros resultados obtenidos indican que tal vez éstos pudieran mejorarse si pudiéramos aislar en las páginas web aquéllas partes susceptibles de contener información relevante para nuestras necesidades, descartando bloques y áreas de dichas páginas no relevantes. Igualmente, la detección y eliminación de información redundante, pero no idéntica formalmente, podría mejorar de forma notoria los resultados.

Referencias

1. Braschler, M., Peters, C.: Cross-language evaluation forum: Objectives, results, achievements. Information Retrieval **7**(1-2) 7–31
2. Jijkoun, V., de Rijke, M.: Overview of webclef 2007. In Nardi, A., Peters, C., eds.: Working Notes for the CLEF 2007 Workshop
3. Becchetti, L., Castillo, C., Donato, D., Baeza-YATES, R., Leonardi, S.: Link analysis for web spam detection. ACM Trans. Web **2**(1) (2008) 1–42
4. Li, S., Momoi, K.: A composite approach to language/encoding detection. In: 19th International Conference on Unicode. (2001)
5. Pilgrim, M.: Universal encoding detector
6. Allan, J., Wade, C., Bolivar, A.: Retrieval and novelty detection at the sentence level. In: SIGIR '03. (2003) 314–321
7. Figuerola, C.G., Zazo, Á.F., Rodríguez Vázquez de Aldana, E., Alonso Berrocal, J.L.: La recuperación de información en español y la normalización de términos. Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial **8**(22) (2004) 135–145
8. Singhal, A., Buckley, C., Mitra, M.: Pivoted document length normalization. In: Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, August 18–22, 1996, Zurich, Switzerland (Special Issue of the SIGIR Forum), ACM (1996) 21–29
9. Sigurbjörnsson, B., Kamps, J., Rijke, M.d.: Overview of webclef 2005. [14]
10. Figuerola, C.G., Alonso Berrocal, J.L., Zazo Rodríguez, Á.F., Rodríguez, E.: REINA at the WebCLEF task: Combining evidences and link analysis. [14]
11. Beitzel, S.M., Jensen, E.C., Chowdhury, A., Grossman, D., Frieder, O., Goharian, N.: On fusion of effective retrieval strategies in the same information retrieval system. Journal of the American Society for Information Science and Technology (JASIST) **55**(10) (2004) 859–868
12. Yang, Y., Zhang, H.: Html page analysis based on visual cues. In: ICDAR, IEEE Computer Society (2001) 859–864
13. Kang, J., Choi, J.: A preliminary report for an information extraction system based on visual block segmentation. Technical Report TR-IS-2007-1, Hanyang University, Intelligent Systems Laboratory (2007)
14. Peters, C., ed.: Results of the CLEF 2005 Cross-Language System Evaluation Campaign. Working notes for the CLEF 2005 Workshop, 21-23 September, Vienna, Austria. In Peters, C., ed.: Results of the CLEF 2005 Cross-Language System Evaluation Campaign. Working notes for the CLEF 2005 Workshop, 21-23 September, Vienna, Austria. (2005)