Mejoras en la recuperación de información en la Web mediante el tratamiento de la información de los enlaces (1)

José L. Alonso Berrocal
Carlos García Figuerola
Ángel F. Zazo Rodríguez
Universidad de Salamanca (España)

Resumen

El objetivo principal de este trabajo es intentar comprobar la posible mejora en la recuperación de información en la Web mediante técnicas de posicionamiento o *ranking*. La metodología consistirá en recoger la información del web mediante un robot, en concreto uno elaborado por nosotros y llamado *SACARINO* (Sonda AutomátiCA para la Recuperación de Información en el web), que nos permita disponer de toda la información de los enlaces existentes en el espacio web analizado. Una vez finalizada la recogida de datos procederemos a generar las matrices de datos que nos permitirán aplicar las diferentes técnicas de *ranking* disponibles. En concreto pretendemos centrarnos en el PageRank, y lo calcularemos aplicando algunos de los algoritmos disponibles para ello. Una vez obtenidos estos datos intentaremos matizar este PageRank calculado por nosotros con el asignado por Google mediante un programa que hemos efectuado a tal efecto. Lo que pretendemos analizar es si este nuevo posicionamiento obtenido mejora el posicionamiento inicial calculado. Este matiz puede resultar de interés en función del diferente valor del PageRank calculado por nosotros (en función del espacio web recogido) y el que puede asignar Google con un espacio web mucho mayor y con una intervención de enlaces mucho más amplia.

Palabras clave: Recuperación de información. Cibermetría. Posicionamiento web. PageRank.

Abstract

The main aim of this work is to verify a possible improvement in the information retrieval in the Web by means of ranking techniques. The methodology will involve collecting information from the Web by means of a robot made by us and called *SACARINO* (Sonda AutomátiCA para la Recuperación de Información en la Web), that will allow us to have all the information in the links analyzed. After

collecting these data, we will start to generate the data matrixes that will alow us to apply the different techniques of ranking available. To be precise, we intend to focus on PageRank, so that we will calculate it by applying some of the algorithms available for it. Then we will try to compare this PageRank calculated by us with the one assigned by Google, by means of a program designed by us for this purpose. We want to verify if the new ranking improves the initial one. This can be useful depending on the different value of the PageRank calculated by us (according to the Web space covered) and the one that Google can assign having a much greater Web space and much more links involved.

Keywords: Information retrieval. Cybermetrics. Ranking. PageRank.

1. Introducción

El trabajo planteado en este artículo surgió ante la necesidad de tener que instalar un mecanismo de búsqueda de información en una sede web de nuestra universidad, de modo que tuvimos que pensar en los mecanismos necesarios para poder llevar a cabo el objetivo planteado. Inmediatamente pensamos que debíamos establecer un protocolo de configuración que, además de ofrecernos un sistema de consulta, nos permitiera evaluar la posible mejora de esa recuperación utilizando técnicas de posicionamiento. Los problemas que teníamos que resolver eran los siguientes:

- Debíamos resolver el problema de la recogida de datos en la web.
- Una vez recogidos los datos debíamos implementar un mecanismo de indización que nos facilitara el mecanismo de consulta a los datos.
- Para evaluar la posible mejora en el proceso de recuperación de información creímos conveniente conocer el PageRank de todas las páginas obtenidas e incluirlo en el proceso de consulta para adecuar el posicionamiento de los resultados.
- Finalmente, desarrollamos un programa de consulta que nos permitiera evaluar la posible mejora empleando el posicionamiento por el PageRank.

En la figura 1 podemos ver la arquitectura básica de un mecanismo de búsqueda a partir de la información de la web.

2. Recorrer la web

Según Baeza-Yates *et al.* (2005), una de las grandes ventajas de la Web es la capacidad de relacionar información mediante vínculos o enlaces. Estas relaciones además van a permitir a los usuarios una gran flexibilidad en el momento de buscar la información de su interés. Por esto, el modelo web se planteó ya desde sus inicios como un grafo dirigido. En este grafo, cada página es un nodo y cada arco representa un enlace entre dos páginas. Estos enlaces no están puestos al azar, tie-

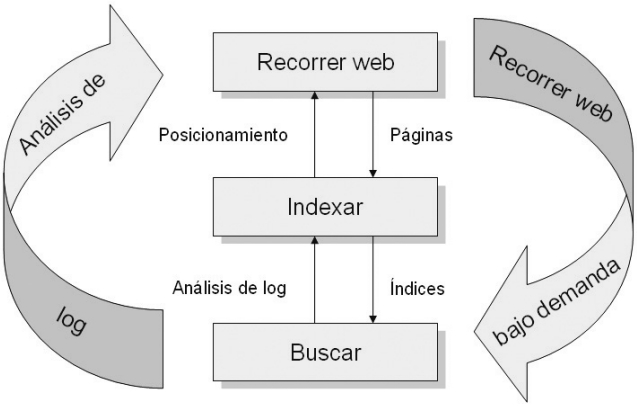

Figura 1. Arquitectura de los buscadores. Esquema modificado de Castillo (2004)

nen una intencionalidad. Las páginas normalmente tienen enlaces hacia otras páginas con el mismo tema. Además, las mejores páginas tienden a ser más referenciadas de lo normal. La Web, como grafo, tiene una estructura que se puede clasificar como *red libre de escala*. Las redes libres de escala, al contrario que las redes aleatorias, se caracterizan por una distribución dispareja de los enlaces. Estas redes han sido el tema de una serie de estudios entre los que cabe resaltar por su claridad los de Barabási (2002), y se caracterizan como redes en las cuales la distribución del número de enlaces sigue una ley de potencias (Baldi *et al.*, 2003; Alonso *et al.*, 2001).

La recuperación de información es el área de la ciencia que nos permite obtener la información necesaria acerca de una materia a partir de una colección de datos. Esto no es lo mismo que la recuperación de datos, en la que el contexto de los documentos consiste principalmente en determinar cuál de los documentos de la colección contiene las palabras de la consulta del usuario. El problema que se nos plantea en la Web es el de la abundancia de información debida a la explosión documental en la que nos encontramos en la actualidad, que puede ser medida por exabytes (10^{18}) de información.

> Print, film, magnetic, and optical storage media produced about 5 exabytes of new information in 2002. [...] We estimate that new stored information grew about 30 a year between 1999 and 2002. [...] Information flows through electronic channels —telephone, radio, TV, and the Internet— contained almost 18 exabytes of new information in 2002, three and a half times more than is recorded in storage media. [...] The World Wide Web contains about 170 terabytes of information on its surface. (Lyman and Varian, 2003)

Una solución al problema planteado es el uso de la estructura hipertexto de la Web, empleando los enlaces entre las páginas, como las citas en los mecanismos

de la literatura clásica, para encontrar los documentos más importantes. La utilidad de este planteamiento ya fue demostrada por Alonso et *al.* (1999) y Berrocal *et al.* (2002). Más recientemente se ha valorado este sistema como muy eficaz en el trabajo de Cothey (2004).

Para el proceso de recorrer la Web hemos utilizado el programa SACARINO, desarrollado por nuestro grupo de investigación y que tuvo sus inicios en Alonso (1996). Las características fundamentales de dicho robot serán comentadas en la ponencia presentada en este mismo número de la revista *Scire*, aunque mostramos aquí su pantalla principal (figura 2).

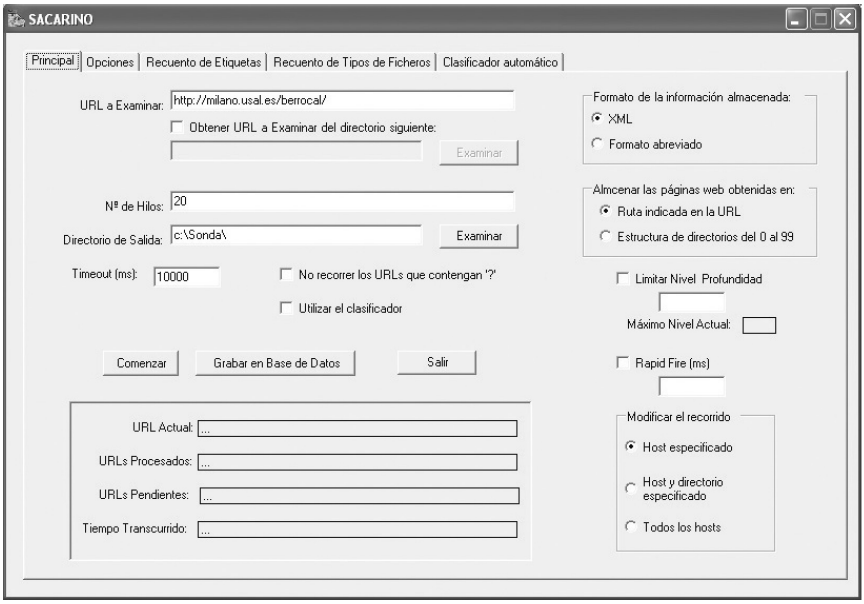

Figura 2. Pantalla principal de SACARINO

3. El proceso de indización

Para el proceso de indización estudiamos las diferentes propuestas que estaban a nuestro alcance intentando siempre que fuese una herramienta de libre acceso con el fin de permitir que cualquier otra sede web pudiera implementar el mismo mecanismo. Después de evaluar diferentes productos, nos decidimos por las características de Swish-e de la Universidad de Berkeley, ya que cumplía los siguientes requisitos:

- Completo (es un bot, indiza y permite las búsquedas).
- Código fuente disponible.
- Multiplataforma.
- Multiformato.
- Modularidad.
- Libre distribución.

Con la utilización de este producto cualquier centro tiene capacidad para poder recorrer toda la información generada en su sede (aunque nosotros empleamos SACARINO por poseer una mayor flexibilidad), indizar esta información y generar mecanismos de consulta de la misma. Por todo ello se convierte en una herramienta interesante para la consecución de los propósitos iniciales.

El indizador es el módulo encargado de generar unos índices para que la posterior búsqueda sea lo más eficiente posible. Está escrito en C y es sencillo y de fácil adaptación (motivo adicional por el que es interesante esta herramienta, con la posibilidad de adecuarse a posibles necesidades en el proceso de indización). Los índices se almacenan en formato texto, guardando por cada palabra un registro compuesto por tantos vectores como apariciones de la palabra, y en cada vector la información necesaria para identificar el documento y la posición de la palabra en el mismo.

Es conveniente indicar dos características importantes que se especificaron en el proceso de indización, que pueden tener incidencia en dicho proceso.

1. Hemos utilizado la técnica de *stemming* optimizada para el castellano. El *stemming* o lematización es para Lovins (1968) un procedimiento informático por el que se reducen todas las palabras a su forma común al quitar los sufijos de derivación y flexión. Esta técnica ha demostrado sobradamente la mejora en los procesos de recuperación de información; puede obtenerse información complementaria y de gran interés en el trabajo de Gómez Díaz (2005).
2. Se han eliminado palabras vacías. Para que los índices no sean demasiado grandes se eliminan de ellos las palabras funcionales de apariencia frecuente. Además, esta eliminación es importante porque afecta a la distribución de los pesos de los términos. Esta lista de palabras vacías es la elaborada por Gómez Díaz (2005).

Como en nuestro caso el proceso de recorrer la Web fue realizado con nuestra herramienta SACARINO, las páginas se almacenaron añadiendo la siguiente modificación:

```
<DOC>
<DOCNO>1</DOCNO>
<DOCHDR>
http://www.usal.es/web-usal/
HTTP/1.1 200 OK
Date: Thu, 23 Jun 2005 07:34:09 GMT
Server: Apache
Last-Modified: Wed, 22 Jun 2005 08:20:08 GMT
ETag: "244a4-a59c-42b91f38"
Accept-Ranges: bytes
Content-Length: 42396
Connection: close
Content-Type: text/html
</DOCHDR>
<html>
Aquí tenemos la información del documento.
</html>
</DOC>
```

Como se puede ver, hemos añadido una etiqueta `<DOCHDR></DOCHDR>` con información relevante del documento. En el proceso de indización simplemente tuvimos que indicarle que ejecutara esta acción para la información almacenada previamente en disco, y añadir la opción `StoreDescription HTML2 <dochdr>` en la configuración del indizador. De esta forma, en la respuesta de la consulta podemos tratar de forma independiente el contenido entre las etiquetas `<DOCHDR>` `</DOCHDR>` y generar así los índices adecuados.

4. La búsqueda de información

Es el módulo que por medio de una interfaz captura las palabras clave en que el usuario basa su pregunta y, consultando los índices, genera una página con enlaces a las páginas que contienen la información correspondiente. Desarrollamos un programa informático para la realización de la consulta (utilizando Swish-e), así como la oportuna interfaz para un mejor desarrollo de todo el proceso. En el comando que se pasa al programa Swish-e, lo más destacable es la especificación del formato de salida `-x %r@$%d`, con lo que podemos controlar el formato de salida, compuesto por *swishrank* y *swishdescription* (con el contenido de `<DOCHDR></DOCHDR>`).

Como deseábamos evaluar la posible mejora en la recuperación de información mediante técnicas de posicionamiento, decidimos emplear los mecanismos de posicionamiento basados en la conectividad con las siguientes características:

- La técnica debe ser independiente de la consulta.
- Nos decidimos por el empleo del PageRank.

La técnica PageRank ha demostrado suficientemente sus características como método de posicionamiento en los procesos de recuperación de información (Dominich and Skrop, 2005). El procedimiento básico de obtención del PageRank es el siguiente:

- La fórmula básica de cálculo del PageRank determina que el valor de una página está influenciado por el número de enlaces que recibe desde otras páginas y matizado por la importancia de las páginas que la enlazan (figura 3).

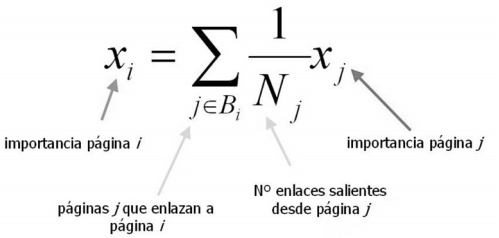

Figura 3. Fórmula básica de cálculo del PageRank

Se han descrito diversos problemas en ese mecanismo básico de obtención del PageRank y se han planteado soluciones a los mismos. En concreto, nosotros obtenemos el PageRank según la reformulación de Sung Jin and Sang Ho (2002), que ofrece solución a bastantes de los problemas observados.

- El cálculo se realiza a partir de la matriz de adyacencia del grafo que se obtiene de las relaciones entre los documentos, multiplicado por un vector de inicialización (figura 4).

	1	2	3	4	5	6	7	8	9	10			
1	0	0	1/2	0	0	0	0	0	0	0		?	
2	1/2	0	0	0	0	0	1/5	0	0	0		?	
3	1/2	1/2	0	0	0	0	0	0	1/2	0		?	
4	0	1/2	0	0	0	1/4	1/5	1	0	0		?	
5	0	0	0	1/2	0	1/4	0	0	1/2	0		?	=
6	0	0	0	0	1	0	0	0	0	0		?	
7	0	0	1/2	1/2	0	0	0	0	0	0		?	
8	0	0	0	0	0	1/4	1/5	0	0	0		?	
9	0	0	0	0	0	0	1/5	0	0	1		?	
10	0	0	0	0	0	1/4	1/5	0	0	0		?	

Figura 4: Cálculo del PageRank

- Es un proceso iterativo al final del cual se obtiene un vector con el valor del PageRank de todas las páginas del grafo (figura 5).

---- Iteración 1 ----	---- Iteratión 3 ----	---- Iteración 10 ----	---- Iteración 20 ----
0.050	0.060	0.046	0.045
0.070	0.072	0.048	0.047
0.150	0.093	0.092	0.091
0.195	0.133	0.153	0.153
0.125	0.126	0.161	0.162
0.100	0.183	0.160	0.161
0.100	0.123	0.121	0.122
0.045	0.066	0.066	0.065
0.120	0.080	0.088	0.089
0.045	0.066	0.066	0.065
NORM es 1.000000	NORM es 1.000000	NORM es 1.000000	NORM es 1.000000
Residual es 0.380000	Residual es 0.297500	Residual es 0.017855	Residual es 0.000896

El "eigen vector" principal (contiene los valores del PageRank)

Figura 5. Sistema iterativo en cálculo de PageRank

En un primer momento, el PageRank con el que íbamos a trabajar era el calculado para el propio espacio web recogido y valoraba por lo tanto la importancia de las páginas según el diseño de los administradores del espacio web. Con los primeros trabajos ya pensamos en poder incorporar también a nuestro sistema el PageRank que Google tiene de esas páginas. Con este PageRank de Google, la importancia de las páginas va a estar determinada por la valoración de las páginas existentes en Google. Tenemos por tanto dos cálculos del PageRank, uno con la valoración interna de la propia sede recogida y otro con la valoración de un número bastante mayor de páginas y posiblemente muchas de ellas de fuera de la sede recorrida.

Para obtener el PageRank local generamos el grafo de la sede web y, con los programas desarrollados al efecto para Matlab (que forman parte del paquete EloisaBot Tools), obtuvimos en un corto espacio de tiempo todos los datos correspondientes al PageRank de todas las páginas recorridas con SACARINO. Para obtener el PageRank de Google realizamos un programa que permite obtener este dato, que utiliza un módulo existente para lenguaje PERL. Es un proceso rápido que nos permitió obtener este dato de forma correcta. Se puede descargar un paquete RPM para sistema Linux de `http://rpm.pbone.net/index.php3/stat/4/idpl/1920109/com/perl-Algorithm-PageRank-0.08-8.noarch.rpm.html`.

De esta forma contamos con la posibilidad de matizar los posibles resultados teniendo en cuenta dos valoraciones distintas. Esto determinó la necesidad de utilizar unos mecanismos de normalización y de fusión de los datos para garantizar los resultados.

5. Normalización y fusión de los datos

5.1. Normalización

En todo proceso de transformación de datos es preciso realizar diferentes modificaciones para un tratamiento adecuado de los mismos. Existen diferentes técnicas para esta transformación de datos:

1. *Smoothing*. Esta técnica habitualmente se utiliza para eliminar el ruido que puedan contener los datos, e incluye procedimientos de *clustering* y regresión.
2. Generalización. En esta técnica se sustituyen descripciones de bajo nivel por otras de alto nivel.
3. Normalización. Consiste en reescalar los valores de los atributos.
4. Construcción de atributos. Se crean nuevos atributos y se añaden a los ya existentes.

En nuestro caso hemos tenido que recurrir al procedimiento de normalización para ajustar convenientemente todos los datos que hemos obtenido. Por un lado, los resultados de la consulta en Swish-e; por otro, el valor del PageRank local, y por otro el PageRank de Google. Existen diferentes técnicas de normalización que tienden a ajustar los valores a 1.

- Normalización Min-Max. Realiza una transformación lineal de los valores de los datos originales. La fórmula correspondiente se muestra en la figura 6.

$$nuevo\_v = \frac{v - \min_x}{\max_x - \min_x} . (nuevo\_\max_x - nuevo\_\min_x) + nuevo\_\min_x$$

Figura 6. Normalización Min-Max

En esta normalización se preservan las relaciones entre los datos originales.

- Normalización Z-score. En este procedimiento se utilizan la media y la desviación estándar para calcular el nuevo valor. Esta normalización es muy sensible a valores pequeños de la desviación estándar.

$$nuevo\_v = \frac{v - \mu_x}{\sigma_x}$$

Figura 7. Normalización Z-score

- Normalización a escala decimal. En este caso se trata de un mecanismo simple por el que se ajustan todos los valores a sus datos decimales.

$$nuevo\_v = \frac{v}{10^c}$$

Figura 8. Normalización a escala decimal

Con los datos normalizados según Z-score alimentamos el sistema de consulta para poder aplicarlos en el proceso de búsqueda del sistema.

5.2. Fusión

Los experimentos de fusión se remontan a las TREC-1, combinando resultados de diferentes búsquedas de información a partir de una colección de los *n* documentos recuperados de cada búsqueda de información. En estos inicios los esfuerzos se centraban en posicionar adecuadamente los documentos recuperados sin tener en cuenta los valores de similaridad de los mismos. En las TREC-2 los experimentos se concentraron fundamentalmente en combinar los resultados en función de los valores de simililaridad de los documentos para cada una de las consultas realizadas. En este momento se estudiaron seis métodos de fusión empleando los valores de similaridad. Mostramos un breve resumen a continuación (Fox y Shaw, 1993):

CombMAX
Max (similitudes individuales)
CombMIN
Min (similitudes individuales)
CombSUM
SUM (similitudes individuales)
CombANZ
SUM (similitudes individuales) / Nº de similitudes distintas de 0
CombMNZ
SUM (similitudes individuales) * Nº de similitudes distintas de 0
CombMED MED (similitudes individuales)

En concreto, nosotros vamos a utilizar una modificación empleada por nosotros en el CLEF 2005 (Figuerola *et al.*, 2005), basada en CombMNZ, y que a continuación mostramos:

$$Score = \sum_{i=1}^{n} score_i \times k_i \times numberofscore! = 0$$

Figura 9. Modificación de fusión CombMNZ

6. Resultados

Para la fase de evaluación de los resultados construimos un conjunto de 20 preguntas. Las preguntas se pasaron al sistema de interrogación y obtuvimos tres conjuntos de resultados:

- Según la respuesta de Swish-e.
- Según la respuesta de PageRank local.
- Según la respuesta de PageRank de Google.

Estas tres series de resultados fueron sometidas a la consideración de un grupo de usuarios que podríamos englobar dentro del apartado de expertos. Se pasaron las encuestas y se procesaron las respuestas de las mismas, con los siguientes resultados preliminares:

- En el 85% de las consultas existe una sustancial mejora empleando las técnicas de posicionamiento. La mejora se produce tanto en el PageRank local como en el PageRank de Google.
- Cuando la respuesta de Swish-e aporta documentos muy relevantes entre los 10 primeros, el posicionamiento recoloca de forma más adecuada los mismos.
- Cuando la respuesta de Swish-e no aporta documentos relevantes entre los 10 primeros, el posicionamiento hace aflorar documentos relevantes de posiciones posteriores.
- Dependiendo de la consulta, el comportamiento entre el PageRank local y el PageRank de Google sufre ligeras variantes.
- El empleo de Swish-e y técnicas de posicionamiento mejora nuestra recuperación de forma apreciable.

7. Trabajo de futuro

El trabajo que nos queda por delante es interesante y nos permitirá finalizar nuestro trabajo de investigación sobre técnicas de *ranking*. Algunos de los nuevos datos que debemos analizar son:

- Valorar los resultados para sistemas al mejor de 20 y al mejor de 30.
- Hay que evaluar si tiene importancia la fusión entre los valores del PageRank local y el PageRank de Google. Para ello hemos desarrollado un programa de consulta en el que se nos permita matizar el factor de importancia que le asignamos a cada uno de los valores del PageRank (figura 10).
- Es preciso evaluar esta fusión para diferentes valores de la misma.
- Hay que probar con otros mecanismos de fusión.

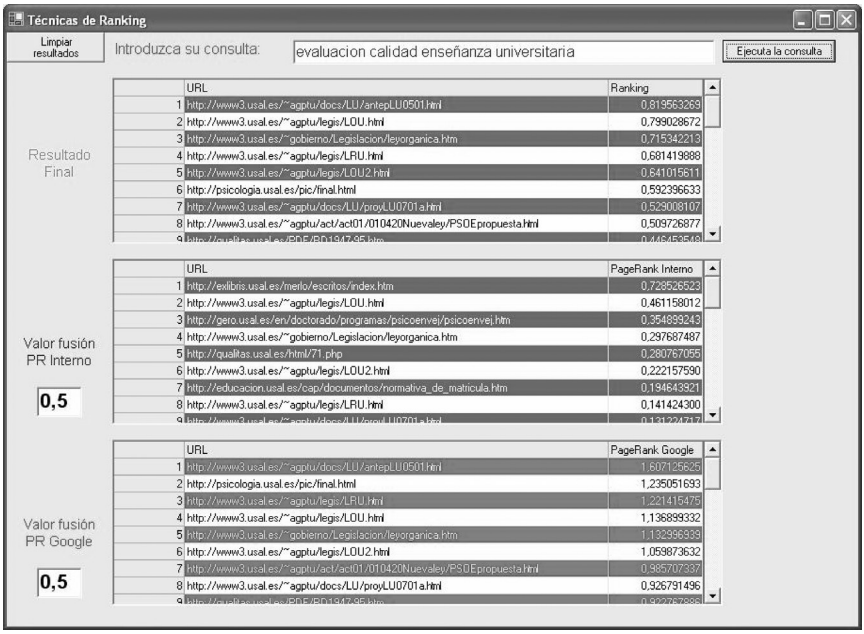

Figura 10. Interfaz de consulta. Simulador de fusión

- Es interesante incluir en este proceso la información de las etiquetas META y la información que contiene cada uno de los enlaces.
- Algunas técnicas de análisis de *path* de las URL pueden ofrecer una buena vía de exploración.

Notas

(1) Este trabajo se enmarca dentro de los resultados obtenidos por el proyecto de investigación de la Junta de Castilla y León (Proyecto SA089/04) titulado *Nuevas técnicas de* ranking *en la recuperación de información en la Web*.

Referencias

Alonso Berrocal, J. L. (1996). Herramienta software para el análisis de la documentación web: rastreo de dominios, estudio de etiquetas, tipología de ficheros, evolución de los enlaces. Grado de Salamanca, Facultad de Traducción y Documentación. Universidad de Salamanca.

Alonso Berrocal, J. L.; Figuerola, C. G.; Zazo Rodríguez, Á. F. (1999). Representación de páginas web a través de sus enlaces y su aplicación a la recuperación de información. // Scire. 5:2 (1999) 91-98.

Alonso Berrocal, J. L.; Figuerola, C. G.; Zazo Rodríguez, Á. F. (2001). Cibermetría del Web: las leyes de exponenciación. // Revista General de Información y Documentación. 11:1 (2001) 201-209.

Baeza-Yates, R.; Castillo, C.; López, V. (2005). Characteristics of the Web of Spain. // Cybermetrics, 9:1 (2005).

Baldi, P.; Frasconi, P.; Smyth, P. (2003). Modeling the Internet and the Web: probabilistic methods and algorithms. Wiley: Chichester, 2003.

Barabási, A.-L. (2002). Linked: the new science of networks. Cambridge: Perseus Publishing, 2002.

Berrocal, J. L. A.; Figuerola, C. G.; Zazo, Á. F.; Rodríguez, E. (2002). La cibermetría en la recuperación de información en el Web. // Primeras Jornadas de Tratamiento y Recuperación de Información, JOTRI-2002, Valencia, España, 4 y 5 de julio de 2002. Valencia: Universidad Politécnica, 2002. 117-124.

Castillo, C. (2004). Effective Web Crawling. Tesis doctoral. Departament of Computer Science. University of Chile. URL: <http://www.chato.cl/534/article-63160.html>.

Cothey, V. (2004). Web-crawling reliability. // Journal of the American Society for Information Science and Technology. 55:14 (2004) 1228-1238.

Dominich, S.; Skrop, A. (2005). Pagerank and interaction information retrieval. // Journal of the American Society for Information Science and Technology. 56:1 (2005) 63-69.

Figuerola, C. G.; Alonso Berrocal, J. L.; Zazo Rodríguez, Á. F.; Rodríguez, E. (2005). REINA at the WebCLEF Task: Combining evidences and link analysis. // Working Notes for the CLEF 2005 Workshop, 21-23 September, Vienna, Austria. 8.

Fox, E. A.; Shaw, J. A. (1993). Combination of multiple searches. // The Second Text REtrieval Conference (TREC-2). NIST Special Publication. 500-215.

Gómez Díaz, R. (2005). La lematización en español: una aplicación para la recuperación de información. Gijón: Trea, 2005.

Lovins, J. (1968). Development of a stemmning algorithm. // Mechanical Translations and Computacional Linguistics.11:1-2 (1968) 22-31.

Lyman, P.; Varian, H. R. (2003). How much information 2003?. Berkeley: University of California, 2003. URL: <http://www.sims.berkeley.edu/how-much-info-2003>.

Sung Jin, K.; Sang Ho, L. (2002). An improved computation of the pagerank algorithm. // ECIR. Lecture Notes in Computer Science. 2291 (2002) 73-85.