Software libre en la representación, búsqueda, recuperación e intercambio de información.

Dante Ortiz Ancona
Dirección General de Bibliotecas,
UNAM, México

Objetivo

Exponer experiencias en la aplicación de herramientas de software libre para la representación, búsqueda, recuperación e intercambio de información.

Resumen:

Ilustrar las técnicas mas recientes y sofisticadas tales como II, LSI y ECI (de sus siglas en inglés “Inverted Indexing”, “Latent Semantic Indexing” y “Effective Conceptual Indexing” respectivamente) para la creación de índices en los sistemas de recuperación de información.

Presentar algunas herramientas de software libre, tales como Lucene, Zilverline, Lius, Regain , Zebra y Managing Gigabytes, para indexación y búsqueda de información, ilustrando su funcionamiento y operación así como la técnica de indexación que emplean. Se describirá como incorporar un analizador lexicográfico para el lenguaje español.

Presentar un proyecto para mejora de relevancia en recuperación de información en español, utilizando: raíces para formación de pala-

bras (stemming), eliminación de palabras sin importancia en la búsqueda de información (stopwords) y tesauros.

Describir algunos paradigmas de investigación y desarrollo en los sistemas de representación, organización, búsqueda y recuperación de información. Los paradigmas a describir serán: paradigma estadístico, paradigma lingüístico, paradigma bibliotecológico y paradigma de inteligencia artificial (redes neuronales, sistemas basados en conocimiento, procesamiento de lenguaje natural).

Presentar y describir algunas herramientas de software libre, tales como Fedora, DSpace Greenstone para la implantación de repositorios digitales ilustrando sus protocolos para el intercambio de información así como sus aplicaciones en el desarrollo de bibliotecas digitales y para la conservación y preservación de información digital.

1. Métodos para crear índices en los sistemas de recuperación de información

La representación de Índice Invertido (II) es el método dominante para indexar texto, pero no es conveniente en búsqueda de similitud entre documentos. La representación II consiste de una lista de identificadores de documentos, uno por cada palabra en el léxico. Cada palabra W tiene asociada una lista de todos los identificadores de documentos que la contienen. Adicionalmente, meta-información que se almacena junto con el identificador del documento, tal como frecuencia de la palabra y posición en el documento. Mediante alguna función de similitud se detectan, en el índice, las palabras correspondientes de una búsqueda. Los detalles de esta función se muestran en [1] y [2]. El desempeño de la representación II empeora cuando se incrementa el número de palabras en un documento o en los casos en que una palabra tiene una lista invertida demasiado grande.

El Indexado Semántico Latente (ISL) es un método para mejorar la calidad en la búsqueda por similitud transformando documentos del conjunto de palabras original a un espacio de conceptos [3]. La idea principal de este método es proyectar los datos en un espacio pequeño, de los datos originales, eliminando los efectos nocivos de sinonimia y polisemia. Trata de minimizar ambigüedad, redundancia y

vocabulario sin comprimir la representación. ISL transforma los datos de una representación indexada dispersa (como en II) con dimensionalidad alta a una representación en un espacio real mucho menos disperso. Desafortunadamente, ISL transforma los datos en un dominio que no es posible brindar técnicas de indexado efectivas.

En el Indexado Conceptual Efectivo (ICE), un documento se representa como conjuntos de atributos que corresponden a conceptos con significado. Cada uno de estos conceptos es definido por una palabra con un peso asociado (frecuencia). La palabra con el peso representa a un conjunto de palabras relacionadas semánticamente. La representación ICE es una representación comprimida que reduce ambigüedad, redundancia y vocabulario no relacionado en un documento. Un vez que se reduce la dimensión de un documento se utiliza el método de índices invertidos para indexar los documentos. El método ICE es mucho mejor que II en búsquedas de similitud y preserva la misma calidad de los resultados, tiene una gran eficiencia computacional y de almacenamiento. Por ejemplo, en una muestra de 167000 documentos se requirió 87.7 Mb usando el método II y 8.3 Mb usando ICE [4].

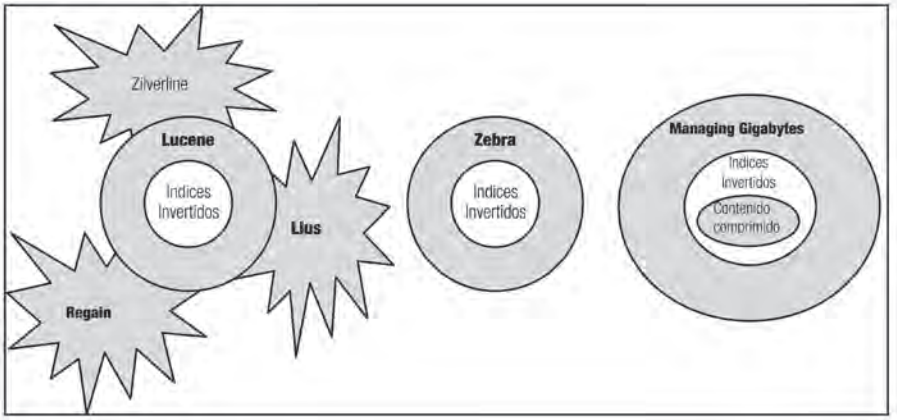
2. Software libre para indexación, búsqueda y recuperación de información

Las herramientas de software libre para indexación, búsqueda y recuperación de información más comunes son Lucene [5], Zebra [6] y Managing Gigabytes [7]. Todas estas herramientas tienen como núcleo la representación de índices invertidos (Véase figura 1). Managing Gigabytes incorpora un núcleo de nivel más bajo que maneja algoritmos bastante sofisticados para comprimir la información y manejar de manera más eficiente imágenes, audio y video.

Lucene es una interfaz para programas de aplicaciones que contiene un motor para indexar, buscar y recuperar información tanto de registros como texto completo. Es sin duda, dentro de su clase, el software con mayor respaldo en soporte, documentación y desarrollo de proyectos. Aunque utiliza principalmente el idioma inglés, provee una interfaz de programación que le permite incorporar, con gran facilidad, otros idiomas. Fue desarrollado en el lenguaje de programa-

ción Java, esto le ha permitido una gran portabilidad para funcionar en cualquier sistema operativo y arquitectura de cómputo, así como una gran interoperabilidad con otros sistemas computacionales. No utiliza un sistema de metadatos descriptivo, sin embargo, resulta bastante simple adaptarle cualquier sistema de metadatos.

Figura 1 (Software libre para indexado, búsqueda y recuperación de información)



Lucene utiliza por defecto un analizador lexicográfico, para texto en idioma inglés, eliminando del vocabulario palabras sin importancia en búsqueda de información (stopwords) tales como artículos, preposiciones, verbos comunes (is, have, get, etc.), palabras acerca de la estructura del texto, etcétera. El analizador sintáctico permite que un usuario o programador especifique un archivo que contenga esta lista de palabras dando la posibilidad de aumentar o disminuir el diccionario de palabras sin importancia. Además reconoce términos con las características siguientes: secuencias de letras y dígitos (p175waugh), apóstrofos (O´Reilly), acrónimos (H.P.), compañías (AT&T), direcciones de correo electrónico (dante@dgb.unam.mx), nombres de servidores WEB (www.dspace.org), números seriales (direcciones IP como 132.248.9.31, números de punto flotante como 3.1416). Puede desarrollarse un analizador lexicográfico para el idioma español o cualquier

otro idioma (que quizá incorpore el uso de raíces de palabras y/o tesauros) y utilizarlo en lugar del que se tiene por defecto. Por ejemplo, sea "vacias.txt" el archivo que contiene las palabras irrelevantes para la búsqueda y sea "SpanishAnalizer" el Analizador lexicográfico para el idioma español, entonces, el fragmento de código, en el lenguaje de programación Java, para construir dicho analizador lexicográfico sería: **new StandardAnalyzer(new File("vacias.txt").**

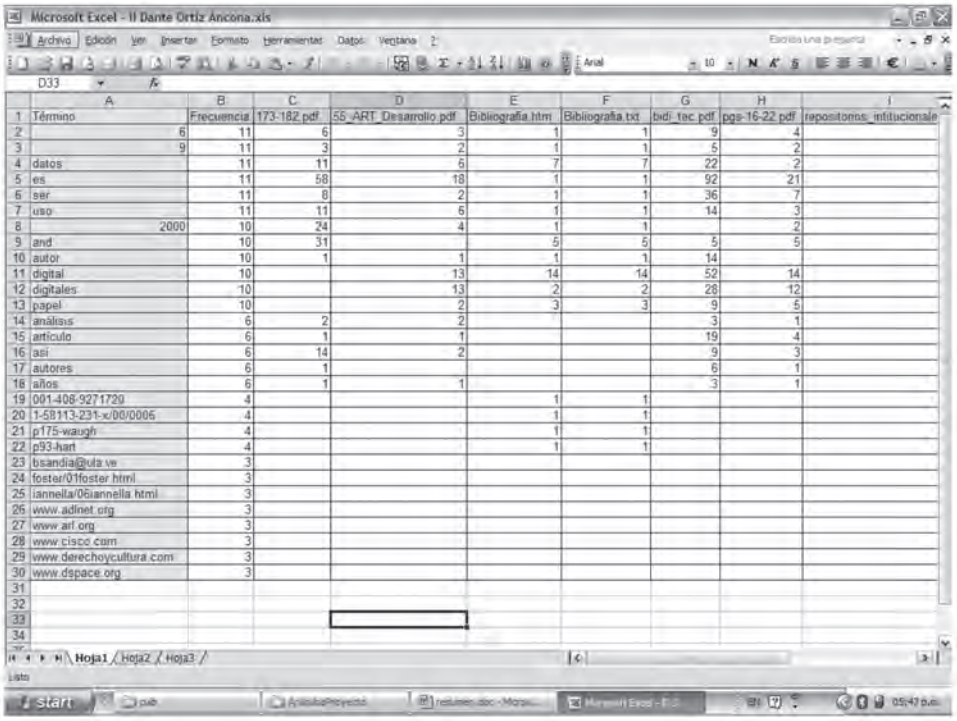
La figura 2 muestra una representación en Excel de un subconjunto, del índice construido, utilizando el analizador lexicográfico en idioma español anteriormente mencionado. La primera columna contiene la lista de términos, el primer renglón ilustra parte de una lista con un total de 11 identificadores de documentos de texto en diversos formatos. La segunda columna ilustra la cantidad de documentos en que aparece el término. La celda de intersección entre el término y el identificador del documento muestra el número de veces que aparece el término en el documento de texto. Por simplicidad no se muestran las posiciones del término dentro del documento de texto.

Aunque el motor de Lucene fue desarrollado para indexar, buscar y recuperar información en texto plano. Se han desarrollado otras herramientas de software libre tales como Zilverline [8], LIUS (Lucene Index Update and Search) [9] y Regain [10] que tienen como núcleo a Lucene (ver figura 1) y que amplían su funcionalidad al incorporar filtros que permiten transformar documentos de diferentes formatos (Word, Powerpoint, Excel, Postscript, PDF, HTML, XML, etc.) a texto plano. Estas herramientas proveen una interfaz de usuario vía WEB para administrar el índice, incorporar documentos de texto, realizar búsquedas avanzadas (incorporando operadores lógicos, de agrupamiento, de selección de campos, comodines, de proximidad y de rangos) y recuperar información ya sea textual o descriptiva.

Zebra es una interfaz para programas de aplicaciones que contiene un motor para indexar, buscar y recuperar información. Es una herramienta de propósito general y de rendimiento alto, indexa texto estructurado, lee registros en una variedad de formatos de entrada (correo electrónico, XML, MARC) proporcionando acceso a ellos a través de una poderosa combinación de expresiones de búsquedas lógicas y de relevancia. Soporta bases de datos grandes (decenas de

millones de registros, decenas de gigabytes de datos) permitiendo actualizaciones seguras en tiempo real. Soporta el protocolo estandarizado Z39.30 para recuperación e intercambio de información. Cuenta con un amplio respaldo en soporte, documentación [12] y desarrollo. Fue desarrollado en el lenguaje de programación C estándar, esto le ha permitido una gran portabilidad para funcionar en cualquier sistema operativo y arquitectura de cómputo, así como interoperabilidad con otros sistemas computacionales. Provee una interfaz en modo de comandos para administrar el índice y para búsqueda y recuperación de información, permitiendo realizar búsquedas avanzadas (incorporando operadores lógicos, de agrupamiento, de selección de campos, comodines y proximidad).

Figura 2 (Representación en Excel de un subconjunto del índice invertido, generado con un analizador lexicográfico del idioma español).



Microsoft Excel - II Dante Ortiz Ancona.xls

	A	B	C	D	E	F	G	H	I
1	Término	Frecuencia	173-182.pdf	55_ART_Desarrollo.pdf	Bibliografia.htm	Bibliografia.txt	bidi_tac.pdf	pgs-16-22.pdf	repositorios_infitucionale
2	6	11	6	3	1	1	9	4	
3	9	11	3	2	1	1	5	2	
4	datos	11	11	5	7	7	22	2	
5	es	11	58	18	1	1	92	21	
6	ser	11	8	2	1	1	36	7	
7	uso	11	11	6	1	1	14	3	
8	2000	10	24	4	1	1		2	
9	and	10	31		5	5	5	5	
10	autor	10	1	1	1	1	14		
11	digital	10		13	14	14	52	14	
12	digitales	10		13	2	2	28	12	
13	papel	10		2	3	3	9	5	
14	análisis	6	2	2			3	1	
15	artículo	6	1	1			19	4	
16	así	6	14	2			9	3	
17	autores	6	1				6	1	
18	años	6	1	1			3	1	
19	001-408-9271729	4			1	1			
20	1-58113-231-x/00/0006	4			1	1			
21	p175-waugh	4			1	1			
22	p93-hart	4			1	1			
23	bsandia@uta.ve	3							
24	foster/01foster.html	3							
25	iannella/06iannella.html	3							
26	www.adlnet.org	3							
27	www.arl.org	3							
28	www.cisco.com	3							
29	www.derechoycultura.com	3							
30	www.dspace.org	3							
31									
32									
33									
34									

Hoja1 / Hoja2 / Hoja3 /

Listo

Managing Gigabytes es una interfaz para programas de aplicaciones que contiene un motor para indexar, buscar y recuperar información de texto completo, archivos binarios, imágenes pictóricas o textuales. Tiene muy poco respaldo en soporte y la única documentación es el libro [14]. A diferencia de otras herramientas, provee algoritmos bastante sofisticados para comprimir texto e imágenes. Fue desarrollado en el lenguaje de programación C y puede interactuar con otros sistemas computacionales. No utiliza un sistema de metadatos descriptivo. Provee un diccionario, en idioma inglés, de palabras sin importancia en búsqueda de información (stopwords) y manejo de raíces de palabras (stemming) para aumentar relevancia en recuperación de documentos y reducir la dimensión del índice. Tiene implantados métodos estadísticos para clasificación y organización de información. Permite realizar búsquedas avanzadas (incorporando operadores lógicos, de agrupamiento, de selección de campos, comodines y proximidad) y recuperación de información de forma interactiva y distribuida.

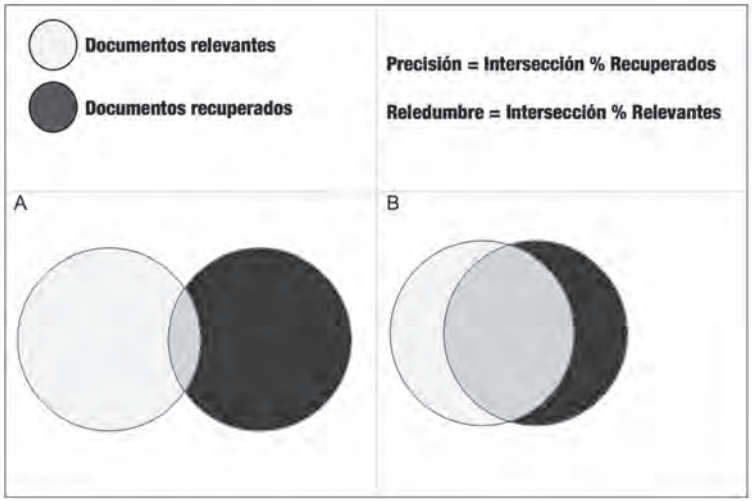
3. Proyecto para mejora de relevancia en recuperación de información en español

La figura 3 muestra dos medidas básicas en los sistemas de recuperación de información: precisión (precision) y reledumbre[1] (recall) [26]. En color verde se muestra la intersección entre los documentos relevantes para el usuario y los que fueron recuperados en la consulta que planteó. La precisión se define como el número de documentos relevantes recuperados por un usuario, en una búsqueda, dividido entre el total de documentos recuperados. La reledumbre se define como el número de documentos relevantes recuperados por un usuario, en una búsqueda, dividido entre el total de documentos que son de importancia para el usuario. Los sistemas de recuperación de infor-

1 Se construyo el término reledumbre (relevancia, certidumbre) usando reglas gramaticales y del mínimo esfuerzo, porque representa más completamente el significado del uso de la palabra recall en lugar de otras traducciones incorporadas tales como exhaustividad, llamada, revocación o recuerdo.

mación tienen como uno de sus objetivos principales incrementar el valor de la precisión y la reledumbre, es decir, transformar el diagrama de la figura 3A por el diagrama de la figura 3B. Algunas técnicas como la eliminación de *stopwords*, filtrado y poda de documentos [14] están orientados a reducir la dimensión de documentos y por tanto la reducción del tamaño del índice; propiciando un mejor desempeño y eficiencia en la búsqueda, recuperación e intercambio de información. Otras técnicas como el tratamiento de la sinonimia, polisemia, raíces de palabras y uso de tesauros, además de reducir la dimensión de los documentos, ayudan a mejorar la precisión y reledumbre en la búsqueda y recuperación de información. Se propone el desarrollo de un proyecto para mejora de reledumbre en recuperación de información en español, utilizando el software snowball [15] que implanta un algoritmo de raíces de palabras (stemming) en español, eliminación de palabras sin importancia en la búsqueda de información (stopwords) y un tesauro con términos en español para tratamiento de sinonimia y polisemia. A diferencia de [16] el tesauro propuesto debe agrupar palabras relacionadas semánticamente y que se utilice para implantar el Indexado Conceptual Efectivo en Lucene.

Figura 3 (Medidas básicas en recuperación de información)



4. Paradigmas de investigación y desarrollo en los sistemas de representación, organización, búsqueda y recuperación de información

Aunque existen algunas combinaciones de paradigmas, los más comunes son: paradigma estadístico, paradigma lingüístico, paradigma bibliotecológico y paradigma de inteligencia artificial (más generalmente sería paradigma computacional). Existen además muchas herramientas de software libre para filtrado, clasificación, agrupamiento (clustering) y visualización que incorporan todos estos paradigmas, como por ejemplo Weka [17], Yale [18], KDDnuggets [19] y UCI [20].

El paradigma estadístico tiene una gran cantidad de técnicas y metodologías matemáticas, pero que finalmente se basa en la frecuencia u ocurrencia de términos dentro de los documentos. Por ejemplo, los tesauros generados con este paradigma agrupan términos que quizás se manejen como sinónimos porque aparecen en documentos de temas similares y no porque estén relacionados semánticamente. Las técnicas estadísticas tienen la limitante computacional para el análisis de información inmensa (en el orden de terabytes) y para obtener resultados de calidad en búsqueda de información.

El paradigma lingüístico propone soluciones basadas en el conocimiento del lenguaje. Algunos ejemplos son el uso de palabras clave asociadas con el tema de un área del conocimiento, la construcción de tesauros especializados en diversos niveles de abstracción.

El paradigma bibliotecológico está enfocado a proporcionar métodos, técnicas y reglas lógicos para organizar la información. Aportando ideas como normalización, catálogos de autoridades (temas, países, instituciones, dependencias, editores, organismos, etc.) y tesauros. La normalización la definimos como el conjunto de pasos necesarios para implantar sistemas homogéneos y eliminar los heterogéneos (por ejemplo en interfaces de búsqueda, lenguajes para representar información, presentación de información, metadatos, protocolos de intercambio de información, contenidos, etc.) [21].

El paradigma de inteligencia artificial se divide en 3 áreas: redes neuronales, sistemas basados en conocimiento y procesamiento de lenguaje natural. Las redes neuronales artificiales son modelos ma-

temáticos inspirados en la anatomía y fisiología del cerebro humano que permiten hacer computación inteligente realizando procesamiento masivo de datos gracias a su infraestructura paralela y distribuida [25]. Esta área de la inteligencia artificial está relacionada también con temas de estudio de la inteligencia artificial como aprendizaje de máquinas y minería de datos. Los sistemas basados en conocimiento utilizan técnicas computacionales nuevas y herramientas sofisticadas que apoyen a los humanos a almacenar y extraer información útil (conocimiento) de volúmenes de datos inmensos (en el orden de terabytes). Estos sistemas tienen una relación muy fuerte con bases de datos, minería de textos, aprendizaje de máquinas y redes neuronales. El procesamiento de lenguaje natural combina técnicas computacionales con lingüística para procesamiento de texto con el fin de hacer transformaciones de un idioma a otro, introducir texto por voz a una computadora, que un robot entienda instrucciones de un humano y finalmente para el búsqueda y recuperación de información.

5. Software libre para implantación de repositorios de información digital

En la actualidad existe un mínimo de 15 herramientas diferentes, de software libre, para implantación de repositorios de información digital. En el presente trabajo únicamente se presentarán Dspace[22], Fedora [23] y Greenstone [24] tomando en consideración su amplia difusión, soporte técnico, documentación y en especial porque utilizan los métodos de indexación descritos en la sección 1. Estas herramientas aceptan documentos digitales en una gran variedad de formatos tales como Word, Powerpoint, Excel, Postscript, PDF, HTML, XML, GIF, JPEG, TIFF, MP3 y MPEG.

Dspace es el software más difundido y de mayor uso en la construcción de repositorios digitales. Tiene un fuerte respaldo en soporte, documentación y está en constante desarrollo. Está programado en Java, esto le ha permitido una gran portabilidad para funcionar en cualquier sistema operativo y arquitectura de cómputo. Lucene forma parte de su núcleo y utiliza sistemas de administración de bases de datos como Postgres y Oracle para el almacenamiento de metadatos y

documentos digitales. Aunque maneja Dublín Core por defecto, puede configurarse para aceptar otro sistema de metadatos como MARC 21. Para el intercambio de información usa el estándar OAI-PMH. Provee una interfaz de usuario, que funciona en WEB, que permite administrar colecciones, definir niveles de acceso a la información y crear usuarios con diferentes permisos para la administración, búsqueda y acceso a los recursos digitales. Se basa en el modelo de referencia OAIS (Open Archive Information System) para la preservación y conservación digital, incorporando algunas estrategias tales como la autenticidad y replicación.

Fedora es un software con una gran simplicidad, para instalar y usar, comparado con otras herramientas de software similares. Tiene un fuerte respaldo en soporte, documentación y está en constante desarrollo. Está programado en Java, esto le ha permitido una gran portabilidad para funcionar en cualquier sistema operativo y arquitectura de cómputo. Lucene forma parte de su núcleo y utiliza sistemas de administración de bases de datos como McKoi, MySQL, Postgres y Oracle para el almacenamiento de metadatos y documentos digitales. Su sistema de metadatos es FOXML, Dublín Core y aparentemente, puede configurarse para aceptar otro sistema de metadatos. Para el intercambio de información usa el estándar OAI-PMH y Z39.50. Provee una interfaz de usuario que permite administrar colecciones. Se han desarrollado otras herramientas para ampliar su funcionalidad y definir niveles de acceso a la información y crear usuarios con diferentes permisos para la administración, búsqueda y acceso a los recursos digitales. Ofrece un conjunto de servicios WEB para proveer, a otros programas de aplicación, acceso a la información. Incorpora algunas estrategias de preservación digital como autenticidad, replicación y manejo de versiones.

Greenstone es un software que ofrece una gran funcionalidad, tiene un fuerte respaldo en soporte, documentación y está en constante desarrollo. Está programado en varios lenguajes pero principalmente en C estándar y Perl, esto le ha permitido una gran portabilidad para funcionar en cualquier sistema operativo y arquitectura de cómputo. Managing Gigabytes y Zebra forman parte de su núcleo. Cuenta con un sistema de administración de bases de datos propio

(GDBM) para el almacenamiento de metadatos y documentos digitales. Provee Dublín Core por defecto, pero tiene conectores con una gran variedad de metadatos como MARC, XML, METS, CDS/ISIS, etc. Para el intercambio de información usa el estándar OAI-PMH y Z39.50. Provee una interfaz de usuario, que funciona en WEB, que permite administrar colecciones, definir niveles de acceso a la información y crear usuarios con diferentes permisos para la administración, búsqueda y acceso a los recursos digitales. Provee algunas estrategias para la preservación y conservación digital tales como reformateo, rejuvenecimiento y replicación, incorporando algunas herramientas de software para este fin.

Referencias

[1] G. Salton, M.J. McGill. Introduction to Modern Information Retrieval. Mc Graw Hill, New York, 1983.

[2] C. Faloutsos. Access Methods for Text. ACM Computer Surveys 17, 1, March 1995.

[3] Dumais S., Furnas G., Landeaur T., Deerwester S., Using Latent Semantic Indexing to improve information retrieval. ACM SIGCHI Conference, 1988.

[4] C. Aggarwal. On effective conceptual indexing and similarity search in text data. In *IEEE ICDM,* 2001.

[5] http://lucene.apache.org

[6] http://www.indexdata.dk/zebra/

[7] http://www.cs.mu.oz.au/mg/

[8] http://www.zilverline.org

[9] http://sourceforge.net/projects/lius/

[10] http://regain.sourceforge.net/

[11] Gospodnetic O., Hatcher E., Lucene in action, Diciembre 2004.

[12] Hammer S., Dickmeiss A., Levanto H., Taylor M., Zebra-User´s Guide and Reference, 2005.

[13] Ian H. Witten, Alistair Moffat, Timothy C. Bell, *Managing Gigabytes, Compressing and Indexing Documents and Images,* Second Edition, Morgan Kaufman Publishers, Inc., Springer 1999.

[14] J. Lu and J. Callan. Pruning long documents for distributed information retrieval. In *ACM CIKM,* pages 332–339, 2002.

[15] http://snowball.tartarus.org/spanish/stemmer.html

[16] Ángel F. Zazo, Carlos G. Figuerola, José L. Berrocal, Emilio Rodríguez, Raquel Gómez. *Experiments in Term Expansion Using Thesauri in Spanish*. Grupo de Recuperación Automatizada de la Información (REINA). Depto. de Informática y Automática - Universidad de Salamanca, España. 2003. http://reina.usal.es

[17] http://www.cs.waikato.ac.nz/ml/weka/

[18] http://yale.cs.uni-dortmund.de/

[19] http://www.kdnuggets.com/

[20] http://www1.ics.uci.edu

[21] Castro T. A., García C. E., Ortiz A. D., *Propuesta para interconexión de catálogos,* XVI Asamblea general de ISTEC, UTPL, Loja, Ecuador, 2007.

[22] http://www.dspace.org

[23] http://www.fedora.info

[24] http://greenstone.org

[25] Trejo A. M. C., González A. J. G., Data SOMinning Software para el Descubrimiento de Conocimiento en Grandes Bases de Datos de Información Científico Tecnológica. Tesis de Licenciatura en Ciencias de la Computación, Facultad de Ciencias, UNAM, 2006.

[26] Han J., Lamber M., *Data Mining Concepts and Techniques*, Second Edition, Morgan Kaufman.