Facultad de Ingeniería
Carrera de Ingeniería Informática

Implantación de un Sistema para la Recuperación de Información en la Empresa de Telecomunicaciones de Cuba S.A. (ETECSA)

Autor:
Jorge Luis Armenteros Villegas

Tutor:

Msc. Denis Morejón López

Msc. Jorge Luis Rivero Pérez

CIENFUEGOS, CUBA
CURSO 2013-2014
Año 56 de la Revolución.

Declaración de autoría

Yo Jorge Luis Armenteros Villegas declaro que soy el único autor de este trabajo de diploma titulado: Implantación de un Sistema para la Recuperación de Información en la Empresa de Telecomunicaciones de Cuba S.A. (ETECSA) y autorizo al Departamento de Informática de la Facultad de Ingeniería en la Universidad de Cienfuegos “Carlos Rafael Rodríguez”, para que hagan el uso que estimen pertinente con el mismo.

Para que así conste firmo (firmamos) la presente a los 11 días del mes de junio del 2014.

_____

Firma Autor

Jorge Luis Armenteros Villegas

Los firmantes abajo certificamos que el presente trabajo ha sido revisado según acuerdo de la dirección de nuestro centro y el mismo cumple los requisitos que debe tener un trabajo de esta envergadura referente a la temática señalada.

_____	_____
Firma Tutor	Firma Tutor
Msc. Jorge Luis Rivero Pérez	Msc. Denis Morejón López
.	
_____	_____
Firma ICT	Firma Vicedecano

Opinión del usuario

El Trabajo de Diploma, titulado Implantación de un Sistema para la Recuperación de Información en la Empresa de Telecomunicaciones de Cuba S.A. (ETECSA), fue realizado en nuestra universidad Carlos Rafael Rodríguez. Se considera que, en correspondencia con los objetivos trazados, el trabajo realizado nos satisface:

- ☐ Totalmente
- ☐ Parcialmente en un ____ %

Los resultados de este Trabajo de Diploma le reportan a nuestra entidad los beneficios siguientes (cuantificar):

_____

_____

_____

_____

_____

Como resultado de la implantación de este trabajo se reporta un efecto económico que asciende a <valor> MN y/o <valor> CUC. (Este valor debe ser REAL, no indica lo que se reportará, sino lo que reporta a la entidad. Puede desglosarse por conceptos, tales como: cuanto cuesta un software análogo en el mercado internacional, valor de los materiales que se ahorran por la existencia del software, valor anual del (de los) salario(s) equivalente al tiempo que se ahorra por la existencia del software).

Y para que así conste, se firma la presente a los __ días del mes de ____ del año ____.

_____ _____

Nombre del representante de la entidad Cargo

_____

Firma Cuño

Agradecimientos

Quisiera agradecer a todos los que de una forma u otra han contribuido con su ayuda y sin los cuales no hubiese sido posible la realización de esta investigación, en especial:

A mis padres por la confianza que depositaron en mí, por su apoyo y su amor.

A mi abuelos que los quiero mucho y siempre me cuida y me aconsejan.

A mi novia que amo y su familia que me quiere y me han tomado como un hijo.

A mi familia que siempre me ha respaldado y apoyado.

A mis compañeros de estudios por brindarme su amistad.

A Yanedky y Yosdeny por ser mis amigos incondicionales y brindarme su espíritu de estudio para cada prueba o trabajo.

A Jorge Luis Rivero (chicho) mi tutor por tener paciencia y ayudar en la confección de esta investigación.

A la Empresa de ETECSA en especial a Denis e Ignacio por hacer posible este trabajo de investigación.

A todos muchas gracias por confiar en mí.

Dedicatoria

A mis padres.
A mi familia.
A mi novia y su familia que es también la mía.
Y a todas las personas que quiero y que me quieren.

Resumen

Para garantizar la calidad del trabajo y los servicios que brinda la Empresa de Telecomunicaciones en Cuba (ETECSA), se han desarrollado un grupo de sitios web que gestionan la información generada en la empresa. El volumen de información crece exponencialmente, por lo que se hace muy compleja la búsqueda y recuperación de información en la misma. Para revertir esta situación, se implantó y evaluó un Sistema de Recuperación de Información, utilizando herramientas de software libre, que permitió optimizar la búsqueda y recuperación de la información generada en la institución. Para la implantación de la solución informática, se realizó un análisis de las soluciones similares existentes, el cual permitió la selección de las herramientas y algoritmos que más se adecuan a la solución deseada. Luego de desplegada la solución en la empresa, se realizó una prueba piloto al sistema en la cual se tomó una muestra representativa de sitios web, obteniéndose resultados alentadores en su utilización práctica. En esta investigación se detalla la implantación y evaluación de la solución propuesta y su impacto para la institución.

Índice General

Índice General

Índice de Imágenes

Índice de Tablas

Introducción.

Desde su surgimiento, el Internet ha desarrollado un gran número de servicios para facilitar el intercambio de información, los que han ido evolucionando en calidad y aceptación para satisfacer necesidades y expectativas del usuario; uno de estos servicios fue creado por Tim Berners-Lee[1] con el término *World Wide Web (WWW)*[2].[1] Es en ese momento que la WWW evoluciona hasta convertirse en el primero de los servicios que ofrece la red de redes. En esa época, se produce la aparición del Internet comercial, seguido, las empresas hacen su aparición en Internet ofreciendo todo tipo de servicios en línea: tiendas, bancos, por nombrar algunos de ellos.[2]

En la actualidad, el crecimiento de la información presente en Internet, se comporta de manera exponencial la cual se almacena de forma distribuida por el aumento de tamaño de las colecciones de documentos y del número de peticiones que se realizan contra estas hace casi imposible que un sistema de búsqueda real se asiente sobre una única máquina. Por ello el concepto de Recuperación de Información Distribuida ha tomado especial relevancia en los últimos años. Se puede considerar Recuperación de Información Distribuida en cierta manera, como si se tratara de Recuperación de Información Paralela en máquinas MIMD[3] que tienen una red de interconexión lenta. Cada una de las máquinas luego puede soportar algún tipo de procesamiento paralelo de forma individual. Por eso interesa que la comunicación entre máquinas sea la menor posible.[3][4][5]

---

[1] SIR TIMOTHY "TIM" JOHN BERNERS-LEE científico de la computación británico, conocido por ser el padre de la Web.

[2] *World Wide Web* (**WWW**) o Red informática mundial, conocida como web, la cual es un sistema de distribución de documentos de hipertexto o hipermedios interconectados y accesibles vía Internet.

[3] **MIMD** (del inglés ***M**ultiple **I**nstruction, **M**ultiple **D**ata)* técnica empleada para lograr paralelismo. Las máquinas que usan MIMD tienen un número de procesadores que funcionan de manera asíncrona e independiente.

Cuba no está exenta de esta regla, el volumen de información que se genera cada día va en aumento, por lo cual en muchos casos este crecimiento acelerado de la cantidad de información a la que tenemos acceso puede desorientarnos. Por lo mismo la búsqueda y recuperación de la información en Internet e Intranet corporativa tiene cada vez más impacto en el desarrollo del conocimiento de ella. La información ha sido considerada por muchos como un activo intangible que adquiere cada vez más valor en cada una de las organizaciones. Es dicha información la que permite, si es bien gestionada, aportar a la toma de decisiones de una empresa, aumentando la productividad de la misma.[6]

Los Sistemas de Recuperación de Información, en lo adelante SRI, constituyen el mecanismo ideal para resolver este tipo de problemas; los que permiten localizar y procesar la información de forma rápida y en forma automática. Son sistemas capaces de localizar cualquier contenido existente en la web, como pueden ser textos, imágenes, videos, archivos de sonido, entre otros. En este sentido destacan los directorios temáticos, los motores de búsqueda o buscadores y los metabuscadores.

La mayoría los sistemas de recuperación de información comparten una misma arquitectura:[7]

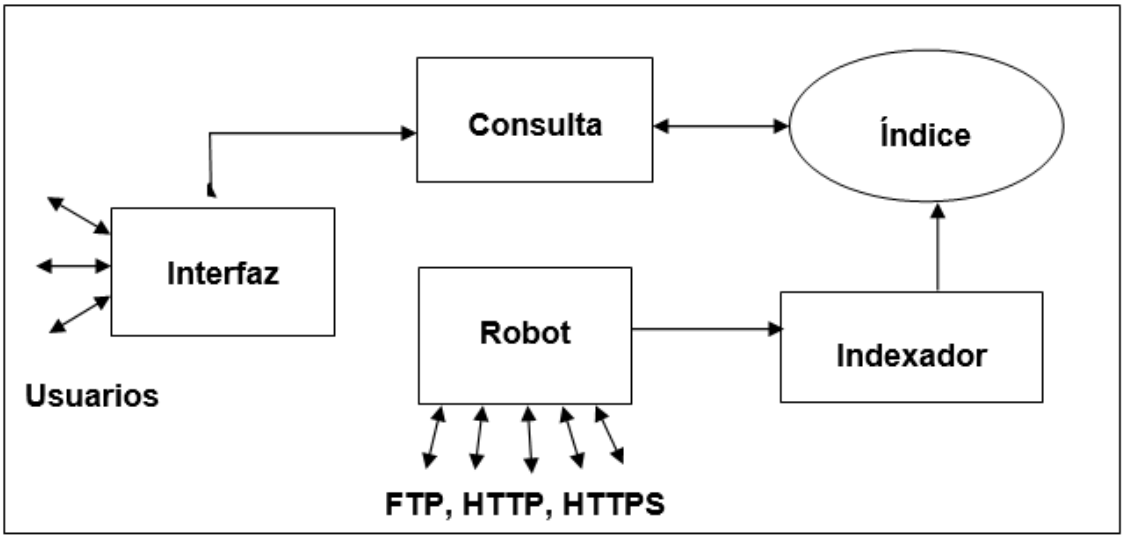


Figura 1: Arquitectura de los sistemas de recuperación de información.

Interfaz[4]**:** un usuario con necesidades de información bien definidas, interactúa con la interfaz del sistema, mediante la cual introduce las consultas al mismo. La interfaz puede estar basada en una interfaz web que es la más común, una interfaz de escritorio o ambas.

Sistema de Formulación de Consultas: realiza un preprocesamiento de las consultas trasladando las consultas hechas en lenguaje natural a consultas entendibles por los sistemas de información.

Mecanismo de evaluación de consultas: compara los documentos representados en el sistema de información, con la consulta preprocesada para obtener un subconjunto de documentos relevantes que satisfagan la consulta introducida por el usuario, ordenados estos de acuerdo a un criterio de relevancia.

[4] Interfaz se conoce en inglés como *interface* (“superficie de contacto”). En informática se utiliza para nombrar a la conexión física y funcional entre dos sistemas o dispositivos de cualquier tipo dando una comunicación entre distintos niveles.

En estos momentos no se cuenta con un sistema similar que permita optimizar la búsqueda y recuperación de la información existente en la Empresa de Telecomunicaciones de Cuba S.A (ETECSA) la cual cada año va perfeccionándose y adquiriendo cierto grado de madurez.

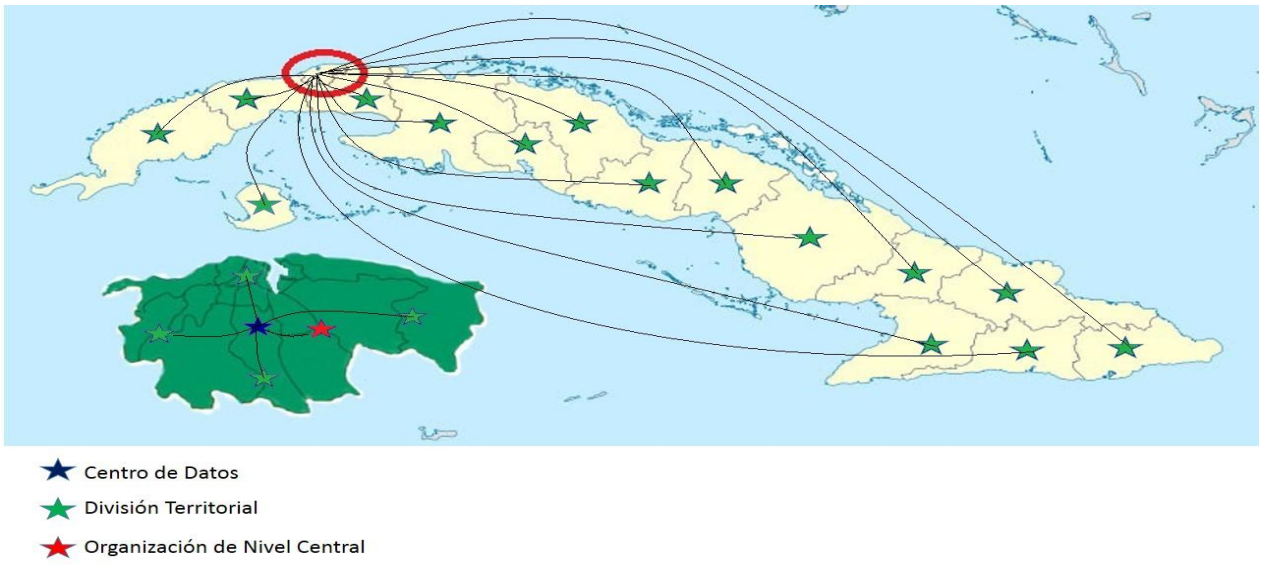


Figura 2: Red de la Empresa de Telecomunicaciones de Cuba (ETECSA).

El país cuenta con quince Divisiones Territoriales una por cada provincia incluyendo al municipio especial Isla de la Juventud y sin contar a Ciudad Habana, en dicha provincia existe una organización de nivel central que es donde se encuentran todos los directivos nacionales, un Centro de Datos para la administración de las redes y cuatro Divisiones Territoriales conocidas como Norte, Sur, Este, Oeste. La empresa muestra una topología de red en forma de estrella. Cada División Territorial contiene tres subredes conocidas como Red GESNET, Red DMZ y Red Corporativa. El conjunto de

las redes corporativas conforman la red WAN[5] o Intranet corporativa. Se calcula que existan aproximadamente 400 usuarios por cada División Territorial como promedio, contando con un ancho de banda mínimo de 10 MB/s. En cada provincia se cuenta con al menos dos sitios HTTP[6], y dos FTP[7] sin detallar Ciudad Habana que presenta aproximadamente diez sitios HTTP y FTP. En total existen más de 40 sitos HTTP y 40 sitios FTP de forma aproximada. Estos contienen información de todo tipo como legislaciones, indicaciones, procedimientos, manuales técnicos, noticias, artículos de investigación, o instalaciones de programas que pueden ser necesarios con determinada urgencia. Por tanto es necesario la implementación de un Sistema de Recuperación de Información (SRI) que facilite la recuperación de documentos y archivos en general.

Por lo planteado con anterioridad se puede identificar como situación problémica: Resulta complicado buscar determinada información en la red nacional de ETECSA debido a la existencia de cambios de URL en reiteradas ocasiones, falta de conocimiento de las direcciones de los sitios HTTP y FTP lo que conlleva a iterar por cada URL debido mala organización de la información. Se puede citar como ejemplo a la Asesora Jurídica que en varias ocasiones necesita buscar resoluciones, indicaciones de la empresa, también el Ejecutivo Comercial necesita buscar información actual sobre procedimientos comerciales entre otros documentos o el Administrador de Red se encuentra necesitado de instalaciones, actualizaciones, manuales técnicos.

---

[5] Una red de área amplia, o **WAN**, por las siglas de (***w**ide **a**rea **n**etwork* en inglés), es una red de computadoras que abarca varias ubicaciones físicas, proveyendo servicio a una zona, un país, incluso varios continentes.

[6] Hypertext Transfer Protocol o **HTTP** (en español *protocolo de transferencia de hipertexto*) es el protocolo usado en cada transacción de la World Wide Web.

[7] **FTP** (siglas en inglés de ***F**ile **T**ransfer **P**rotocol*, 'Protocolo de Transferencia de Archivos') en informática, es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red.

Por lo planteado con anterioridad se puede identificar como **situación problémica** que: Resulta complicado buscar determinada información en la red nacional de ETECSA debido a la existencia de cambios de URL en reiteradas ocasiones, falta de conocimiento de las direcciones de los sitios HTTP y FTP lo que conlleva a iterar por cada URL debido mala organización de la información. Se puede citar como ejemplo a la Asesora Jurídica que en varias ocasiones necesita buscar resoluciones, indicaciones de la empresa, también el Ejecutivo Comercial necesita buscar información actual sobre procedimientos comerciales entre otros documentos o el Administrador de Red se encuentra necesitado de instalaciones, actualizaciones, manuales técnicos.

Atendiendo a esta situación problémica se puede definir como **problema científico**: ¿Cómo implantar un Sistema de Recuperación de Información la red WAN de ETECSA?

En consecuencia el **objeto de estudio** de la presente investigación es**:** La recuperación de información a partir de sistemas informáticos de software libre.

Teniendo como **campo de acción**: La recuperación de Información a partir de sistemas informáticos de software libre en la red WAN de ETECSA.

Como **objetivo general** se plantea**:** Implantar un sistema de recuperación de información en la red WAN de ETECSA para facilitar la búsqueda de información.

El objetivo general se dividió en los siguientes **objetivos específicos**:

1. Estudiar los sistemas de recuperación de información atendiendo a los requerimientos de la empresa y las metodologías para su evaluación.
2. Caracterizar los algoritmos y métodos utilizados por cada sistema de recuperación de información antes seleccionados.
3. Realizar pruebas en escenarios reales que validen el estudio antes realizado.

Las **tareas científicas** realizadas para cumplir con los objetivos son:

1. Estudio que permita conocer los requerimientos de la empresa.
2. Análisis de cada uno de los Sistemas de Recuperación de Información atendiendo a los requerimientos de saturación del canal de comunicación, la

indexación de sitios HTTP, FTP y la búsqueda por el protocolo HTTPS en la empresa.

3. Selección de los sistemas de recuperación de información que cumplan con los requerimientos ante señalados.
4. Análisis de los algoritmos y métodos utilizados por los sistemas de recuperación de Información.
5. Implantación en escenarios reales los sistemas de recuperación de información seleccionados.
6. Pruebas en escenarios reales de los sistemas de recuperación de información.
7. Evaluación de los sistemas utilizando las metodologías estudiadas.

Todas estas tareas fueron trazadas con miras a la siguiente **idea a defender**: El sistema implantado, facilitará la búsqueda de información en la red WAN de la Empresa de Telecomunicaciones ETECSA.

La implantación del sistema brinda el siguiente **aporte práctico**: El mejoramiento de la recuperación de información en la red de ETECSA a partir de la implantación de un sistema de recuperación de información.

El trabajo está estructurado de la siguiente forma: Introducción, 3 Capítulos, Conclusiones, Recomendaciones y Referencias Bibliográficas donde se presenta la siguiente información:

Capítulo I: El primer capítulo, titulado “Fundamentación Teórica de la Recuperación de Información”, se hace una introducción al concepto de Recuperación de Información, la evolución que ha tenido a través de los años, se introducen las técnicas automáticas de recuperación de información y su importancia en la actualidad, variantes de la recuperación de la información, los modelos que existen y cómo funcionan.

Capítulo II: El segundo capítulo tiene como título: “Sistemas de Recuperación de Información”, se define que es un Sistema de Recuperación de Información, como funciona, sus partes, la evolución, los modelos existentes y se realiza una comparación

funcional teniendo como principales aspectos los requerimientos planteados por la empresa.

Capítulo III: El tercer capítulo, titulado “Instalación y evaluación de los sistemas de recuperación de información seleccionados.”, en él se detallan pasos a seguir para la instalación de los sistemas seleccionados que cumplen en gran medida con los requerimientos planteados por la empresa, se realiza un análisis de las ventajas que proporcionan cada uno de ellos y se aplica un método de evaluación para definir cuál de ellos es el idóneo para la empresa.

Capítulo I: Fundamentación Teórica de la Recuperación de Información.

Este capítulo es una presentación del concepto de recuperación de información, y del conjunto de diferencias que posee con otras aplicaciones de la Informática en lo relacionado con la gestión y recuperación de datos. Al mismo tiempo se exponen los distintos modelos sobre los que se basan los sistemas que permiten la recuperación de información.

1.1– ¿Qué es la recuperación de información?

La Recuperación de Información, llamada en inglés *Information Retrieval* (IR), es la ciencia de la búsqueda de información en documentos electrónicos y cualquier tipo de colección documental digital, encargada de la búsqueda dentro de éstos mismos, búsqueda de metadatos[8] que describan documentos, o también la búsqueda en bases de datos relacionales, ya sea a través de internet, intranet, y como objetivo realiza la recuperación en textos, imágenes, sonido o datos de otras características, de manera pertinente y relevante.[8][9][10]

Según [11] la recuperación de la información “se trata de una disciplina que involucra la localización de una determinada información dentro de un almacén de información o base de datos”. El autor, implícitamente, establece que el concepto de recuperación de información se encuentra asociado con el concepto de *selectividad*, ya que la información específica ha de extraerse siguiendo algún tipo de criterio discriminatorio (selectivo por tanto).

Desde un punto de vista práctico, dada una necesidad de información del usuario, un proceso de IR produce como salida un conjunto de documentos cuyo contenido

[8] Metadatos, literalmente «sobre datos», son datos que describen otros datos. En general, un grupo de metadatos se refiere a un grupo de datos.

satisface potencialmente dicha necesidad. Esta última puntualización es de suma importancia, ya que la función de un sistema de IR no es la de devolver la información deseada por el usuario, sino únicamente la de indicar qué documentos son potencialmente relevantes para dicha necesidad de información.[12]

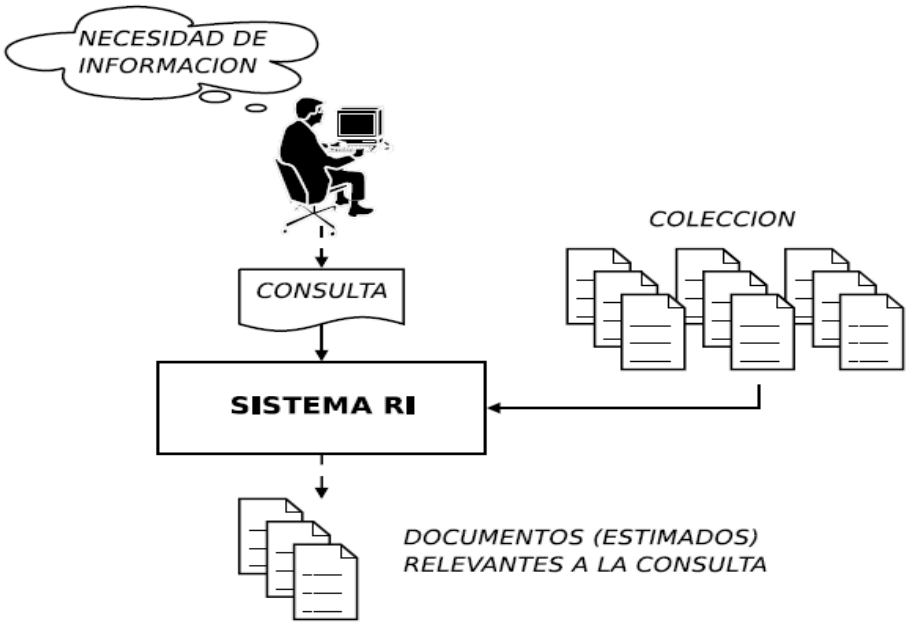


Figura 3: Recuperación de información.

1.2 – Evolución del significado del término.

Tradicionalmente se limitaba a la recuperación de documentos escritos, el término se redefinió para incorporar la creciente aparición de materiales multimedia. De esta forma los nuevos buscadores de información en Internet, que originariamente buscaban textos, expandieron su actividad a imágenes, videos o audios. De esta forma términos como Recuperación de textos, recuperación documental y recuperación de información son utilizados como equivalentes.[13]

Por otro lado, la necesidad de localizar datos concretos ha ido expandiendo su área de actuación. En la actualidad se está migrando desde la recuperación de documentos a la

recuperación pregunta-respuesta, que responden con el dato concreto y no con el conjunto de documentos que posiblemente contenga este dato.[14]

1.3 – Técnicas automáticas de recuperación de información.

Según [15] la frase *técnicas automáticas* se hace referencia al conjunto de procedimientos y recursos que se aplican para que el sistema explote capacidades que el usuario no posee, lo alivie de las tareas rutinarias y trabajosas, o complemente y amplíe sus capacidades. Muchas de las cosas que se tratan en las bibliografías sobre RI son técnicas; por ejemplo, la base de datos es un recurso de almacenamiento y recuperación de información que supera ampliamente a la memoria humana.

Una interfaz de búsqueda gráfica es un procedimiento que permite el acceso temático a una colección de miles de documentos. Todas estas técnicas son empleadas para lograr una interacción exitosa entre un usuario, que puede ser humano o máquina, con cierta necesidad informativa; y una masa de información variada, registrada en documentos digitales o no, susceptible de satisfacer dicha necesidad y que puede o no haber sido sometida a algún proceso de descripción previo.[16]

Sin embargo, en este trabajo se acota el concepto de técnicas automáticas de recuperación a la clasificación propuesta por Spärck Jones[9], quien sostiene, no sin cierta dificultad, que las técnicas se pueden agrupar en:

- Técnicas de indexación
- Técnicas de búsqueda

Las técnicas de indexación tienen que ver con la construcción de la representación del documento y de la representación de la necesidad de información del usuario, en el sistema de recuperación. Mientras que las técnicas de búsqueda, tienen que ver con la

[9] KAREN SPÄRCK JONES, científica británica especializada en lingüística computacional. Investigadora pionera en recuperación de información.

manera en que el archivo de documentos es examinado y los ítems son extraídos de acuerdo a la interrogación que se formuló.

1.4 – Modelos de recuperación de Información.

Los modelos de recuperación planteado por los autores en el libro [9] tratan de calcular el grado en que un elemento de información responde a determinada consulta. Los tres modelos más utilizados son:

- **Booleano:** se crea un conjunto con los elementos de la consulta y otro con los documentos, y se mide la correspondencia.
- **Vectorial:** en el que la consulta y los términos del documento se representan mediante dos vectores, y se mide el grado en que ambos vectores divergen.
- **Probabilístico:** se calcula la probabilidad en que el documento responde a la consulta y se utiliza la retroalimentación, la cual se basa en que el usuario indique que documentos se parecen más a su respuesta idónea, para así reformular la consulta.

1.4.1– Modelo Booleano.

El modelo booleano es el más sencillo de los tres aquí descritos, y se basa en la teoría de conjuntos y el álgebra de Boole. En este modelo inicial el usuario especifica en su consulta una expresión booleana formada por una serie de términos ligados mediante operadores booleanos comúnmente and, or y not. Dada la expresión lógica de la consulta, el sistema devolverá aquellos documentos que la satisfacen y que conformarán el conjunto de documentos relevantes. De esta forma, el sistema simplemente particional los documentos de la colección en dos conjuntos, aquéllos que cumplen la condición especificada (relevantes), y aquéllos que no la cumplen (no relevantes), sin ordenación interna alguna, de forma similar a lo que ocurriría con una base de datos tradicional. Un documento es, por tanto, simplemente relevante o no.

Capítulo I: Fundamentación Teórica de la Recuperación de Información.

La popularidad del modelo booleano, sobre todo en sus inicios, viene dada por su sencillez tanto a nivel conceptual, por la claridad de sus formalismos, como a nivel de implementación. Además, puesto que las consultas son formuladas a modo de expresiones booleanas, de semántica sumamente precisa, el usuario sabe porque un documento ha sido devuelto por el sistema, lo que no siempre ocurre en otros modelos más complejos.

Sin embargo, existen también una serie de desventajas importantes asociadas al Modelo Booleano. La primera de ellas viene dada por la dificultad que conlleva la formalización de la necesidad de información del usuario en forma de expresión booleana, sobre todo cuando se trata de usuarios inexpertos y de necesidades complejas. A ello se suma el hecho de que ligeros cambios en la formulación pueden dar lugar a cambios considerables en el conjunto respuesta.

Otro de los grandes inconvenientes del modelo booleano viene dado por su propia naturaleza, de carácter binario. De esta forma, dada una consulta, un documento simplemente es o no relevante dependiendo de si cumple la condición expresada por la consulta. Por lo tanto, no existen ni el concepto de correspondencia parcial ni el concepto de gradación de relevancia. Al no permitir correspondencias parciales, el sistema podría no devolver documentos que, aun siendo relevantes, no verificasen por completo la condición estipulada. Del mismo modo, todos los términos de la consulta tienen la misma importancia, cuando es lógico pensar que la semántica de un texto dado se concentre en mayor grado en ciertos términos. Por otra parte, al no existir ninguna ordenación por relevancia, el usuario se ve obligado a examinar la totalidad del conjunto resultado devuelto.

En la actualidad el modelo booleano se encuentra desplazado dentro de los grandes sistemas de Recuperación de Información frente a los restantes modelos a causa de sus desventajas. Sin embargo, continúa empleándose en ciertos ámbitos donde se precisan correspondencias exactas, como en el caso de algunos sistemas de información legislativa.[1]

1.4.2– Modelo Vectorial

Para dar solución a los problemas planteados por el modelo booleano, el modelo vectorial sugiere un marco formal diferente en el que se permite tanto la asignación de correspondencias parciales, como la existencia de grados de relevancia en base a los pesos de los términos en consultas y documentos. El modelo vectorial no se limita, pues, a comprobar si los términos especificados en la consulta están o no presentes en el documento, como en el caso del modelo booleano, sino que la similitud entre ambos se calculan en base a los pesos de los términos involucrados, permitiendo de este modo, por un lado, la existencia de correspondencias parciales, y por otro, el cálculo de grados de similaridad o relevancia conforme a los cuales los documentos pueden ser devueltos por orden de mayor a menor relevancia, facilitando notablemente el trabajo del usuario, que puede concentrar sus esfuerzos en los primeros documentos devueltos, aquellos más relevantes o incluso definir umbrales de relevancia por debajo de los cuales un documento no es tenido en consideración.

Los buenos resultados obtenidos con el modelo vectorial, unidos a la simplicidad a nivel de concepto e implementación, su bondad a la hora de aceptar consultas en lenguaje natural, y su capacidad para permitir correspondencias parciales y ordenamiento por relevancia, han hecho de este modelo una de las principales bases sobre la que se han desarrollado gran parte de los experimentos y sistemas en todo el ámbito de la Recuperación de Información. Sus buenas características, unidas al hecho de que sea uno de los modelos de representación más utilizados, le han convertido frecuentemente en el sistema de referencia respecto al cual comparar resultados a la hora de desarrollar nuevos modelos de recuperación.[9][1][17]

1.4.3– Modelo Probabilístico

Frente al modelo booleano, basado en teoría de conjuntos y el modelo vectorial, de carácter algebraico, el modelo probabilístico formaliza el proceso de recuperación en términos de teoría de probabilidades. Las bases del modelo probabilístico fueron

establecidas por Robertson[10] y Spärck Jones[11]. El objetivo perseguido en el modelo es el de calcular la probabilidad de que un documento sea relevante para la consulta dado que dicho documento posee ciertas propiedades, propiedades en forma de los términos índice que dicho documento contiene.

Según el principio de orden por probabilidades, el rendimiento óptimo de un sistema se consigue cuando los documentos son ordenados de acuerdo a sus probabilidades de relevancia. En consecuencia, el sistema devolverá los documentos en orden decreciente de las probabilidades de relevancia estimadas mediante el modelo probabilístico.

El modelo parte de las siguientes suposiciones:

- Todo documento es, bien relevante, bien no relevante para la consulta.
- El hecho de juzgar un documento dado como relevante o no relevante no aporta información alguna sobre la posible relevancia o no relevancia de otros documentos (suposición de independencia). [9][1][17]

[10] STEPHEN E. ROBERTSON, programador e informatólogo británico. Junto a KAREN SPÄRCK JONES diseñó el modelo probabilístico de recuperación de información.

Conclusiones parciales del capítulo.

La Recuperación de Información es una disciplina que involucra la localización de una determinada información dentro de un almacén de información o base de datos. Se plantea desde un punto de vista práctico, que dada una necesidad de información del usuario, un proceso de recuperación de información produce como salida un conjunto de documentos cuyo contenido satisface potencialmente dicha necesidad. La función de un sistema de recuperación de indormación no es la de devolver la información deseada por el usuario, sino únicamente la de indicar qué documentos son potencialmente relevantes para dicha necesidad de información. En la actualidad se está migrando desde la recuperación de documentos a la recuperación pregunta-respuesta, que responden con el dato concreto y no con el conjunto de documentos que posiblemente contenga este dato.

Las técnicas automáticas hacen referencia al conjunto de procedimientos y recursos que se aplican para que el sistema explote capacidades que el usuario no posee, lo alivie de las tareas rutinarias y trabajosas, o complemente y amplíe sus capacidades. Dichas técnicas tienen que ver con la construcción de la representación del documento y de la representación de la necesidad de información del usuario, en el sistema de recuperación. Mientras que las técnicas de búsqueda, tienen que ver con la manera en que el archivo de documentos es examinado y los ítems son extraídos de acuerdo a la interrogación que se formuló.

Capítulo II: Sistemas de Recuperación de Información.

2.1 – Modelo general.

Construir un modelo significa elaborar una representación simplificada de una realidad compleja con la finalidad de entenderla. Se utilizan para analizar, describir, explicar o exponer problemas, y su uso está muy difundido en la ciencia moderna.[19]

La construcción de modelos en el área de la Recuperación de Información apareció a mediados de la década del setenta, con la aspiración de brindar una base teórica a la gran cantidad de experimentos y desarrollos que se estaban llevando acabo. Las numerosas técnicas diferentes y combinadas que empleaban, junto con la creciente necesidad de evaluar la efectividad de los sistemas, hacia imprescindible la construcción de patrones más generales en los cuales se pudiera agrupar la variedad de propuestas.[20]

Un modelo sencillo de un proceso llevado a cabo en un SRI puede presentarse por la interacción de 3 elementos:

- La necesidad de información de un usuario expresada en términos de una interrogación al sistema hecha en lenguaje libre o artificial.
- La representación de la información contenida en el sistema realizada en términos de indexación, categorías codificadas, representaciones gráficas.
- La función de equiparación entre la interrogación y la representación de la información con la finalidad de encontrar coincidencias o similaridades.

Se puede visualizar en la siguiente figura los elementos que componen un SRI y sus interacciones:

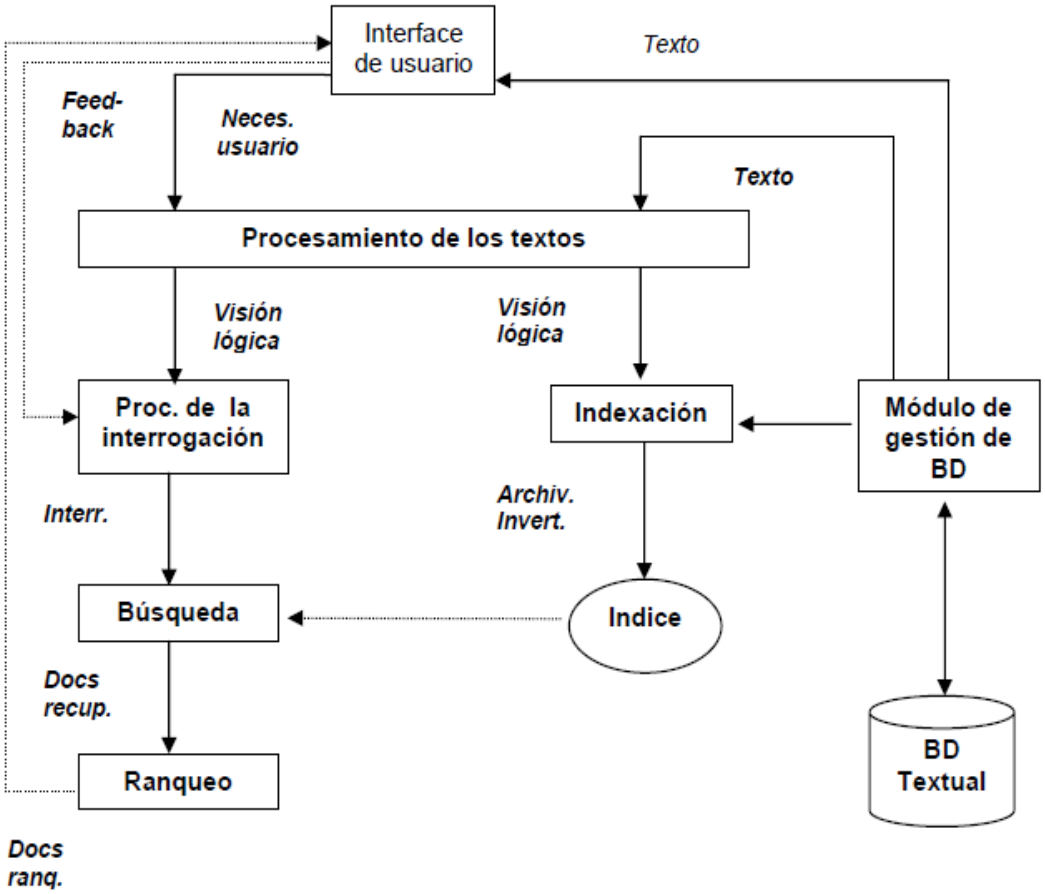


Figura 4: Modelo general.

Partiendo de un repositorio de información se construye una base de datos textual[12]. Para esto, se determinan los textos que se incluirán y las operaciones que se desarrollarán sobre ellos. Esto conformará la visión lógica del texto/documento. Luego que esto ha sido definido, y basándose en ello, el gestor de la base de datos construye el índice. Este será una de las partes centrales del sistema ya que es lo que permite

[12] Los SRI tradicionales están basados en una colección de items de información (documentos o partes de documentos) almacenados en una Base de Datos. Cada item constituye un registro de la base, el cual contendrá los valores de las características o atributos que los identifican. Estas bases se conocen, generalmente, como Bases de Datos Bibliográficas (hacen sólo referencia a los items existentes) y Bases de Datos de Texto completo. Estas bases de datos son creadas, mantenidas y controladas por los llamados Sistemas de Gestión de Bases de Datos.

acelerar los procesos de búsqueda de manera que sean viables en términos de volumen de información/unidad de tiempo. Una de las estructuras de índices más simple y muy usada es el archivo invertido[13].

Con estos elementos disponibles, el usuario puede iniciar un proceso de interacción con el sistema. Su necesidad de información es transformada por las mismas operaciones textuales aplicadas a los textos del repositorio y su necesidad es representada en términos de una interrogación concreta al sistema. Dicha interrogación permite, usando el índice ya generado, recuperar las representaciones de los documentos. Antes de enviárselas al usuario el sistema las ordena por probable relevancia[14]. A partir de allí, el propio usuario puede comenzar un ciclo de ajuste de su interrogación con la finalidad de recuperar los documentos más pertinentes para su requerimiento.[1]

2.2 – Componentes de un sistema de recuperación de información.

Un motor de búsqueda es un sistema de recuperación de información diseñado para la búsqueda de información en la Web[22]. Sus componentes básicos son:

- **Crawler:** es un programa que inspecciona las páginas web de forma metódica y automatizada. Uno de los usos más frecuentes que se les da consiste en crear una copia de todas las páginas web visitadas para ser procesada posteriormente. Las arañas web suelen ser bots[15] (el tipo más usado de éstos). Las arañas web comienzan visitando una lista de URLs, identifica los hiperenlaces en dichas

[13] El archivo invertido contiene los términos de indexación asignados. Cada término tiene asociado la lista de números de referencia a documentos que lo contienen. La recuperación de documentos identificados con algún término requiere que se busque primero en el archivo invertido el término y luego se llegue mediante la lista de referencias a los números de documentos en el archivo general.

[14] BOOKSTEIN [21] define a la relevancia como la relación entre un individuo, en el momento que necesita una información, y el texto que se la provee. Se dice que el texto es relevante para esa persona si ella siente que la necesidad de información que tenía ha sido satisfecha al menos en parte por dicho texto.

[15] El termino *Bot* proviene de robot, es un programa informático, imitando el comportamiento de un humano, puede realizar funciones rutinarias de edición.

páginas y los añade a la lista de URLs a visitar de manera recurrente de acuerdo a determinado conjunto de reglas. Funcionando de la siguiente manera: se le da al programa un grupo de direcciones iniciales, la araña descarga estas direcciones, analiza las páginas y busca enlaces a páginas nuevas. Luego descarga estas páginas nuevas, analiza sus enlaces, y así sucesivamente.
Entre las tareas más comunes de las arañas de la web tenemos:

- ✓ Crear el índice de una máquina de búsqueda.
- ✓ Analizar los enlaces de un sitio para buscar links rotos.
- ✓ Recolectar información de un cierto tipo, como precios de productos para recopilar un catálogo.

- **Indexador:** Encargado de mantener un índice invertido con el contenido de las páginas recorridas por el Crawler.
- **Máquina de consultas:** Encargado de procesar las consultas y buscar en el índice los documentos con mayor similitud a ella.
- **Función de score:** Es la función que tiene la máquina de consulta para computar la similitud entre la consulta y los documentos indexados. La función es usada para rankear los documentos por su similitud con la consulta entregada por un usuario.
- **Interfaz:** Interactúa con el usuario, recibe la consulta como entrada y retorna los documentos rankeados por similitud.

2.3 – Arquitectura de los sistemas de recuperación de información.

2.3.1 – Sistemas de recuperación de información de arquitectura centralizada.

En el libro [23] se plantea que los motores de búsqueda centralizados se basan en una arquitectura de tres componentes principales: recolector, indexador y motor de consulta. Dicha arquitectura se muestra en la figura 2.1.

- **Recolector:** también denominado crawler, spider ó robot. Este componente opera estableciendo conexiones con servidores web y solicitándole documentos. Los nombres y direcciones (URLs) de los documentos que debe recuperar se

encuentran en archivos de datos que se actualizan a medida que se extraen más referencias de los documentos procesados.

- **Indexador:** este módulo implementa mecanismos de pre-procesamiento de documentos y construye las estructuras de datos (en general, variantes de archivos invertidos) que soportarán la búsqueda. En el pre-procesamiento se llevan a cabo tareas de normalización de los documentos (palabras en mayúsculas y minúsculas, acentos, signos de puntuación) y de extracción de términos no relevantes. La actualización de los índices depende de la frecuencia con que un crawler obtenga nuevos documentos ó versiones actualizadas de documentos existentes.
- **Motor de consulta:** es el módulo que interactúa con el usuario aceptando de éste una expresión de consulta para realizar la búsqueda. Con ésta, consultará los índices a los efectos de encontrar los documentos que satisfagan la consulta y con ellos armará la respuesta. Para este último paso, implementará alguna estrategia de clasificación ó rankeo para determinar el orden en que las referencias encontradas serán mostradas al usuario.

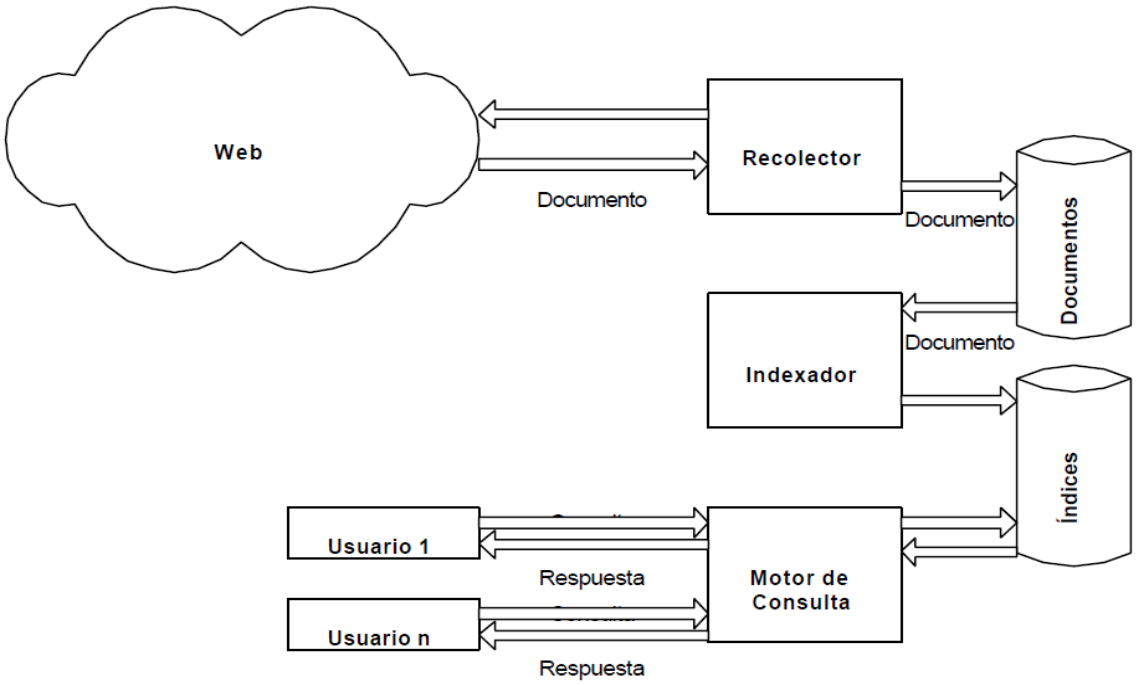


Figura 5: Componentes de un Sistema de Recuperación de Información de arquitectura centralizada.

Uno de los problemas más importantes de esta arquitectura está relacionado con la recolección de los documentos. Esta tarea es la que permite obtener la información para la construcción de los índices. El dinamismo de la web plantea un serio problema en este sentido. El índice de un SRI no puede ser una estructura estática, sino todo lo contrario.

La frecuencia de actualización de los documentos requiere que el crawler constantemente vuelva a solicitar a los servidores web los mismos a los efectos de poder detectar cambios en el contendido, que determinarán cambios en los índices. Además, la recolección de documentos es una tarea que consume una alta cantidad de recursos computacionales y que se encuentra en cierta medida limitada por el tráfico existente en las redes, la disponibilidad de los servidores web y la carga que tengan en el momento que el crawler les solicita documentos.

Por otra parte, se puede plantear como ventajas de las arquitecturas centralizadas la posibilidad de almacenar información acerca de la ubicación y distancia de determinado contenido al usuario que lo busca. Además por tener todas las referencias en un único

índice, se pueden implementar mecanismos de clasificación y rankeo por relevancia que abarcan a todo el contenido, por ejemplo el mecanismo implementado por Google, denominado PageRank, tiene este requerimiento.

2.3.2 – Problemas asociados al enfoque centralizado.

Las características enunciadas de la web establecen restricciones en cuanto a la cantidad y calidad de la información recolectada y a las respuestas entregadas a un usuario ante una solicitud, en especial utilizando los motores de búsqueda tradicionales (basados en una arquitectura centralizada). De dichas características, el tamaño actual del espacio web, su constante crecimiento y la frecuencia de actualización de contenidos generan una serie de problemas importantes, a saber [24]:

- **Aumento de la proporción de respuestas irrelevantes:** Ante una consulta sobre una temática determinada, los motores de búsqueda de propósito general, retornan por su propia naturaleza demasiadas referencias a páginas cuyo contenido no es relevante según la consulta. Esto se debe a la diversidad del espacio web que cubren. Como se mencionó anteriormente, una alternativa para resolver esta problemática la brindan los motores de búsqueda especializados, dado que solo operan en un área temática.
- **Mantenimiento de los índices:** Dados los permanentes cambios e incorporación en los contenidos de los sitios web, la calidad del motor de búsqueda estará condicionada, entre otros factores, a la frecuencia de su esquema de visita y reindexación.

  Actualmente, es una necesidad real que este tiempo sea lo más corto posible. Un estudio realizado por la compañía Search Engine Showdown muestra que el promedio de actualización de los índices de los motores de búsqueda más importantes varía entre 2 semanas y 3 meses.
- **Cobertura limitada:** Solo una fracción de la web es indexada por los motores de búsqueda y en particular la cobertura de un motor es significativamente limitada. Además, existe una parte de la web conocida como la “web invisible” que no es indexada por los motores de búsqueda. En general, se trata de contenido que es

generado dinámicamente a partir de contenido almacenado en bases de datos y en selecciones de los usuarios ó bien, páginas que se desean excluir de los índices por alguna política particular.

- **Alto requerimiento de recursos:** Dada la cantidad de información y usuarios en Internet, se requiere alta cantidad de recursos de hardware como equipos y ancho de banda, tanto para recolectar los documentos a indexar, construir las estructura de datos sobre las que se efectuarán las búsquedas y atender los requerimientos de los usuarios.

Un estudio sobre acceso a información científica en la web en el artículo [3], también expone limitaciones de los motores de búsqueda en cuanto a la cobertura, frecuencia de actualización, debilidades en los algoritmos de rankeo y la flexibilidad de los lenguajes de consulta. Desde el punto de vista de la disponibilidad del servicio, se puede plantear que un motor de búsqueda de arquitectura centralizada tiene un único punto de falla, aunque se implementan técnicas de replicación, balanceo de cargas y tolerancia a fallas para poder manejar la cantidad de información y la cantidad de usuarios existentes en la red.

2.3.3 – Sistemas de recuperación de información de arquitectura distribuida.

Existen variantes en motores de búsqueda de arquitectura distribuida. Cada una de éstas se debe a que su naturaleza distribuida puede darse, en general, en el mecanismo de recolección, en la construcción de los índices o en ambas. En esta arquitectura se plantea la existencia de componentes de recolección de información que operan coordinadamente con agentes de búsqueda para realizar la tarea. (Figura 2.2).

Los recolectores obtienen la información de varios servidores web y pueden llevar a cabo tareas de procesamiento cómo manejar diferentes formatos de documentos y generar la información que se enviará a los agentes. [3][4][5]

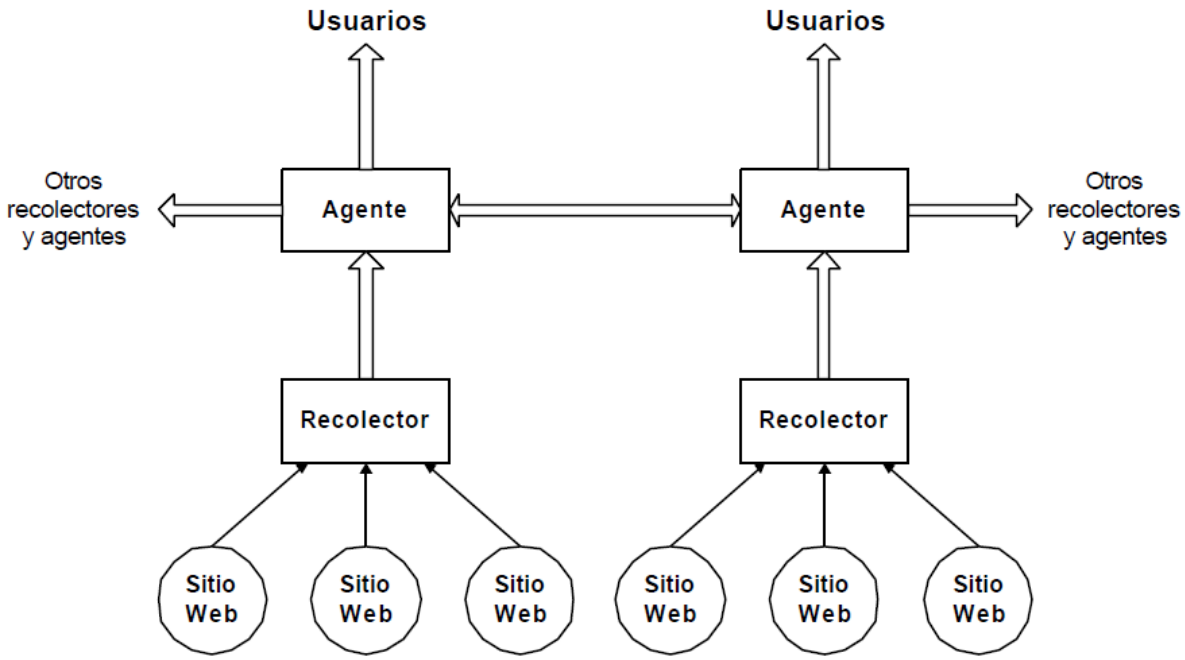


Figura 6: Componentes de un sistema de recuperación de información de arquitectura distribuida.

Los agentes son los encargados de consultar a los recolectores sobre la información obtenida y con ésta construir los índices que soportarán las búsquedas. Además, pueden operar en coordinación con otros agentes a los efectos de intercambiar información. Además, brindan al usuario una interfaz para la especificación de la consulta, realizan la búsqueda sobre las estructura de datos construidas (que almacenan localmente) y entregan los resultados. Además de poder realizar de forma opcional tareas de rankeo o clasificación de respuestas.

Algunas de las ventajas de operar bajo esta arquitectura están dadas por la reducción del tráfico en las redes, ya que se puede situar a los recolectores “cerca” de las fuentes de información (e inclusive junto a éstas). También se evita el trabajo redundante ya que un mismo recolector puede enviar información a diferentes agentes, lo que reduce la carga de trabajo sobre el servidor web ya que se trabaja coordinadamente.

2.4 – Selección de los sistemas de recuperación de información para la evaluación.

Debido a que las pruebas se realizaron en una red sensible en cuanto a la importancia de sus servicios para la empresa y un error podría perjudicar al desempeño de la empresa se realizó un estudio sobre los principales sistemas de recuperación de Información de software libre existentes, con el objetivo de seleccionar el adecuado a los propósitos del proyecto teniendo en cuenta los requerimientos dictados por la empresa.

- Saturación del canal de comunicación.
- Indexación de sitios HTTP y FTP.
- Búsqueda por los protocolos de comunicación HTTP, FTP y HTTPS.

Tabla 1: Selección de los sistemas acordes con los requerimientos de la empresa.

SRI	1	2	3	4
Nutch	X	X	X	
Mnogosearch	X	X	X	X
Lemur	X	X		X
Htdig	X	X		

1_Saturación del canal de comunicación, 2_Indexación de sitios HTTP, 3_Indexación de sitios FTP, 4_Busqueda por el protocolo HTTPS

A continuación se describen los Sistemas de Recuperación de Información más conocidos y utilizados actualmente que cumplen con las normas planteadas anterior mente.

2.4.1– Nutch.

En la documentación del SRI Nutch [25] se plantea que es un programa de código libre[16] diseñado para realizar las tareas de robot de búsqueda, para rastrear la web recuperando las páginas que componen la red a través de la estructura de enlaces existente entre ellas. Nutch crea una base de datos con todos los enlaces encontrados, al tiempo que guarda una copia de todas las páginas localizadas y el resultado del análisis de su contenido, pues incorpora parsers[17] para muchos formatos, no solamente HTML. Sin embargo, las tareas de búsqueda y recuperación de dicha información no están incorporadas, dependiendo del programa Lucene[18] para su indexación. De igual forma, tampoco incluye una interfaz web que facilite la administración y uso del programa, debiéndose instalar el programa Tomcat[19] para dichas tareas.

Sin embargo, la versión Nutch 1.3 ayuda a resolver en buena medida estos inconvenientes. Elimina la dependencia de Tomcat, incluyendo el contenedor de servlets Jetty[20], lo que implica disponer de una consola de administración web. Además, esta versión Nutch 1.3 incorpora comandos para la integración y funcionamiento simultáneo del programa Solr, que es el programa de código libre perteneciente al proyecto Apache que se ocupa de las tareas de búsqueda y recuperación de información. Solr emplea la librería Lucene para las tareas esenciales de indexación y

---

[16] Código libre (en inglés "*free software*", aunque esta denominación a veces se confunde con *"*gratis*"* por la ambigüedad del término *"free"* en el idioma inglés, por lo que también se usa **"***libre software***"**) es la denominación del software que respeta la libertad de todos los usuarios que adquirieron el producto y, portanto, una vez obtenido el mismo puede ser usado, copiado, estudiado, modificado, y redistribuido libremente de varias formas.

[17] Parser o analizador sintáctico: es una de las partes de un compilador que transforma su entrada en un árbol de derivación.

[18] Lucene es una API de código abierto para recuperación de información, originalmente implementada en Java

[19] Apache Tomcat (también llamado Jakarta Tomcat o simplemente Tomcat) funciona como un contenedor de servlets desarrollado bajo el proyecto Jakarta en la Apache Software Foundation. Tomcat implementa las especificaciones de los servlets y de JavaServer Pages (JSP) de Sun Microsystems.

[20] Jetty es un servidor HTTP basado en Java.

recuperación, pero a través de Tomcat o Jetty permite una configuración externa integral que posibilita el desarrollo de cualquier aplicación de búsqueda y recuperación de manera relativamente sencilla. Sin embargo, es necesario instalar por separado Nutch y Solr, configurarlos correctamente, y posteriormente integrarlos de manera que las páginas obtenidas con Nutch puedan ser indexadas y recuperadas mediante Solr en una consola web.

También extraer información desde archivos PDF, XML, HTML, y otros. Es considerada la solución de código abierto más usada en SRI. Posee una amplia comunidad de desarrolladores y usuarios. Su desarrollo está patrocinado por la Fundación Apache[21]. Entre sus principales características destacan:

- Captura, *parser* e indexación en modo paralelo y distribuido.
- Extensible mediante *plugins*[22].
- Soporte para diversos formatos, tales como: texto plano, HTML, XML, ZIP, ODF, PDF; JS, RSS, etc.
- Solución basada en clúster.
- Sistema de fichero distribuido.
- Soporte para autenticación NTLM[23].

2.4.2 – Mnogosearch

Mnogosearch es un motor de búsqueda completo de código abierto y basado en SQL. Consiste en dos partes. La primera parte es un mecanismo de indexación *indexer.conf* el cual se mueve a través de vínculos de hipertexto HTML y almacena información

[21] Apache Software Foundation (ASF) es una organización no lucrativa, creada para dar soporte a los proyectos de software bajo la denominación *Apache.*

[22] Plugins**:** es una aplicación que se relaciona con otra para aportarle una función nueva y generalmente muy específica.

[23] Autenticación NTML: es un procedimiento de autenticación que habilita al usuario para acceder a varios sistemas con una sola instancia de identificación. Su traducción literal sería algo como "sistema centralizado de autenticación y autorización"

acerca de los documentos en la base de datos. La segunda parte es una interfaz web CGI[24], *search.cgi* la cual muestra en el navegador un formulario HTML y los 5 resultados de búsquedas. Search.cgi utiliza información recopilada por el indexador[26]. Entre sus principales características destacan:

- Soporte para diversos protocolos: HTTP, HTTPS, FTP, NNTP.
- Analizadores incorporados para diversos formatos de archivo: text/html, text/xml, text/plain y audio/mpeg.
- Soporte para autenticación de proxy.
- indexación multihilo.
- Interfaces web CGI, Perl[25] y PHP.
- Lenguaje de consulta booleano.
- Soporte para la mayoría de los conjuntos de caracteres modernos.
- Soporte para múltiples bases de datos: MySql, PostgreSQl, SQLite, Mimer, Virtuoso, Interbase, Oracle, MS SQL, DB2, Sysbase.
- Posee una API externa para PHP.
- Fácil de configurar.

MnoGoSearch es software libre, lanzado bajo la licencia GPL de la Free Software Fundation. Integra la mayoría de los conceptos y características asociadas a los motores de búsqueda actuales. Para el cálculo del Page Rank[26], emplea los algoritmos

---

[24]Interfaz de entrada común (en inglés *Common Gateway Interface*, abreviado CGI) es una importante tecnología de la WWW que permite a un cliente (navegador web) solicitar datos de un programa ejecutado en un servidor web. CGI especifica un estándar para transferir datos entre el cliente y el programa. Es un mecanismo de comunicación entre el servidor web y una aplicación externa cuyo resultado final de la ejecución son objetos MIME o *Multipurpose Internet Mail Extensions (en español* extensiones multipropósito de correo de internet). Las aplicaciones que se ejecutan en el servidor reciben el nombre de CGIs.

[25] Perl es un lenguaje de programación diseñado por LARRY WALL en 1987.

[26] PageRank es una marca registrada y patentada por Google el 9 de enero de 1999 que ampara una familia de algoritmos utilizados para asignar de forma numérica la relevancia de los documentos (o páginas web) indexados por un motor de búsqueda.

Espacio Vectorial y Booleano. Ambos algoritmos han sido ampliamente probados y utilizados, obteniéndose muy buenos resultados con los mismos.

Conclusiones parciales del capítulo.

En el capítulo se realizó un análisis del modelo general y como está conformada su arquitectura, se describieron los componentes importantes de los sistemas de recuperación de información y se caracterizaron por separado los Sistemas de Recuperación de Información de arquitectura centralizada y distribuida. También realizó un estudio sobre los principales Sistemas de Recuperación de Información de software libre existentes, con el objetivo de seleccionar el más adecuado a los propósitos del proyecto teniendo en cuenta los requerimientos dictados por la empresa y se seleccionaron dos de los sistemas de recuperación de información para luego aplicar un método de evaluación que valide su desempeño.

Capítulo III: Instalación y evaluación de los Sistemas de recuperación de Información seleccionados.

Los Sistemas de Recuperación de Información, resultan susceptibles, como cualquier otro sistema, de ser sometidos a evaluación, para que sus usuarios puedan valorar su efectividad. La tradición de la evaluación es tan antigua casi como el desarrollo de los primeros SRI, encontrándose estrechamente vinculadas con la investigación y el desarrollo de la recuperación de información. Realmente, "la propia naturaleza de los SRI propicia su necesidad crítica de evaluación, justo como cualquier otro campo de trabajo que aspire a ser clasificado como campo científico"[27]. [27]

En [15] se manifiesta que "un SRI puede ser evaluado por diversos criterios, incluyendo entre los mismos: la eficacia en la ejecución, el efectivo almacenamiento de los datos, la efectividad en la recuperación de la información y la serie de características que ofrece el sistema al usuario". Estos criterios no deben confundirse, la eficacia en la ejecución: es la medida del tiempo para realizar una operación, la eficiencia del almacenamiento: es el espacio que se precisa para almacenar los datos y por último está la efectividad de la recuperación: "normalmente basada en la relevancia de los documentos recuperados"[28].

3.1 – Instalación de los sistemas de recuperación de información.

3.1.1 – Instalación de Nutch.

Para la instalación del sistema los autores plantean una serie de pasos y requisitos:[25]

Prerrequisitos: Es preciso tener previamente instalado Java SDK 1.5 o superior.

---

[27] BLAIR, D.C. Language and representation in information retrieval. Amsterdam: Elsevier Science Publishers, 1990.

[28] BAEZA-YATES, R. and FRAKES, W.B. Information retrieval: data structures & algorithms Englewood Cliffs, New Yersey: Prentice Hall, 1992.

Paso 1: Ir a http://www.apache.org/dyn/closer.cgi/nutch/ donde sugieren un repositorio desde el que descargar Nutch. En nuestro caso, nos remite a http://apache.rediris.es/ /nutch. Allí buscamos la versión más adecuada, en nuestro caso, apache-nutch-1.3-bin.tar.gz, y guardamos el archivo en el Escritorio.

Paso 2: Mover el archivo al lugar donde vaya a instalarse el programa, en nuestro caso, /usr/local. Para ello tecleamos en un terminal:

```
$ mv /home/juan/Escritorio/apache-nutch-1.3-bin.tar.gz /usr/local/
```

Paso 3: Descomprimir el archivo en su lugar de instalación. Para ello tecleamos en un terminal:

```
$ cd /usr/local
$ tar -zxvf apache-nutch-1.3-bin.tar.gz
```

Ello creará el directorio “nutch-1.3” en /usr/local/

Paso 4: Verificar la correcta instalación de Nutch. Para ello, ir a /usr/local/nutch-1.3/runtime/local tecleando en un terminal:

```
$ cd /usr/local/nutch-1.3/runtime/local
```

Una vez allí, se le permite la ejecución del programa al usuario que habitualmente empleará el programa.

```
$ chmod +x bin/nutch
```

De igual forma, se comprueba que la variable de entorno JAVA_HOME está correctamente configurada.

Efectuadas estas comprobaciones, se ejecuta el programa Nutch tecleando:

```
$ bin/nutch
```

La instalación es correcta si se observan las siguientes líneas:

```
Usage: nutch [-core] COMMAND
where COMMAND is one of:
…..........................................
NOTE: this works only for jobs executed in 'local' mode
```

Paso 5: Se configura inicialmente el programa para poder realizar rastreos de la web.

En primer lugar, se incluye el nombre del agente que llevará a cabo el crawling. Esta información sobre quién lleva a cabo las tareas de rastreo debe añadirse a uno de los archivos de configuración del programa. Suponiendo que el agente es “Mi robot Nutch”, primeramente nos situamos en:

$ cd /usr/local/nutch-1.3/runtime/local/conf

A continuación, para conservar la versión inicial del archivo nutch-site.xml, lo renombramos:

$ mv nutch-site.xml nutch-site-old.xml

Luego se hace una copia del archivo nutch-default.xml para emplearlo como archivo nutch-site.xml, que a su vez se modificara posteriormente para configurar el programa adaptándolo a las necesidades:

$ cp nutch-default.xml nutch-site.xml

Se procede a editar el archivo nutch-site.xml:

$ gedit nutch-site.xml

Y modificar la propiedad “agent.name” de manera que quede:

<property>
<name>http.agent.name</name>
<value>Mi robot Nutch</value>
<property>

Se guardan los cambios efectuados en el archivo.

Paso 6: Crear un archivo de texto plano con la/las url/urls inicial/iniciales que se emplearán a modo de semillas (seeds) para rastrear la web.

Para ello, es necesaria la creación de un directorio denominado “urls” directamente bajo nutch-1.3/runtime/local:

$ mkdir /usr/local/nutch-1.3/runtime/local/urls

Se crea el archivo “lista”:

$ gedit /usr/local/nutch-1.3/runtime/local/urls/lista

Una vez que se haya tecleado la dirección url, se guardan y cierran los archivos.

Paso 7: Después de la realización de los pasos anteriores se tienen las condiciones para efectuar un primer rastreo del sitio web de Nutch. Para ello hay que situarse en:

$ cd /usr/local/nutch-1.3/runtime/local

$ bin/nutch crawl urls -dir crawl -depth 3 -topN 5

Se habrá creado el directorio nutch-1.3/runtime/local/crawl con los siguientes subdirectorios: crawldb, linkdb y segments, donde se hallan los resultados del rastreo del sitio web de Nutch.

Paso 8: Para comprobar cuántas urls hay en la base de datos y cuántas han sido rastreadas, se ejecuta el siguiente comando en un terminal:

$ bin/nutch readdb /usr/local/nutch-1.3/runtime/local/crawl/crawldb -stats

Se mostrará en pantalla una información semejante a esta:

TOTAL urls: 413

…….....

…….....

status 1 (db_unfetched): 402

status 2 (db_fetched): 11

CrawlDb statistics: done

Paso 9: Se utiliza la versión de apache-solr 3.3.0, y se descarga el Escritorio.

Paso 10: Se mueve el archivo al lugar donde vaya a instalarse, en este caso, /usr/local. Para ello se teclea en un terminal:

$ mv /home/juan/Escritorio/apache-solr-3.3.0.zip /usr/local/

Paso 11: Se descomprime el archivo en el lugar donde se vaya a instalar Solr. Para ello, tecleamos en un terminal:

$ cd /usr/local

$ unzip apache-solr-3.3.0.zip

Se creará el directorio /usr/local/apache-solr-3.3.0

Paso 12: Solr puede funcionar con cualquier contenedor de servlets, como Tomcat, pero es suficiente con emplear Jetty, cuya instalación es muy simple. Para ello teclear en un terminal:

$ cd /usr/local/apache-solr-3.3.0/example

$ java -jar start.jar

Si se Observa un mensaje parecido a este en pantalla:

08-jul-2011 19:26:47 org.apache.solr.core.SolrCore registerSearcher

INFO: [ ] Registered new searcher Searcher@18efaea main

….....

….....

2011-07-08 19:26:47.488:INFO:Started SocketConnector@0.0.0.0:8983

No se debe dar a ninguna tecla, aunque el cursor esté parpadeando. De hecho, con este último comando se a iniciado Jetty en el puerto 8983, y está en funcionamiento. Hay que limitarse a minimizar el terminal, de manera que siga funcionando Jetty, y abrir el navegador introduciendo la dirección:

http://localhost:8983/solr/admin/

Si el proceso se ha efectuado correctamente, se observará en la pantalla inicial de la consola de administración del programa Solr (que se ha iniciado también junto con Jetty). De igual forma, si introducimos la dirección:

http://localhost:8983/solr/admin/stats.jsp

Se tiene una pantalla con toda la información acerca del programa Solr y de la colección cargada (la primera vez, lógicamente, indica numDocs: 0, esto es, que no hay ningún documento cargado en el sistema). Para cerrar la pantalla y consiguientemente el programa Solr, basta teclear Ctrl-C en el terminal que tenemos minimizado.

Paso 13: Una vez que tanto Nutch como Solr se han instalado y configurado correctamente, se deben integrar de manera que las urls obtenidas con Nutch puedan ser recuperadas mediante Solr. Para ello, es necesario teclearen un terminal:

$ cp /usr/local/nutch-1.3/runtime/local/conf/schema.xml /usr/local/apache-solr-

3.3.0/example/solr/conf/

De manera que se reemplace el archivo schema.xml por defecto de Solr con el de Nutch.

Paso 14: Reiniciar Solr (junto con la consola web facilitada por Jetty) con el comando empleado en el Paso 12:

```
$ cd /usr/local/apache-solr-3.3.0/example
$ java -jar start.jar
```

Minimizar el terminal para seguir teniendo acceso a la consola web.

Paso 15: Abrir otro terminal, dejando el anterior minimizado, e introducir el comando de Nutch que efectúa la indexación de las urls rastreadas en el Paso 7 anterior. Para ello, teclear en el nuevo terminal:

```
$ cd /usr/local/nutch-1.3/runtime/local
$ bin/nutch solrindex http://127.0.0.1:8983/solr/ crawl/crawldb crawl/linkdb
crawl/segments/*
```

Este comando consigue que Solr indexe todos los datos del rastreo efectuado en el Paso 7. En pantalla observaremos un mensaje semejante a este:

SolrIndexer: starting at 2014-03-09 20:13:24

SolrIndexer: finished at 2014-03-09 20:13:26 elapsed: 00:00:02

Paso 16: Si el proceso se ha efectuado correctamente, podemos empezar a realizar búsquedas sobre esas páginas. Para ello, abrimos el navegador e introducimos la dirección:

http://localhost:8983/solr/admin/

En Query String se puede introducir, por ejemplo, la búsqueda del término “nutch”: Query String: +nutch

Hacer clic en el botón “Search” y obtener un fichero en XML con los documentos (páginas web rastreadas) que satisfacen esa consulta.

3.1.2 – Instalación de Mnogosearch

Para la instalación del sistema los autores plantean una serie de pasos y requisitos:[26]

Paso 1: Expandir la distribución fuente y cambiar el directorio a las fuentes descomprimidas. Por ejemplo:

tar-zxf mnoGoSearch-3.3.8.tar.gz

cd mnoGoSearch-3.3.8

Paso 2: Configurar el paquete.

Nota: Para simplificar el proceso de configuración, distribución fuente mnogosearch incluye un install.pl script de configuración opcional. Puede ejecutar install.pl y seleccionar las opciones de configuración mnogosearch en la forma de preguntas y respuestas. Después de responder a todas las preguntas, el script se ejecute y configure con las opciones que elija. Así, se creará el archivo install.options que contiene sus preferencias de configuración que puede utilizar para ejecutar el script más tarde, sin pasar por las preguntas.

Si ha decidido utilizar install.pl, vaya al Paso 3, después de que se haya finalizado la configuración. En caso de que prefiera configurar Mnogosearch de la manera tradicional (sin usar install.pl), haga lo siguiente:

sh / configure - with-mysql

o

sh / configure - with-pgsql

Con otra base de datos de su elección o con múltiples bases de datos:

sh / configure - with-mysql - with-pgsql - with-freetds

Por defecto, mnogosearch se instala en el directorio / usr / local / mnogosearch con los siguientes subdirectorios:

Tabla 2: Directorios que son instalados por defecto.

Directorio	Contenido
bin	mconv, mguesser, search.cgi, udm-config
lib	libmnogocharset.a (so), libmnogosearch.a (so)
sbin	indexador,
etc	indexer.conf-dist, search.htm-dist, langmap.conf-dist, stopwords.conf-dist
man	indexer.1, indexer.conf.5
doc	diversa documentación

Si no tiene permiso para escribir en ese directorio o simplemente desea instalar mnogosearch a otra ubicación, puede utilizar configure con el -prefix opción, por ejemplo:

. / Configure - prefix = / user / home / mnogo - with-mysql

Para instalar mnoGoSearch con el uso de soporte HTTPS configure con la opción siguiente:

. / Configure - with-openssl

o en el caso cuando la biblioteca OpenSSL está instalada en una ubicación no estándar:

. / Configure - with-openssl = / ruta / al / library

Nota: Se requiere la biblioteca OpenSSL para construir mnogosearch con soporte HTTPS.

De esta forma se pueden ver todas las opciones disponibles, escriba: / Configure - ayuda

Para construir mnogosearch con soporte para todos los conjuntos de caracteres adicionales, utilice:

. / Configure - with-extra-charsets = all

Para construir mnogosearch con varios conjuntos de caracteres adicionales, use una lista separada por comas de los juegos de caracteres que desee:

. / Configure - with-extra-charset = big5, gb2312

Paso 3: Construir e instalar el paquete.

sh make

sh make install

Paso 4: Crear una nueva base de datos mnogosearch usará para almacenar los datos. Por ejemplo, mnogosearch.

Se puede utilizar una base de datos existente. En este caso, omita este paso.

Para MySQL:

sh mysqladmin create mnoGoSearch

Para PostgreSQL:

mnoGoSearch sh createdb

Paso 5: Cree el archivo indexer.conf y establecer el DBAddr comando.

Cambia al directorio / usr / local / mnoGoSearch / etc /

Copie indexer.conf-dist en indexer.conf:

cp-indexer.conf dist indexer.conf

Abra indexer.conf en su editor de texto favorito y edite el DBAddr comando para establecer la cadena de conexión de base de datos adecuada.

Paso 6: Crear search.htm y establecer DBAddr

Cambiar la dirección a / etc directorio de su instalación mnoGoSearch, normalmente /usr /local / mnoGoSearch / etc /

Copie search.htm-dist en search.htm:

cp-search.htm dist search.htm

Abrir search.htm en su editor favorito y modificar el DBAddr para establecer las cadenas de conexión de base de datos, de manera similar a lo que se hizo en el paso anterior.

Paso 7: Crear tablas.

Cambiar dirección al directorio sbin / de la instalación, normalmente / usr / local / mnoGoSearch / sbin, y crear la estructura de base de datos:

sh . / indexador-Ecreate

Paso 8: Instalación de secuencias de comandos de búsqueda

Copia search.cgi a su directorio cgi-bin del servidor Web o añadir un alias Apache en el directorio bin mnogosearch.

Paso 9: Introducir las guías de búsqueda para comenzar el aprendizaje.

Paso 10: Ir a la página principal de SRI

3.2 – Ventajas que proporcionan.

3.2.1 – Ventajas que proporciona Mnogosearch

Búsqueda de contenidos generales y búsqueda por tipo de contenidos (imágenes, documentos). El motor de búsqueda tiene la capacidad de buscar distintos tipos de contenidos existentes en la red tales como: imágenes (GIF, PNG, JPEG), documentos (PDF, TXT, DOC, PPT). El *robot* durante el proceso de indexación, utiliza programas externos conocidos como *parsers* los cuales son encargados de extraer la información existente en los archivos PDF, DOC y PPT.[26]

Módulo de noticias obtenidas mediante canales RSS[29].

El módulo de noticias obtiene las últimas noticias publicadas en los sitios dados de alta en el buscador mediante sus propios canales RSS. La base de datos que contiene las últimas noticias es actualizada automáticamente cada 30 minutos. Las noticias pueden ser votadas positiva o negativamente por los usuarios, lo que unido a la cantidad de lecturas de la noticia, influye en la posición ocupada en la sección de noticias destacadas.

Autocompletamiento de palabras a buscar con búsquedas realizadas anteriormente. Las búsquedas realizadas por los usuarios, son almacenadas en la base de datos del sistema. El motor de búsqueda permite, haciendo uso de este conocimiento previo,

[29] RSS (Really Simple Syndication), un formato XML para sindicar o compartir contenido en la web. Se utiliza para difundir información actualizada frecuentemente a usuarios que se han suscrito a la fuente de contenidos.

autocompletar las consultas que teclean los usuarios, mejorando así la experiencia de los mismos en la búsqueda de información relevante.

Sugerencia de palabra correcta cuando se comete un error al escribir el criterio de búsqueda. Cuando el usuario comete un error al teclear un criterio de búsqueda, el sistema responde con una sugerencia del posible criterio de búsqueda deseado empleando algoritmos de búsqueda fonética. Imaginemos que el usuario introduce el criterio de búsqueda **"serbicios"**, evidentemente, este vocablo está escrito incorrectamente. El sistema sugiere el vocablo "**servicios**".

Agrupamiento de los resultados por sitio.

El SRI permite agrupar los resultados encontrados por sitio, o sea, resultados de un mismo sitio, son agrupados en un enlace que puede ser consultado en cualquier momento.

3.2.2 – Ventajas que proporciona Nutch.

Las búsquedas se efectúan mediante un interfaz web, cuyo formulario admite búsquedas mediante lenguaje natural, y también el uso de operadores. Éstos pueden ser los clásicos booleanos, pero también operadores de adyacencia; el sistema admite búsquedas de frases, así como una mezcla o combinación de todo lo anterior.

Los resultados se ofrecen mediante una o varias páginas web, ordenados por relevancia. Para cada documento encontrado, además de su enlace correspondiente y otros elementos de información (título, formato y resumen), De manera opcional, el sistema puede almacenar internamente las consultas efectuadas y los enlaces o documentos, de entre los recuperados, navegados después. Estos datos, junto con los recogidos en el log del propio servidor web, pueden ser muy útiles para estudiar las pautas y comportamientos de los usuarios. [26]

3.3 – Evaluación.

3.3.1 – Selección de variables de Evaluación.

Existe una serie de trabajos centrados la definición de un marco global para la evaluación de los SRI [28][29], haciendo análisis comparativos. Aunque el objeto principal de estos trabajos varía de uno a otro, resulta complicado establecer una línea común entre ellos, coinciden en su pretensión de no ofrecer como resultado qué motor es más preciso o cuál de ellos ha crecido más en los últimos tiempos, sino en la idea de concebir una propuesta integral de evaluación de estos sistemas.

El análisis realizado en el trabajo titulado “Evaluación de los motores de búsqueda WWW” [30], donde los autores presentan una sugerencia de criterios mínimos necesarios a tener en cuenta para la evaluación de un SRI, fruto de una exhaustiva síntesis de las medidas empleadas en otros trabajos de evaluación de estos sistemas.

3.3.1– Evaluación de los motores de búsqueda WWW

Los autores recogen un amplio conjunto de criterios empleados en trabajos previos de evaluación y sus conclusiones. Tras realizar esta exhaustiva síntesis de criterios para la evaluación de los motores de búsqueda, los autores consideran que “resulta claro que muchas de las investigaciones desarrolladas son inconsistentes en método y enfoque. Esto demuestra la inmadurez de este campo de estudio”[30]. Para los autores las evaluaciones de los SRI deberían incluir, como mínimo alguno de los siguientes criterios:

- Precisión
- Velocidad de respuesta, analizada varias veces al día y calculada en términos de promedio.
- Consistencia de resultados a lo largo de un determinado período de tiempo.
- Proporción de enlaces fallidos.
- Proporción de duplicados.
- Calidad promedio de resultados

- Evaluación de la amigabilidad de la interface
- Calidad de la ayuda
- Opciones para la visualización de los resultados
- Presencia de avisos en la pantalla
- Longitud esperada de búsqueda.
- Longitud y legibilidad del resumen.

Así mismo, los autores consideran que las pruebas deben realizarse haciendo uso de tres tipos de búsqueda: búsqueda normal, frases en lenguaje natural y expresiones booleanas.

Tabla 3: Evaluación de búsqueda.

Variables medidas	Nutch	Mnogosearch
Precisión	93%	98%
Velocidad de respuesta, analizada varias veces al día y calculada en términos de promedio.	De 0 a 2 segundos	De 0 a 2 segundos
Consistencia de resultados a lo largo de un determinado período de tiempo.	Bueno	Bueno
Proporción de enlaces fallidos.	5%	3%
Calidad promedio de resultados	Regular	Buena
Evaluación de la amigabilidad de la interface	Regular	Regular

Calidad de la ayuda	Buena	Buena
Opciones para la visualización de los resultados	Regular	Buena
Presencia de avisos en la pantalla	Buena	Regular
Longitud y legibilidad del resumen.	Regular	Buena

Conclusiones parciales del capítulo.

En éste capítulo se realizó una descripción de la instalación de los sistemas de recuperación de información seleccionados en el capítulo anterior para tener una visión de las ventajas de cada uno de ellos y que nos brindan. Además se le da cumplimiento al método de evaluación propuesto en el artículo [30] y se tomaron las variables descritas para hacer una selección fundamentada en los criterios de los autores, obteniendo como el más destacado a Mnogosearch por presentar resultados de búsqueda alentadores para la empresa.

Conclusiones

Con el desarrollo de esta investigación se mejoró la búsqueda y recuperación de la información existente en la red WAN de ETECSA, lo que permitió ahorrar tiempo y esfuerzo en una tarea que realizarla de forma manual, puede llevar algo de tiempo y en muchos casos, no se logra encontrar la información con un adecuado grado de relevancia, para lo cual se realizó una selección e instalación de un sistema de recuperación de información. Además se puede concluir que:

- Se realizó un estudio que permitió conocer los requerimientos de la empresa.
- Se realizó un análisis a los sistemas de recuperación de información atendiendo a los requerimientos de la empresa.
- Se estudiaron las diferentes metodologías para la evaluación de los métodos de recuperación de información que se utilizaron.
- Se seleccionó el sistema de recuperación de información Mnogosearch debido a los resultados alcanzados en la evaluación.
- Se analizaron los algoritmos y métodos utilizados por los sistemas de recuperación de información.
- Se implantó un sistema de recuperación de información Mnogosearch para facilitar la búsqueda de información en la red WAN de ETECSA.
- Se realizaron pruebas en escenarios reales que validaron el estudio antes realizado.

Recomendaciones

Concluida la investigación se recomienda realizar cambios a la interfaz teniendo en cuenta el criterio de los trabajadores, actualizar la base de búsqueda semanalmente debido a la variabilidad de estado de la información, incorporar un diccionario y un traductor para garantizar a los usuarios del sistema la formación personal y profesional en cuanto a lenguas extranjeras se refiere.

Referencias bibliográficas

[1] W. B. Frakes y R. Baeza-Yates, «Information retrieval: data structures and algorithms», 1992.

[2] C. D. Manning, P. Raghavan, y H. Schütze, *Introduction to information retrieval*, vol. 1. Cambridge university press Cambridge, 2008.

[3] P. Sotolongo, «Complejidad, no linealidad y redes distribuidas», *INGENIERIA CIVIL*, vol. 1, n.º 1, 2014.

[4] X. Zhu y S. Gauch, «Incorporating quality metrics in centralized/distributed information retrieval on the World Wide Web», en *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, 2000, pp. 288–295.

[5] J. Callan, «Distributed information retrieval», en *Advances in information retrieval*, Springer, 2000, pp. 127–150.

[6] Baeza-Yates, R. y Ribeiro-Neto, B, *Modern information retrieval*. New York, 1999.

[7] M. Pérez-Montoro, «Arquitectura de la información en entornos web», *El profesional de la información*, vol. 19, n.º 4, pp. 333–338, 2010.

[8] F. W. Lancaster, *El control del vocabulario en la recuperación de información*, vol. 12. Universitat de València, 2002.

[9] S. Ceri, A. Bozzon, M. Brambilla, E. Della Valle, P. Fraternali, y S. Quarteroni, «Information Retrieval Models», en *Web Information Retrieval*, Springer, 2013, pp. 27–37.

[10] W. A. Katz, *Introduction to Reference Work â€“*. 1969.

[11] C. T. Meadow, *Text information retrieval systems*. Academic Press, Inc., 1992.

[12] F. L. Pascual, «IR-n un sistema de Recuperación de Información basado en pasajes», *Procesamiento del lenguaje natural*, n.º 30, pp. 127–128, 2003.

[13] B. Chor, E. Kushilevitz, O. Goldreich, y M. Sudan, «Private information retrieval», *Journal of the ACM (JACM)*, vol. 45, n.º 6, pp. 965–981, 1998.

[14] D. E. Rose, *A symbolic and connectionist approach to legal information retrieval*. Psychology Press, 2013.

Referencias Bibliográficas

[15] R. Baeza-Yates, B. Ribeiro-Neto, y others, *Modern information retrieval*, vol. 463. ACM press New York, 1999.

[16] Antonio G. López Herrera, *Modelos de Sistemas de Recuperación de Información Lingüística Difusa*. Psychology Press, 2006.

[17] P. I. Borkar y L. H. Patil, «Web information retrieval using genetic algorithm-particle swarm optimization», *International Journal of Future Computer and Communication*, vol. 2, n.º 6, pp. 595–599, 2013.

[18] Archuby, C., «Modelos, analogías, metáforas y equivalencias, como instrumentos del trabajo intelectual». La Plata: UNLP, 2003.

[19] J. C. R. Cano, «TÉCNICAS DE RECUPERACIÓN COLABORATIVA DE INFORMACIÓN EN ENTORNOS DISTRIBUIDOS», 2010.

[20] G. Salton y C. Buckley, «Term-weighting approaches in automatic text retrieval», *Information processing & management*, vol. 24, n.º 5, pp. 513–523, 1988.

[21] Bookstein,A, *Journal of the American Society for Information Science*. 1979.

[22] V. V. Raghavan y S. Wong, «A critical analysis of vector space model for information retrieval», *Journal of the American Society for information Science*, vol. 37, n.º 5, pp. 279–287, 1986.

[23] S.-F. Chang, J. R. Smith, M. Beigi, y A. Benitez, «Visual information retrieval from large distributed online repositories», *Communications of the ACM*, vol. 40, n.º 12, pp. 63–71, 1997.

[24] G. Kowalski, *Information retrieval systems: theory and implementation*. Kluwer Academic Publishers, 1997.

[25] «About Apache Nutch». [En línea]. Disponible en: http://nutch.apache.org/about.html. [Accedido: 04-jun-2014].

[26] «mnoGoSearch 3.3.15 reference manual». [En línea]. Disponible en: http://www.mnogosearch.org/doc33/. [Accedido: 04-jun-2014].

[27] L. Codina, «Evaluación de recursos digitales en línea: conceptos, indicadores y métodos», *Revista española de documentación científica*, vol. 23, n.º 1, pp. 9–44, 2000.

Referencias Bibliográficas

[28] F. J. Martínez Méndez y J. V. Rodríguez Muñoz, «Reflexiones sobre la evaluación de los sistemas de recuperación de información: necesidad, utilidad y viabilidad.», 2004.

[29] M. D. Olvera Lobo, «Evaluación de sistemas de recuperación de información: aproximaciones y nuevas tendencias», *Profesional de la Información, El: Information World en Español*, vol. 8, n.º 11, pp. 4–14, 1999.

[30] Oppenheim, C, Morris, A, y McKnight, C, «The evaluation of WWW search engines», vol. 56, n.º 2, pp. 190-211, mar. 2000.

Bibliografía

[1] V. V. Raghavan y S. Wong, «A critical analysis of vector space model for information retrieval», *Journal of the American Society for information Science*, vol. 37, n.º 5, pp. 279–287, 1986.

[2] J. M. Ponte y W. B. Croft, «A language modeling approach to information retrieval», en *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, 1998, pp. 275–281.

[3] D. E. Rose, *A symbolic and connectionist approach to legal information retrieval*. Psychology Press, 2013.

[4] P. Castells, M. Fernandez, y D. Vallet, «An adaptation of the vector-space model for ontology-based information retrieval», *Knowledge and Data Engineering, IEEE Transactions on*, vol. 19, n.º 2, pp. 261–272, 2007.

[5] M. Pérez-Montoro, «Arquitectura de la información en entornos web», *El profesional de la información*, vol. 19, n.º 4, pp. 333–338, 2010.

[6] R. Baeza-Yates, C. R. Loaiza, y J. V. Martín, «Arquitectura de la información y usabilidad en la web», *El profesional de la información*, vol. 13, n.º 3, pp. 168–178, 2004.

[7] P. Sotolongo, «Complejidad, no linealidad y redes distribuidas», *INGENIERIA CIVIL*, vol. 1, n.º 1, 2014.

[8] A. Beltrán Fonollosa, L. Díaz Sánchez, y J. Huerta Guijarro, «Construyendo un sistema de indexación y búsqueda de recursos georreferenciados», 2012.

[9] M. S. Lew, N. Sebe, C. Djeraba, y R. Jain, «Content-based multimedia information retrieval: State of the art and challenges», *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)*, vol. 2, n.º 1, pp. 1–19, 2006.

[10] P. Hípola, B. Vargas-Quesada, y A. Montes, «Descripción y evaluación de agentes multibuscadores», *El profesional de la información*, vol. 8, n.º 11, 1999.

[11] C. G. Figuerola, A. Berrocal, y J. Luis, «Diseño de un motor de recuperación de la información para uso experimental y educativo», 2000.

Bibliografía

[12] J. Callan, «Distributed information retrieval», en *Advances in information retrieval*, Springer, 2000, pp. 127–150.

[13] J. A. Senso y A. de la R. Piñero, «El concepto de metadato. Algo más que descripción de recursos electrónicos», *Ciência da Informação*, vol. 32, n.º 2, pp. 95–106, 2003.

[14] F. W. Lancaster, *El control del vocabulario en la recuperación de información*, vol. 12. Universitat de València, 2002.

[15] L. Codina, «Evaluación de recursos digitales en línea: conceptos, indicadores y métodos», *Revista española de documentación científica*, vol. 23, n.º 1, pp. 9–44, 2000.

[16] M. D. Olvera Lobo, «Evaluación de sistemas de recuperación de información: aproximaciones y nuevas tendencias», *Profesional de la Información, El: Information World en Español*, vol. 8, n.º 11, pp. 4–14, 1999.

[17] I. Clarke, O. Sandberg, B. Wiley, y T. W. Hong, «Freenet: A distributed anonymous information storage and retrieval system», en *Designing Privacy Enhancing Technologies*, 2001, pp. 46–66.

[18] T. K. Landauer, D. S. McNamara, S. Dennis, y W. Kintsch, *Handbook of latent semantic analysis*. Psychology Press, 2013.

[19] X. Zhu y S. Gauch, «Incorporating quality metrics in centralized/distributed information retrieval on the World Wide Web», en *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, 2000, pp. 288–295.

[20] P. De Bra, G.-J. Houben, Y. Kornatzky, y R. Post, «Information Retrieval in Distributed Hypertexts.», en *RIAO*, 1994, pp. 481–493.

[21] J. Zuo, M. Wang, J. Wan, y W. Luo, «Information Retrieval Model Combining Sentence Level Retrieval», en *Asian Language Processing (IALP), 2013 International Conference on*, 2013, pp. 37–40.

[22] S. Ceri, A. Bozzon, M. Brambilla, E. Della Valle, P. Fraternali, y S. Quarteroni, «Information Retrieval Models», en *Web Information Retrieval*, Springer, 2013, pp.

Bibliografía

27–37.

[23] G. Kowalski, *Information retrieval systems: theory and implementation*. Kluwer Academic Publishers, 1997.

[24] W. B. Frakes y R. Baeza-Yates, «Information retrieval: data structures and algorithms», 1992.

[25] R. Braden, D. Clark, S. Shenker, y others, *Integrated services in the internet architecture: an overview*. rfc 1633, June, 1994.

[26] A. Afuah y C. L. Tucci, *Internet business models and strategies: Text and cases*. McGraw-Hill Higher Education, 2000.

[27] R. Albert, H. Jeong, y A.-L. Barabási, «Internet: Diameter of the world-wide web», *Nature*, vol. 401, n.º 6749, pp. 130–131, 1999.

[28] C. D. Manning, P. Raghavan, y H. Schütze, *Introduction to information retrieval*, vol. 1. Cambridge university press Cambridge, 2008.

[29] W. A. Katz, *Introduction to Reference Work â€“*. 1969.

[30] F. L. Pascual, «IR-n un sistema de Recuperación de Información basado en pasajes», *Procesamiento del lenguaje natural*, n.º 30, pp. 127–128, 2003.

[31] L. C. García Figuerola, «La investigación sobre Recuperación de la Información en español», 2000.

[32] E. Yom-Tov, S. Fine, D. Carmel, y A. Darlow, «Learning to estimate query difficulty: including applications to missing content detection and distributed information retrieval», en *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, 2005, pp. 512–519.

[33] M. W. Berry, Z. Drmac, y E. R. Jessup, «Matrices, vector spaces, and information retrieval», *SIAM review*, vol. 41, n.º 2, pp. 335–362, 1999.

[34] E. M. M. Rodríguez, *Metadatos y recuperación de información: estándares, problemas y aplicabilidad en bibliotecas digitales*. Trea, 2002.

[35] L. Codina, «Metodología de análisis y evaluación de recursos digitales en línea», *Documento reprografiado*, vol. 6, 2005.

[36] R. Baeza-Yates, B. Ribeiro-Neto, y others, *Modern information retrieval*, vol. 463.

Bibliografía

ACM press New York, 1999.

[37] B. Chor, E. Kushilevitz, O. Goldreich, y M. Sudan, «Private information retrieval», *Journal of the ACM (JACM)*, vol. 45, n.º 6, pp. 965–981, 1998.

[38] K. S. Jones, *Readings in information retrieval*. Morgan Kaufmann, 1997.

[39] F. J. Martínez Méndez y J. V. Rodríguez Muñoz, «Reflexiones sobre la evaluación de los sistemas de recuperación de información: necesidad, utilidad y viabilidad.», 2004.

[40] M. D. Olvera Lobo, «Rendimiento de los sistemas de recuperación en la world wide web: revisión metodológica.», *Revista española de documentación científica*, vol. 23, n.º 1, pp. 63–77, 2000.

[41] P. Thompson, «Satisficing or the Right Information at the Right Time: Artificial Intelligence and Information Retrieval, a Comparative Study in Medicine and Law», *Medical Applications of Artificial Intelligence*, p. 71, 2013.

[42] W. B. Croft, D. Metzler, y T. Strohman, *Search engines: Information retrieval in practice*. Addison-Wesley Reading, 2010.

[43] J. C. R. Cano, «TÉCNICAS DE RECUPERACIÓN COLABORATIVA DE INFORMACIÓN EN ENTORNOS DISTRIBUIDOS», 2010.

[44] G. Salton y C. Buckley, «Term-weighting approaches in automatic text retrieval», *Information processing & management*, vol. 24, n.º 5, pp. 513–523, 1988.

[45] J. S. Wibowo y S. Hartati, «Text Document Retrieval In English Using Keywords of Indonesian Dictionary Based», *IJCCS-Indonesian Journal of Computing and Cybernetics Systems*, vol. 5, n.º 1, 2013.

[46] C. T. Meadow, *Text information retrieval systems*. Academic Press, Inc., 1992.

[47] C. Carrascosa, V. Julián, y J. Soler, «Una arquitectura de sistema multi-agente para la recuperación y presentación de la información», en *La representación y la organización del conocimiento en sus distintas perspectivas: su influencia en la recuperación de la información: actas del IV Congreso ISKO-España EOCONSID'99, 22-24 de abril de 1999, Granada*, 1999, pp. 291–296.

[48] S.-F. Chang, J. R. Smith, M. Beigi, y A. Benitez, «Visual information retrieval from

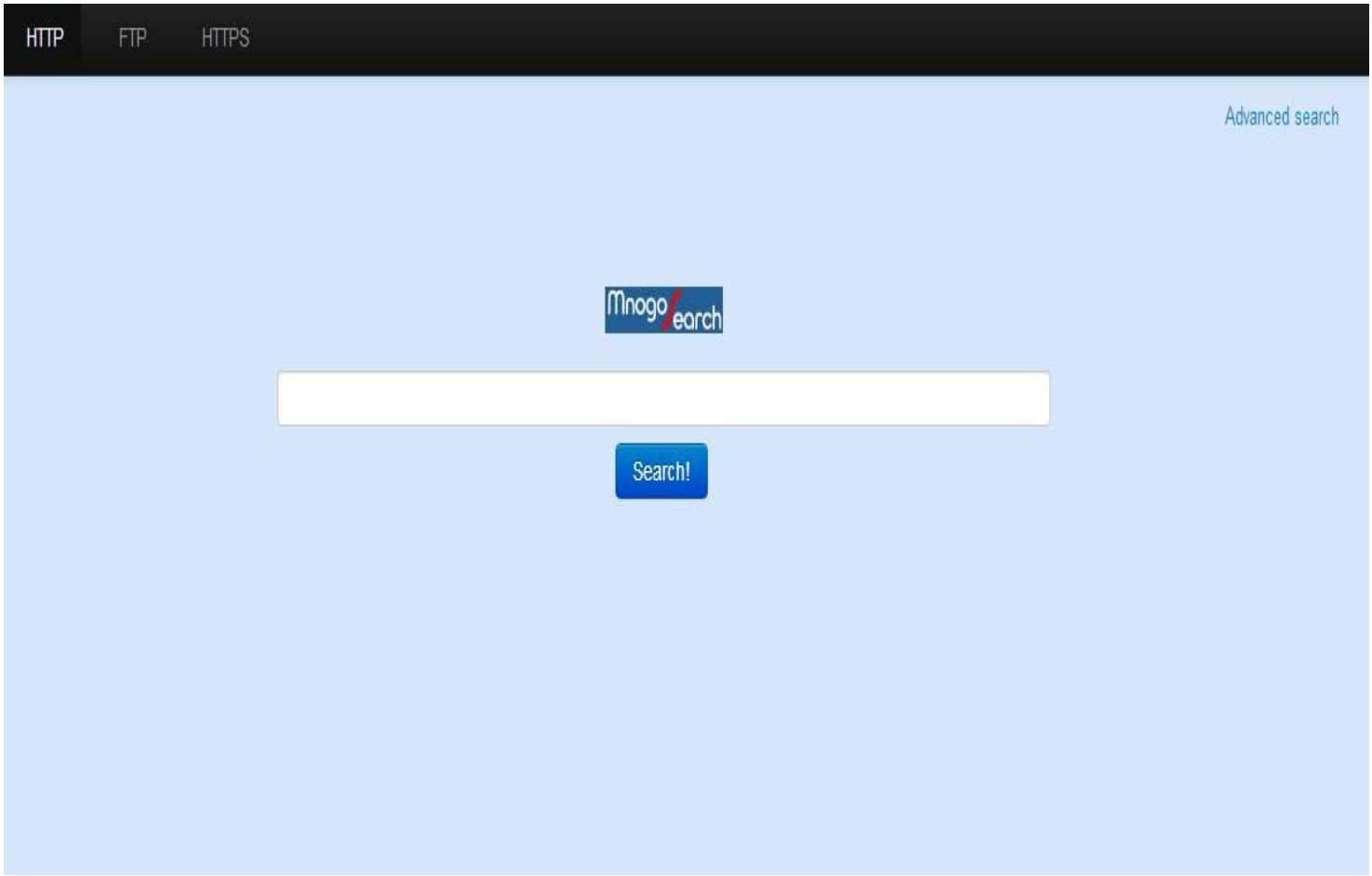
Bibliografía

large distributed online repositories», *Communications of the ACM*, vol. 40, n.º 12, pp. 63–71, 1997.

[49] P. I. Borkar y L. H. Patil, «Web information retrieval using genetic algorithm-particle swarm optimization», *International Journal of Future Computer and Communication*, vol. 2, n.º 6, pp. 595–599, 2013.

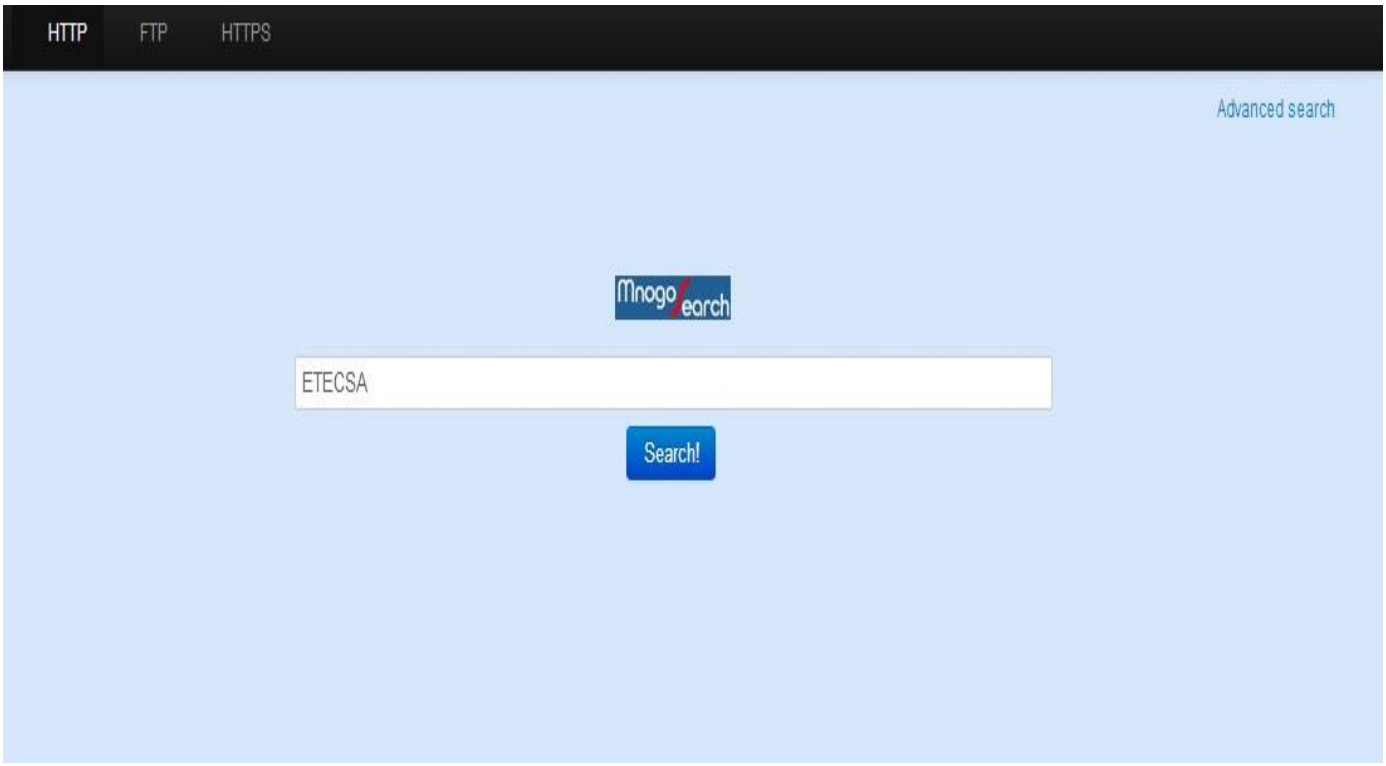
Anexos

Anexo 1: Portada



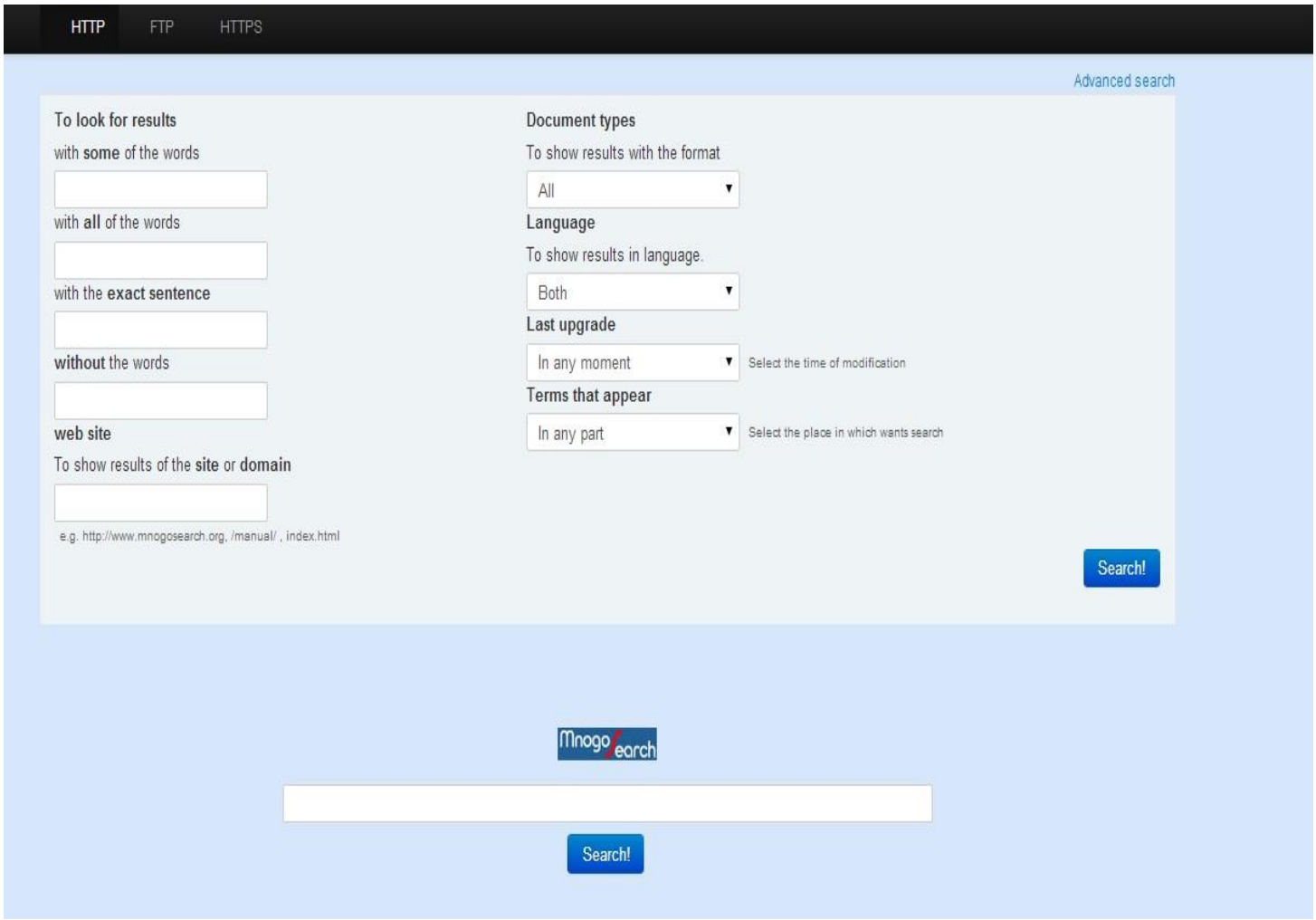
Anexos

Anexo 2: Portada



Anexos

Anexo 3: Portada con búsqueda avanzada



Anexos

Anexo 4: Resultados de búsqueda