Artículo de Revisión

Citación: Camargo, F. y Ordóñez, S. (2013). "Evolución y tendencias actuales de los Web *crawlers*". En: Ingeniería, Vol. 18, No. 2, pp. 19–35

Evolución y tendencias actuales de los Web crawlers

Web crawlers: Evolution and current trends

Fernando Iván Camargo Sarmiento

Estudiante MCIC-UD
Grupo GESDATOS
ficamargo@hotmail.com

Sonia Ordóñez Salinas

Docente MCIC-UD
Directora Grupo GESDATOS
soniaords@gmail.com

Resumen

La información disponible en redes de datos como la Web o las redes sociales se encuentra en continuo crecimiento, con unas características de dinamismo especiales. Entre los mecanismos encargados de rastrear los cambios en dicha información se encuentran los *Web crawlers*, los cuales por la misma dinámica de la información, deben mejorarse constantemente en busca de algoritmos más eficientes. Este documento presenta el estado actual de los algoritmos de rastreo de la *Web*, sus tendencias, avances, y nuevos enfoques dentro del contexto de la dinámica de las redes sociales.

Palabras clave: Procesamiento de Lenguaje Natural, rastreador, buscador, *Web crawler*, redes sociales, rastreador social.

Abstract

The information stored through the social network services is a growing source of information with special dynamic characteristics. The mechanisms responsible for tracking changes in such information (*Web crawlers*) often must be studied, and it is necessary to review and improve their algorithms. This document presents the current status of tracking algorithms of the Web (Web crawlers), its trends and developments, and its approach towards managing challenges emerging like social networks.

Key words: Natural Language Processing, crawler, search engine, Web crawler, social network, social network crawler.

1. Introducción

Los servicios de redes sociales actuales involucran a millones de usuarios en internet. Entre el año 2008 y el 2013, dicho número de usuarios en Latinoamérica ha pasado de 53.248.000 [1] a 139.000.000 [2], lo que representa un crecimiento de más del 250% en 4 años.

Fecha recibido: abr. 8/2013
Fecha modificado: nov. 6/2013
Fecha aceptado: nov. 29/2013

Debido a esto, la información almacenada a través de dichos servicios se convierte en una fuente creciente de información útil para la búsqueda de relaciones y patrones implícitos en ella; pueden verse como repositorios de información con unas características de dinamismo especiales. Con tales características, el hecho de explorar periódicamente el contenido y mantenerlo catalogado para su utilización representa un reto creciente.

Los mecanismos diseñados para tal fin son llamados *rastreadores***.** Son herramientas que permiten (como su nombre lo indica), rastrear un sitio web para extraer de él cualquier contenido existente. Los rastreadores hacen uso de la estructura de los documentos, sus etiquetas de hipertexto y sus meta-etiquetas para catalogar el contenido.

Sin embargo, los rastreadores que utilizan estos conceptos están diseñados para obtener datos generalizados, que por su dinámica pueden actualizar sus catálogos en periodos que oscilan entre días y semanas [3]. El tiempo promedio en que un usuario actualiza su información en sus servicios de redes sociales es menor a 3 días [4]. Esta dinámica implica la necesidad de técnicas de rastreo cada vez mas eficientes, enfocadas a ciertos temas e incorporando métodos de clasificación basados en el procesamiento de lenguaje allí contenido.

A continuación se presenta la revisión de dichos rastreadores (*Web crawlers*), sus tendencias y avances, y su enfoque hacia el manejo de redes sociales. Para ello, se ha consultado diferentes librerías académicas como Springer, IEEE y ACM, con búsqueda desde 1995 hasta 2013. A partir de una búsqueda de todos los trabajos sobre *crawlers*, rastreadores y arañas, se han revisado un promedio de 250 trabajos organizados por relevancia y por número de citaciones.

2. Contextualización

Para entender la evolución de los rastreadores actuales (especialmente los rastreadores focalizados que se explicarán más adelante) es necesario conocer el tema en el cual se enmarcan, es decir, el Procesamiento del Lenguaje Natural (PLN). En el siguiente capítulo se presentan los conceptos relevantes sobre dicho tema, y luego se aborda el estado del arte de los rastreadores Web, especificando sus objetivos, arquitecturas y avances.

2.1. Procesamiento de lenguaje natural

El término "procesamiento de lenguaje natural" es normalmente utilizado para describir la función de componentes de hardware o software en su sistema de cómputo, que analizan lenguaje hablado o escrito [5]. Combinado con la minería de datos, definida como la "tarea de identificar patrones de interés y describirlos de una forma concisa y con significado [6]", presenta variantes como la minería de texto y la minería Web.

2.1.1. Minería de texto y minería Web

La minería de texto es el proceso de extraer patrones interesantes a partir de grandes colecciones de textos para descubrir conocimientos [7]. Es también el descubrimiento de reglas de asociación importantes dentro de un corpus de texto [8].

En la Web, dicha minería podría ser aplicada a una serie de tareas como la recuperación de información (obtención de documentos ante una solicitud de búsqueda [9]), la extracción de información (búsqueda de información a partir de documentos previamente recuperados), resolución de pregunta – respuesta , entre otras, lo cual hace aparecer otra rama de investigación llamada minería Web, definida como el proceso global de descubrir información o conocimiento, potencialmente útil y previamente desconocido a partir de datos en la Web [10].

En la literatura también se encuentran a su vez tres subdivisiones de la minería Web [11]: Minería de contenido (analiza el contenido disponible en documentos Web), minería de estructura (se enfoca en la información vinculada, es decir, los enlaces entre documentos) y minería de uso (que analiza los datos transaccionales generados cuando los usuarios interactúan en la Web).

2.1.2. Los Rastreadores y el procesamiento del lenguaje natural

Uno de los grandes retos a la hora de utilizar la información existente en la Web es la recolección, clasificación y adaptación de la misma. Dichas tareas a su vez son propias del procesamiento del lenguaje natural.

Si bien es cierto que existen grandes volúmenes de información en los diferentes servidores Web listos para ser procesados, también es cierto que la consecución de dicha información actualizada y organizada no es una tarea fácil [12]. Generalmente para recolectar esta información se utilizan los *Web crawlers* que permiten "visitar" los diferentes repositorios Web de información y extraer lo que allí resida. Del producto de las "visitas" depende la calidad y fiabilidad de los resultados de la tarea en cuestión [13]. Para aclarar estos conceptos se expondrá una de las tareas que se han mencionado y su relación con los Web *crawlers*.

2.1.3. Recuperación de información

La recuperación de información consiste en buscar material de naturaleza no estructurada, que satisfaga una necesidad de información dentro de grandes colecciones [14]. Se ocupa del pre-procesamiento, la representación, el almacenamiento, la organización y el acceso a ítems de información [15]. Puede abarcar la localización y presentación a un usuario de información relevante a una necesidad de información expresada como una pregunta [16].

Los sistemas de recuperación de información pretenden determinar entonces qué contenido es relevante dado uno o varios criterios. Para ello, se deben tener en cuenta dos etapas o instancias [17]:

1. Elección de un modelo para calcular la relevancia de los documentos a la consulta. El modelo adoptado determina las predicciones sobre lo que es relevante (por ejemplo, la noción de relevancia implementada por el sistema). Su bondad se mide comparando las respuestas del sistema contra las que un conjunto de expertos consideran relevantes. Hay diferentes modelos para calcular la relevancia; entre ellos se pueden encontrar el modelo booleano (en el cual un documento puede pertenecer a dos únicas clases: "relevante" o "no relevante") [14], el modelo vectorial (en el cual, cada documento puede incluirse en una lista de valores cerrados, es decir que hay un número finito de clases mayor a 2) [18], probabilístico (el cual le asigna una ponderación o porcentaje de probabilidad a cada documento, indicando "la probabilidad de que un documento pertenezca a una clase determinada") [19] , y los modelos avanzados (conjuntos difusos [20], indexación por semántica latente [21], redes neuronales [22], entre otros)
2. Diseño de algoritmos y estructuras de datos que lo implementen (índices). Su bondad se mide considerando el tiempo de respuesta del sistema, espacio extra de los índices, tiempo de construcción y actualización del índice, entre otros [23].

2.1.4. Clasificación de texto

Los Web *crawler* - como se ya ha mencionado - permiten recolectar (copiar) información de diferentes servidores Web. Sin embargo, este tipo de información puede provenir de diferentes servicios (páginas, documentos *pdf*, redes sociales) que hacen que presenten diferentes formatos, lenguajes y estructuras [24]. Con el fin de poder homogenizar dicha información de acuerdo a un fin particular como la indexación o la extracción de patrones se hace necesario utilizar técnicas propias de la minería de texto como la clasificación.

Un clasificador es una técnica capaz de diferenciar elementos de acuerdo con sus características y agruparlos en órdenes o clases [25]. Estos algoritmos se pueden dividir en dos grandes grupos. Por un lado, se encuentran los que parten de un conjunto de datos para los que se desconocen las clases en las que se pueden agrupar (clasificación no supervisada). Por otro lado están los algoritmos de aprendizaje supervisado, en los que se dispone de un conjunto de datos con ejemplos de entrenamiento que han sido etiquetados previamente [26].

Dentro de los algoritmos de clasificación supervisada, que se utilizan habitualmente para clasificar texto, pueden encontrarse entre otros los algoritmos de SVM (*Support Vector Machine* o Máquinas de Vectores de Soporte [27]), ANN (Artificial Neural Networks o Redes Neuronales Artificiales [28]), Regresión [29], clasificadores Bayesianos (probabilísticos) [30], k-NN (clasificador de vecinos más cercano) [31] y árboles de decisión [27]. Cada uno de ellos presenta características diferentes de rendimiento según su uso [27].

2.2. Rastreadores (crawlers)

Para definir qué es un rastreador, es importante primero definir el entorno: para ello, se explica qué es un buscador Web, sus diferentes componentes y posteriormente se explicará en detalle el concepto de *crawler*.

2.2.1. Crawlers y buscadores Web

Un buscador Web es un sistema de recuperación de información en Internet, basado en páginas previamente catalogadas, y cuyos resultados son enlaces a las páginas reales que contengan ciertos parámetros o criterios. El buscador habitualmente toma como referencia meta-etiquetas de marcado como títulos, descripción o palabras clave dentro de los documentos, y con base en ello clasifican o ponderan los documentos. De esta forma, un documento con sus meta-etiquetas completas y con contenido relevante será mejor ponderado por un buscador web que un documento que no contenga tales etiquetas [32]. A esta optimización se le conoce como "Optimización para Motores de Búsqueda", o SEO por sus siglas en inglés (*Search Engine Optimization*) [33, 34].

Los creadores de Google, Larry Page y Sergei Brin [35] definen un buscador Web en función de cinco componentes: un Web *crawler*, un indexador, un ponderador (o algoritmo de ponderación), un motor de búsqueda y un repositorio de páginas. El *Web crawler es el componente responsable de descargar la información disponible en la Web hacia el repositorio del motor de búsqueda, para su procesamiento posterior.* El indexador genera un índice de términos, información del archivo y algunas otras características importantes de la información descargada por el Web *crawler*. El motor de búsqueda es responsable de procesar una consulta de usuario con una o más palabras y combinaciones de comodines y conectores lógicos. El ponderador es responsable de ordenar la mayoría de entradas del indexador. El repositorio de páginas es a menudo un medio físico para alojar una versión de las páginas rastreadas, en un formato útil para el motor de búsqueda.

2.2.2. Funcionamiento de un crawler

El objetivo principal de un *Web crawler* es proporcionar datos actualizados a un motor de búsqueda [36]. Son utilizados principalmente para crear una copia de todas las páginas rastreadas para su posterior procesamiento por un motor de búsqueda luego de ser indexadas para proporcionar resultados de una forma rápida [37]. Las metas de un *crawler* óptimo son su fácil escalabilidad, su habilidad de determinar qué contenido es susceptible de descarga y cuál se debe desechar, mantener su "responsabilidad social y ética" [38, 39], y su competencia directa con adversarios [40].

A nivel conceptual, el funcionamiento de un *crawler* es sencillo: tomar una dirección URL (o identificador de un sitio Web) a partir de una lista, descargar su contenido (sus páginas HTML), clasificarlo y aprovechar los enlaces de dichas páginas para hacer una

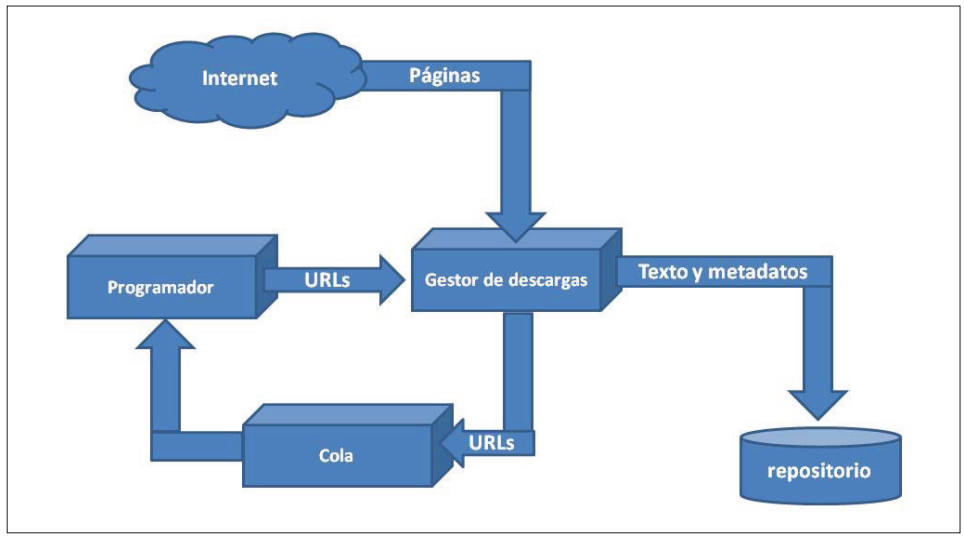

Figura 1. Esquema de funcionamiento de un *crawler* tradicional. Fuente: elaboración propia

nueva búsqueda con cada documento vinculado. A su vez, cada nuevo documento vinculado se clasifica nuevamente. En la figura 1 se puede observar cómo es realizado dicho proceso: un componente llamado "gestor de descargas" examina el contenido de un sitio web, crea un documento con sus metadatos y almacena el contenido en un repositorio. A su vez, busca en dicho sitio más enlaces o URLs, los cuales son enviados a una cola de espera para su procesamiento posterior. Por otro lado, hay un módulo llamado "programador", que se encarga de tomar los enlaces de la cola de espera para enviarlos al programador y realizar con él un nuevo proceso, llamado barrido de segundo nivel.

Sin embargo, debido al a cantidad de sitios web y la cantidad de páginas con que cuenta cada uno de ellos, un *crawler* debe considerar una forma rápida de seleccionar las páginas por descargar, y una forma óptima de verificar qué cambios han tenido dichas páginas a través del tiempo.

Para mantener un corpus actualizado, 10 mil millones de páginas en un estado razonable de actualización, por ejemplo de 4 semanas, el *crawler* debe descargar alrededor de 4000 páginas por segundo; para hacerlo, el *crawler* se debe distribuir sobre múltiples computadores y procesar las búsquedas en paralelo [3]. Por dicha velocidad de actualización, y por el crecimiento de internet planteado, el desarrollo de técnicas, algoritmos y arquitecturas ha sido constante y el tema permanece vigente.

Uno de los tipos de *crawlers* por estudiar son los denominados *crawlers* focalizados [41]. El principal atributo de los *crawlers* focalizados es que no necesitan coleccionar todas las páginas Web, sino que se enfocan en aquellas relevantes o importantes respecto de un conjunto predefinido de tópicos antes de comenzar a rastrear [42].

2.2.3. Evolución histórica

Los *crawlers* más conocidos son UbiCrawler [43], Viuva Negra [44], y el módulo de rastreo distribuido de Google [35], además de otros de naturaleza comercial previos a Google (Altavista, Infoseek, Lycos, Excite y HotBot). En cuanto a *crawlers* de código abierto, se destacan Heritrix [45], Nutch [46], Combine [47] y WIRE [48].

El primer *crawler* oficialmente reconocido es el "Wanderer" de Matthew Gray; fue un algoritmo de rastreo simple en la Web desarrollado en 1993 [49]. Fue presentado como un rastreador para el MIT, con el único propósito de generar estadísticas, y no fue publicado o expuesto a la comunidad científica. Posteriormente aparecen otros cinco en un periodo de dos años: JumpStation [50], RBSE [51], WebCrawler[52], WWWWorm [53] y MOMspider [54]. Dichos *crawlers* han dado origen a la mayoría de los actuales.

Sin embargo, el algoritmo de referencia obligatoria es PageRank, propuesto por Lawrence Page y Sergei Brin [35]; en él se expone cuál debería ser la estructura de un motor de búsqueda Web, incluyendo *crawler*, indexación y búsqueda. En un artículo posterior [55], resultado de un proyecto de investigación de la Universidad de Stanford, los autores describen públicamente este algoritmo que sería el fundamento de Google durante sus primeros diez años. En este documento, definen métricas importantes en el rastreo como similaridad de página (medida para descartar páginas iguales en la Web), "conteo regresivo" (para aumentar el peso a una página entre más referenciada esté en otras páginas), PageRank (ponderar mejor las páginas referenciadas por portales importantes), completitud de la información y ubicación (según el lugar del código en el que se encuentre un resultado). Igualmente presentan un algoritmo simple de rastreo basado en estos criterios.

En el 2002, se presenta un *crawler* llamado WebRACE [56], con capacidades de procesamiento distribuido, almacenamiento temporal de objetos y servicio de filtrado. Para ello, se utilizó un motor desarrollado por la Universidad de California, llamado eRACE, capaz de recolectar, anotar y diseminar información de fuentes heterogéneas. Los autores establecieron que en promedio, 1 de cada 10 documentos cambiaba luego de una semana, con lo que no era necesario reprocesarlo. Para evitar que el crawler descargase de nuevo páginas que no habían sido actualizadas, planteó adicionar a cada sitio web un "meta-documento" en XML con la información necesaria para ser descartado o reprocesado.

Otra propuesta que merece mencionarse es Ubicrawler, un *crawler* distribuido, programado en Java con todas las funciones descentralizadas, el cual es presentado por Boldi et al. en 2004 [57]. Ya que su algoritmo es distribuido, los autores reportan que en cada CPU en la cual se ejecute el crawler se pueden procesar hasta 660 páginas por segundo. Este trabajo fue actualizado en el 2009 [43] y [58] donde se expusieron aspectos para tener en cuenta en la optimización de sus resultados .

En ese mismo año se desarrolló un *crawler* para idioma español en Java y Oracle como repositorio de base de datos, capaz de extraer información de Facebook e ingresarla a un modelo relacional [59]. Este crawler actualmente no permite extraer la información debido a los cambios de seguridad implementados pro Facebook desde entonces.

En el año 2010 hubo una serie de trabajos alrededor de métodos de rastreo; por ejemplo Tadapak [60] propone un *crawler* para un idioma específico (Thai), con métodos supervisados. Shaojie y otros [61] proponen una mejora al algoritmo de PageRank utilizando medidas de similaridad, bajo el nombre de SimRank. Qureshi y otros autores [62] diseñan un *crawler* llamado "visionerBOT". Su principal aporte es el uso de MAPreduce [63], el *framework* de Google para manejo de computación distribuida.

También en 2010, se presenta un proyecto de *crawler* focalizado en base de datos de tópicos (DTB) [64]. Una de las propuestas del trabajo es que se cuente con una base de tópicos estáticos, y una base de tópicos dinámicos con auto-aprendizaje.

En el año 2011, Anbukodi [65] propone el uso de "agentes móviles" para reducir la sobrecarga de máquinas de los *crawlers* actuales. La arquitectura sugiere que cada agente cuente con un conjunto de páginas iniciales (llamadas "semilla") y rastree los vínculos que contenga dicha página recursivamente (cada página contiene vínculos de un nuevo nivel, los cuales contienen a su vez vínculos de un segundo nivel y asís sucesivamente); cuando un vínculo tenga una gran cantidad de niveles debajo de él, el agente tiene la capacidad de pasar dicho vínculo a un nuevo agente, liberando recursos y balanceando su carga.

2.2.3.1. Crawlers focalizados

Este concepto fue introducido por Chakrabarti en 1999 [66], y ha sido ampliamente utilizado en los últimos años como herramienta para rastrear las web para problemas específicos. Un *crawler focalizado (* o *focused crawler* por su nombre original*) es un tipo de* crawler que recibe uno o varios parámetros de entrada (como frases o palabras) y rastrea la web para localizar sitios con contenido relevante a dichos términos. La característica principal de los *crawlers* focalizados es que no necesitan coleccionar todas las páginas web visitadas, sino que se enfocan en URLs relevantes o importantes respecto de un conjunto predefinido de tópicos antes de comenzar a rastrear [42].

Puesto que los crawlers focalizados se utilizan para seleccionar contenido sobre un tema en particular, es usual que incorporen técnicas de clasificación y conceptos tradicionales del procesamiento de lenguaje natural. Por esto mismo, es importante tener en cuenta los retos que implica el uso de lenguaje humano, como la dependencia del idioma, el tratamiento de excepciones y la desambiguación de términos, entro otros [5, 9, 67], que son temas tradicionales en el procesamiento de lenguaje natural.

A partir de 2007 los trabajos sobre crawlers focalizados han sido frecuentes. En ellos ha habido diferentes modelos y técnicas de clasificación como el bayesiano ([68]), k-

Nearest Neighbours [31], modelos de asociación semántica [42], en ontologías [69] y algunos mezclando diferentes modelos como clasificación Bayesiana con el algoritmos de lógica difusa [70]. Los más utilizados por la precisión y exactitud son aquellos basado en Máquinas de Vectores de Soporte (SVM). A nivel de arquitectura, se han utilizado principalmente sistemas multiagente [71]. En términos generales, los *crawlers* expuestos en las diferentes librerías consultadas se basan en idioma inglés.

2.2.3.2. Crawlers sociales

Existen estudios sobre Web *crawlers* dirigidos al contenido de redes sociales, tales como el propuesto por S. Ibrahim [72]; aunque el objetivo de dicho estudio era el desarrollo de una red social, propone utilizar agentes *multicrawler* o MCA (Multi *Crawler* Agents). Los autores centran el trabajo en la extracción de información relevante para los negocios, pero no utilizan un crawler focalizado para ello. Lo que hacen es limitar los enlaces del estudio, de modo que solamente se consulten ciertas páginas web previamente seleccionadas. Con ello, el índice de precisión en la búsqueda posterior, es del 100% al consultar 100 documentos, y del 93% con 500 documentos.

Este mismo año se publica un artículo sobre técnicas de minería Web para redes sociales [73], Los autores hacen una revisión documental sobre el tema, y clasifican dichas técnicas en 3 diferentes tipos según su uso: Minería de contenido web (se analiza el contenido de la web como textos y gráficos y se enfoca en el procesamiento de texto), minería de estructuras web (se centra en el análisis de la estructura de los sitios web a partir de los enlaces de los sitios) y minería de uso web (se basa en analizar cómo son utilizados los sitios web, es decir, analizar el comportamiento de los usuarios cuando visitan cada sitio). Los autores concluyen que un *crawler* para redes sociales debería estar construido con la técnica de "minería de estructuras Web" debido a que la comprensión de la estructura de los sitios podría mejorar la forma en que los crawlers sociales se construyen.

Fard, [74] introduce el concepto de "minería colaborativa", en un proyecto para descubrir patrones de grupos criminales dentro de las redes sociales; utiliza un sistema multiagentes, con un reglas de asociación basadas en el algoritmo "*a priori*" (el algoritmo a priori es uno de los 10 algoritmos más conocidos en la minería de datos según la IEEE hasta el 2007 [75] , y consiste en la creación de clases candidatas basadas en agrupamiento dinámico). El trabajo es presentado en documentos en inglés.

2.2.4. Métricas

Aún cuando los buscadores tienen métricas de evaluación de acuerdo con la confiabilidad de sus resultados, los *crawlers* no tienen una definición comúnmente aceptada sobre la forma de ser evaluados. En los *crawlers* tradicionales hay trabajos que proponen parámetros de medición como la exactitud o "*accuracy*" (número de fallas / número de fallas corregidas) y la disponibilidad o "*availability*" (exactitud en el tiempo) [76] pero estos parámetros no son utilizados en la mayoría de estudios. Sin embargo, en los *crawlers* focalizados sí hay

una aceptación generalizada de dos métricas de evaluación: a partir de 2004, Menczer [77] propuso utilizar los índices de exhaustividad (*recall)* y precisión (*precision)* [9] en ellos.

El índice de recall es la proporción de documentos recuperados que son relevantes para la búsqueda, y su índice viene dado por la siguiente fórmula [5]:

$$recall = \frac{|\{documentos\_relevantes\} \cap \{documentos\_recuperados\}|}{\{documentos\_recuperados\}} \quad (1)$$

Dicho índice busca determinar qué porcentaje de los documentos recuperados son realmente relevantes, debido a que cualquier modelo de recuperación de información puede retornar documentos que no son relevantes a juicio de un experto en el tema, y cuanto mayor sea el porcentaje de documentos relevantes, más preciso será el método estudiado.

El índice de precisión es la proporción de documentos relevantes a la consulta que se han recuperado satisfactoriamente, y su índice viene determinado por la siguiente fórmula:

$$precisión = \frac{|\{documentos\_relevantes\} \cap \{documentos\_recuperados\}|}{\{documentos\_relevantes\}} \quad (2)$$

El índice de *precision* busca determinar cuántos documentos relevantes han sido excluidos de los resultados. Cuando este índice se utiliza combinado con el índice de *recall*, se puede determinar la efectividad de determinado modelo.

2.2.5. Arquitecturas

La mayoría de sistemas de recuperación de información en la Web se basan en las siguientes arquitecturas [17] [35]:

- **Arquitectura centralizada**: está conformada por el *crawler,* un indexador (mantiene un índice de las páginas encontradas), máquina de búsqueda (interfaz con la que interactúa el usuario), un repositorio de páginas y un ponderador (quien se encarga de elegir los resultados a partir del repositorio, y enviarlos a la máquina de búsqueda). El funcionamiento de un buscador centralizado puede verse en la figura 2.

- **Arquitectura distribuida**: es una versión mejorada dela arquitectura centralizada. Cuenta con dos elementos adicionales (robots): los *gatherers* que extraen la información a recopilar periódicamente y los *brokers* encargados de indexar la información recopilada por los *gatherer* y otros *brokers*. Permite compartir el trabajo y evita transmitir mucha información. Su funcionamiento puede verse en la figura 3.

Los buscadores con arquitectura centralizada se utilizan en entornos con capacidades limitadas; por ejemplo en proyectos que residen en una sola máquina. Los buscadores

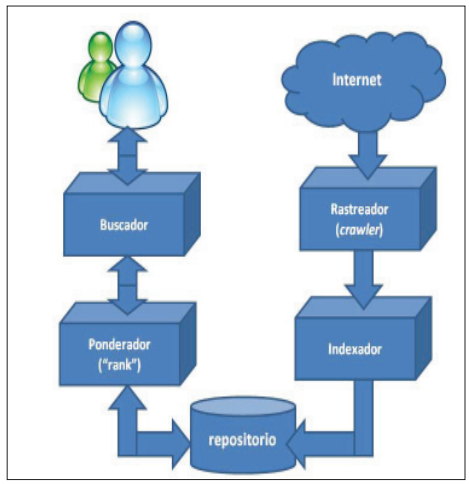

Figura 2. Arquitectura de un buscador centralizado. Fuente: elaboración propia.

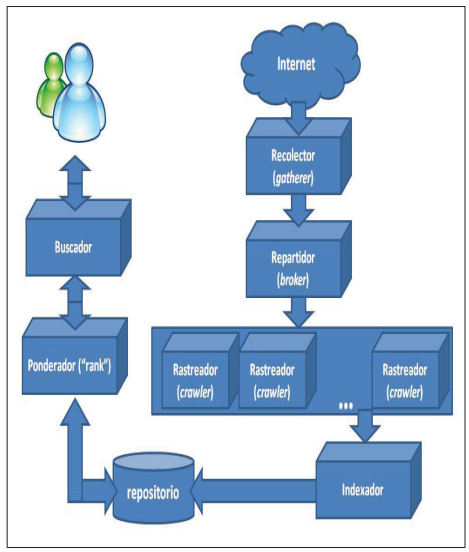

Figura 3. Arquitectura de un buscador distribuido. Fuente: elaboración propia.

con arquitectura distribuida pueden ser alojados en diferentes servidores, y es habitual que utilicen agentes para gestionar cada rastreador. Uno de ellos es el propuesto por Risvik, como tesis doctoral [78], quien habla de un *crawler* escalable, encargada de repartir el trabajo en cada uno de los nodos del *crawler*, de modo que puedan trabajar en paralelo.

En 2008, [79] se plantea utilizar dos componentes: un servidor para despertar cada proceso de *crawling* como un hilo (servidor multi-hilo) y *crawlers* cliente, cada una de las cuales se encuentra en su propio equipo haciendo rastreo en una parte de la Web. La técnica no difiere de los ya tradicionales *crawlers* en paralelo, pero ofrece algoritmos de sincronización entre cada cliente. Igualmente sugiere un único *crawler* ejecutándose sobre cada equipo. Dos años más tarde, Horowitz & Kamvar [80] hacen una analogía con el algoritmo propuesto por Page y Brin, pero incorporando un concepto básico de redes sociales: las personas. Además de catalogar el contenido de las páginas, localiza "entidades nombradas" (busca personas en cada página), y las relaciona con el contenido de la página. De esta forma, cuando el motor de búsqueda recibe una petición, el algoritmo intenta localizar "qué persona podría contestar dicha petición", asignando relevancia a los documentos que tengan personas relacionadas, e incluyendo en la búsqueda otros documentos de dicha persona. En este trabajo se menciona explícitamente el concepto de "*social crawling*".

También en 2010, se puede encontrar una propuesta de arquitectura de *crawler* desarrollada por Hsieh, Gribble & Levy [81]. En ella se indica que cada página debe ser tratada con dos posibles algoritmos: enfocado o generalizado. Si la página es por ejemplo un blog, o una página con una serie de enlaces, debe generalizarse, adicionando la tarea dentro de una lista de tareas controladas por un agente llamado "programador", pero si

es un sistema frecuente como una fuente RSS (tecnología para intercambio de noticias en formato XML) o un portal de noticias de actualización frecuente, debe utilizarse el algoritmo focalizado, y mantener el hilo abierto para los hijos de dicho enlace; El autor plantea que la búsqueda de segundo nivel no se ejecute directamente después de la primera, sino que en principio se agrupe y se determine qué hacer con cada uno de los enlaces, y según se decida, programarlos en la cola (cabe recordar que un *crawler* examina un sitio web y posteriormente examina los enlaces que contenga, y así sucesivamente por múltiples iteraciones llamadas niveles).

Igualmente en 2010, se presenta una mejora a la arquitectura en paralelo [82] por medio de un *bróker* que asigne dinámicamente las URL a cada hilo del *crawler*. Lo novedoso del trabajo es que dicha asignación se hace mediante lógica difusa.

Hay trabajos sobre arquitecturas extensibles [81]; estas arquitecturas buscan permitir que el *crawler* pueda crecer con el tiempo a varios equipos o nodos sin que ello implique cambios en los algoritmos, sino que contengan técnicas de distribución de carga flexibles para su crecimiento posterior.

3. Conclusiones

La investigación sobre rastreadores y minería Web en el ámbito mundial es actual, y el interés se ha incrementado en los últimos años, como se desprende del número de publicaciones sobre este tema; la mayoría de trabajos encontrados han sido desarrollados desde el año 2008 al 2012, variando técnicas, métodos y resultados.

Los rastreadores como elementos de exploración se han direccionado principalmente a la búsqueda de documentos *html*; hay diferentes tipos de rastreadores, según su nivel de especificidad, arquitectura o algoritmo de rastreo. La tendencia en los últimos años ha sido la de especializar la búsqueda, y para ello han surgido los *crawlers* focalizados. La cantidad de información en Internet y la velocidad de actualización de la misma, ha hecho que sea necesario este tipo de rastreo.

Pese a que la arquitectura distribuida ha sido adoptada debido al crecimiento que puede tener un buscador, y a que el uso de agentes ha sido una técnica utilizada para lograrlo, no hay más temas que hayan sido adoptados como norma en la creación de *crawlers*. En la actualidad no existe un método generalizado de rastreo, y tampoco hay consenso en cómo se debe abordar el procesamiento de lenguaje natural cuando se trata de *crawlers* focalizados. De hecho, hay una gran debilidad en cuanto al manejo de diferentes idiomas, y las técnicas y metodologías dependen en muchos casos del problema específico a tratar con dicho *crawler*.

Si bien el tema ha sido ampliamente explorado, no solo por la comunidad académica sino por las grandes multinacionales del software, es claro que los *crawlers* aún no presentan los resultados esperados. Este hecho se refleja, por ejemplo, en que la mayoría de los buscadores incluyen dentro de sus resultados, contenidos iguales bajo diferentes direcciones (urls), fuentes que no se corresponden al contexto deseado por el usuario

y resultados que direccionan a sitios que han dejado de existir. Situaciones que pueden obedecer a cualquiera de los componentes propios de los *crawlers* como son los métodos de indexación (ya que se repite una misma fuente), los métodos de recorrido, la frecuencia del recorrido, los métodos de proximidad para saber si dos páginas o elementos digitales son iguales, o los métodos para identificar cual es la fuente real. Lo anterior permite concluir que el tema de los *crawlers* amerita continuar su investigación.

Referencias bibliográficas

[1] A. Lipsman, "Social Networking Explodes Worldwide as Sites Increase their Focus on Cultural Relevance," comScore2008.

[2] A. L. Zain, "Futuro digital Latinoamérica 2013: El estado actual de la industria digital y las tendencias que están modelando el futuro," ComScore2013.

[3] M. Najork, "Web Crawler Architecture," *Encyclopedia of Database Systems,* 2009.

[4] J. Seguic, "El Crecimiento de Redes Sociales en América Latina: La Influencia de Los Medios Sociales en el Escenario Digital de América Latina. Septiembre 2011.," ComScore2011.

[5] P. Jackson and I. Moulinier, *Natural language processing for online applications: text retrieval, extraction and categorization*: John Benjamins Pub., 2007.

[6] G. Piatetsky-Shapiro and W. Frawley, *Knowledge discovery in databases*: AAAI Press, 1991.

[7] A. H. Tan, "Text Mining: promises and challenges," *South East Asia Regional Computer Confederation, Sigapore,* 1999.

[8] M. Delgado, N. Marin, D. Sanchez, and M. A. Vila, "Fuzzy association rules: general model and applications," *Fuzzy Systems, IEEE Transactions on,* vol. 11, pp. 214-225, 2003.

[9] C. D. Manning and H. Schütze, *Foundations of statistical natural language processing*: MIT Press, 1999.

[10] O. Etzioni, "The World-Wide Web: quagmire or gold mine?," *Commun. ACM,* vol. 39, pp. 65-68, 1996.

[11] S. K. Madria, S. S. Bhowmick, W. K. Ng, and E.-P. Lim, "Research Issues in Web Data Mining," presented at the Proceedings of the First International Conference on Data Warehousing and Knowledge Discovery, 1999.

[12] Springer-Verlag, Ed., *Advances in Web Mining and Web Usage Analysis: 9th International Workshop on Knowledge Discovery on the Web, WebKDD 2007, and 1st International Workshop on Social Networks Analysis, SNA-KDD 2007, San Jose, CA, USA, August 12-15, 2007. Revised Papers*. Springer-Verlag, 2009, p.^pp. Pages.

[13] K. Oyama, H. Ishikawa, K. Eguchi, and A. Aizawa, "Analysis of Topics and Relevant Documents for Navigational Retrieval on the Web," in *Web Information Retrieval and Integration, 2005. WIRI '05. Proceedings. International Workshop on Challenges in*, 2005, pp. 157-163.

[14] C. Manning, Raghavan, P., Schütze, H, *An introduction to information retrieval*, 2009.

[15] R. R.-N. Baeza-Yates, Berthier, *Modern Information Retrieval*: Addison-Wesley Longman Publishing Co., Inc. , 1999.

[16] R. R. Korfhage, *Information storage and retrieval*: Wiley Computer Pub., 1997.

[17] E. Lorenzo. (2005, 2011-10-01). Recuperación de información basada en contenido. Material de estudio, Doctorado en Sistemas Software inteligentes y adaptables. Available: http://trevinca.ei.uvigo.es/~evali/doctorado0507/sri/

[18] M. W. Berry, Z. Drmac, and E. R. Jessup, "Matrices, vector spaces, and information retrieval," *SIAM review,* vol. 41, pp. 335-362, 1999.

[19] M. E. Maron and J. L. Kuhns, "On relevance, probabilistic indexing and information retrieval," *Journal of the ACM (JACM),* vol. 7, pp. 216-244, 1960.

[20] G. Bordogna and G. Pasi, "A fuzzy linguistic approach generalizing boolean information retrieval: A model and its evaluation," *JASIS,* vol. 44, pp. 70-82, 1993.

[21] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman, "Indexing by latent semantic analysis," *JASIS,* vol. 41, pp. 391-407, 1990.

[22] E. Wiener, J. O. Pedersen, and A. S. Weigend, "A neural network approach to topic spotting," in *Proceedings of SDAIR-95, 4th annual symposium on document analysis and information retrieval*, 1995, pp. 317-332.

[23] H. Schütze, D. A. Hull, and J. O. Pedersen, "A comparison of classifiers and document representations for the routing problem," in *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, 1995, pp. 229-237.

[24] L. Hao, R. Fei, and Z. Wanli, "The Preliminary Process of Modeling in Deep Web Information Fusion System," in *Information Technology and Applications, 2009. IFITA '09. International Forum on*, 2009, pp. 723-726.

[25] G. Martinez, "Clasificación mediante Conjuntos," Tesis Doctoral, Departamento de Ingeniería Informática, Universidad Autónoma de Madrid, 2006.

[26] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification*: John Wiley & Sons, 2012.

[27] R. Caruana and A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," presented at the Proceedings of the 23rd international conference on Machine learning, Pittsburgh, Pennsylvania, 2006.

[28] M. A. Sovierzoski, F. I. M. Argoud, and F. M. de Azevedo, "Evaluation of ANN Classifiers During Supervised Training with ROC Analysis and Cross Validation," in *BioMedical Engineering and Informatics, 2008. BMEI 2008. International Conference on*, 2008, pp. 274-278.

[29] A. Khemphila and V. Boonjing, "Comparing performances of logistic regression, decision trees, and neural networks for classifying heart disease patients," in *Computer Information Systems and Industrial Management Applications (CISIM), 2010 International Conference on*, 2010, pp. 193-198.

[30] R. S. Feldman, J, *The Text Mining Handbook*. New York: Cambridge University Press, 2006.

[31] Z. Lijuan, W. Linshuang, G. Xuebin, and S. Qian, "A clustering-Based KNN improved algorithm CLKNN for text classification," in *Informatics in Control, Automation and Robotics (CAR), 2010 2nd International Asia Conference on*, 2010, pp. 212-215.

[32] Google. (2011, 10/12/2012). Guía para principiantes sobre optimización para motores de búsqueda. Available: https://www.google.es/webmasters/docs/guia_optimizacion_motores_busqueda.pdf

[33] Z. Chengling, L. Jiaojiao, and D. Fengfeng, "Application and Research of SEO in the Development of Web2.0 Site," in *Knowledge Acquisition and Modeling, 2009. KAM '09. Second International Symposium on*, 2009, pp. 236-238.

[34] D. Wu, T. Luan, Y. Bai, L. Wei, and Y. Li, "Study on SEO monitoring system based on keywords and links," in *Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on*, 2010, pp. 450-453.

[35] S. Brin and L. Page, "The Anatomy of a Large-Scale Hypertextual Web Search Engine," *Proceedings of the Seventh World-Wide Web Conference,* 1998.

[36] P. Gupta and K. Johari, "Implementation of Web Crawler," in *Emerging Trends in Engineering and Technology (ICETET), 2009 2nd International Conference on*, 2009, pp. 838-843.

[37] M. Abdeen and M. F. Tolba, "Challenges and design issues of an Arabic web crawler," in *Computer Engineering and Systems (ICCES), 2010 International Conference on*, 2010, pp. 203-206.

[38] S. Yang, I. G. Councill, and C. L. Giles, "The Ethicality of Web Crawlers," in *Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on*, 2010, pp. 668-675.

[39] L. Van Wel and L. Royakkers, "Ethical issues in web data mining," *Ethics and Inf. Technol.,* vol. 6, pp. 129-140, 2004.

[40] C. Olston and M. Najork, "Web Crawling," *Foundations and Trends in Information Retrieval,* vol. 4, pp. 175-246, 2010.

[41] Y. Y. Yuekui, Du; Yufeng, Hai; Zhaoqiong, Gao, "A Topic-Specific Web Crawler with Web Page Hierarchy Based on HTML Dom-Tree," in *Information Processing, 2009. APCIP 2009. Asia-Pacific Conference on*, 2009, pp. 420-423.

[42] H. Rui, L. Fen, and S. Zhongzhi, "Focused Crawling with Heterogeneous Semantic Information," in *Web Intelligence and Intelligent Agent Technology, 2008. WI-IAT '08. IEEE/WIC/ACM International Conference on*, 2008, pp. 525-531.

[43] P. Boldi, B. Codenotti, M. Santini, and S. Vigna, "UbiCrawler: a scalable fully distributed Web crawler," *Software: Practice and Experience,* vol. 34, pp. 711-726, 2009.

[44] J. M. Exposto, J. , A. Pina, A. Alves, and J. Rufino. (2005, Geographical Partition for Distributed Web Crawling. *GIR '05: Proc. of the Geographic Information Retrieval,* 55–60.

[45] H. Jinzhu, Z. Xing, S. Jiangbo, X. Chunxiu, and Z. Jun, "Research of Active Information Service System Based on Intelligent Agent," in *Education Technology and Computer Science, 2009. ETCS '09. First International Workshop on*, 2009, pp. 837-841.

[46] Y. Guojun, X. Xiaoyao, and L. Zhijie, "The design and realization of open-source search engine based on Nutch," in *Anti-Counterfeiting Security and Identification in Communication (ASID), 2010 International Conference on*, 2010, pp. 176-180.

[47] A. G. Ardo, Koraljka, "Documentation for the Combine (focused) crawling system," 2009.

[48] R. C. Baeza-Yates, Carlos, "WIRE: an Open-Source Web Information Retrieval Environment," *Workshop on Open Source Web Information Retrieval (OSWIR),* pp. 27-30, Compiegne, France 2005.

[49] M. Gray. (1993). Wanderer. Growth and Usage of the Web and the Internet. Available: http://www.mit.edu/people/mkgray/growth/

[50] T. Seymour, D. Frantsvog, and S. Kumar, "History Of Search Engines," *International Journal of Management & Information Systems – Fourth Quarter 2011,* vol. 15, pp. 47-58, 2011.

[51] D. Eichmann, "The RBSE spider - Balancing effective search against web load," in *Proceedings of the First International World Wide Web Conference*, Ginebra- Suiza, 1994.

[52] B. Pinkerton, "Finding what people want: Experiences with the WebCrawler," in *Proceedings of the 2nd International World Wide Web Conference*, 1994.

[53] O. McBryan, "GENVL and WWWW: Tools for taming the web," in *Proceedings of the First International World Wide Web Conference*, Ginebra- Suiza, 1994.

[54] R. Fielding, "Maintaining distributed hypertext infostructures: Welcome to MOMspider's web," in *Proceedings of the First International World Wide Web Conference*, Ginebra- Suiza, 1994.

[55] J. Cho, H. Garcia-Molina, and P. Lawrence, "Efficient crawling through URL ordering," *Proceedings of the seventh international conference on World Wide Web 7 (WWW7), Amsterdam, The Netherlands,* pp. 161-172, 1998.

[56] D. Zeinalipour-Yazti and M. D. Dikaiakos, "Design and Implementation of a Distributed Crawler and Filtering Processor," presented at the Proceedings of the 5th International Workshop on Next Generation Information Technologies and Systems, 2002.

[57] P. Boldi, B. Codenotti, M. Santini, and S. Vigna, "Ubicrawler: A scalable fully distributed web crawler," *Software: Practice and Experience,* vol. 34, pp. 711-726, 2004.

[58] P. S. Boldi, Massimo; Vigna, Sebastiano, "Do Your Worst to Make the Best: Paradoxical Effects in PageRank Incremental Computations," *Algorithms and Models for the Web-Graph,* pp. 168-180, 2009.

[59] A. Del Coso Santos, "Desarrollo de infraestructuras para el modelado de usuarios," Universidad Carlos III de Madrid, 2009.

[60] P. Tadapak, T. Suebchua, and A. Rungsawang, "A Machine Learning Based Language Specific Web Site Crawler," in *Network-Based Information Systems (NBiS), 2010 13th International Conference on*, 2010, pp. 155-161.

[61] Q. Shaojie, L. Tianrui, L. Hong, Z. Yan, P. Jing, and Q. Jiangtao, "SimRank: A Page Rank approach based on similarity measure," in *Intelligent Systems and Knowledge Engineering (ISKE), 2010 International Conference on*, 2010, pp. 390-395.

[62] M. A. Qureshi, A. Younus, and F. Rojas, "Analyzing the Web Crawler as a Feed Forward Engine for an Efficient Solution to the Search Problem in the Minimum Amount of Time through a Distributed Framework," in *Information Science and Applications (ICISA), 2010 International Conference on*, 2010, pp. 1-8.

[63] J. Dean and S. Ghemawat, "MapReduce: simplified data processing on large clusters," in *Proceedings of the Sixth Symposium on Operating Systems Design and Implementation*, San Francisco, California, 2004, pp. 137-150.

[64] Z. Ming-sheng, Z. Peng, and H. Tian-chi, "An Intelligent Topic Web Crawler Based on DTB," in *Web Information Systems and Mining (WISM), 2010 International Conference on*, 2010, pp. 84-86.

[65] S. Anbukodi and K. M. Manickam, "Reducing web crawler overhead using mobile crawler," in *Emerging Trends in Electrical and Computer Technology (ICETECT), 2011 International Conference on*, 2011, pp. 926-932.

[66] S. Chakrabarti, M. Van den Berg, and B. Dom, "Focused crawling: a new approach to topic-specific Web resource discovery," *Computer Networks,* vol. 31, pp. 1623-1640, 1999.

[67] L. Peng, W. Xiao Long, G. Yi, and Z. Yu Ming, "Extracting answers to natural language questions from large-scale corpus," in *Natural Language Processing and Knowledge Engineering, 2005. IEEE NLP-KE '05. Proceedings of 2005 IEEE International Conference on*, 2005, pp. 690-694.

[68] W. Wenxian, C. Xingshu, Z. Yongbin, W. Haizhou, and D. Zongkun, "A Focused Crawler Based on Naive Bayes Classifier," in *Intelligent Information Technology and Security Informatics (IITSI), 2010 Third International Symposium on*, 2010, pp. 517-521.

[69] D. Mukhopadhyay, A. Biswas, and S. Sinha, "A New Approach to Design Domain Specific Ontology Based Web Crawler," in *Information Technology, (ICIT 2007). 10th International Conference on*, 2007, pp. 289-291.

[70] Z. Qiang, "An Algorithm OFC for the Focused Web Crawler," in *Machine Learning and Cybernetics, 2007 International Conference on*, 2007, pp. 4059-4063.

[71] J. Akilandeswari and N. P. Gopalan, "An Architectural Framework of a Crawler for Locating Deep Web Repositories Using Learning Multi-agent Systems," in *Internet and Web Applications and Services, 2008. ICIW '08. Third International Conference on*, 2008, pp. 558-562.

[72] S. N. A. Ibrahim, A. Selamat, and M. H. Selamat, "Scalable e-business social network using MultiCrawler agent," in *Computer and Communication Engineering, 2008. ICCCE 2008. International Conference on*, 2008, pp. 702-706.

[73] I. H. Ting, W. Hui-Ju, and C. Pei-Shan, "Analyzing Multi-source Social Data for Extracting and Mining Social Networks," in *Computational Science and Engineering, 2009. CSE '09. International Conference on*, 2009, pp. 815-820.

[74] A. M. Fard and M. Ester, "Collaborative Mining in Multiple Social Networks Data for Criminal Group Discovery," in *Computational Science and Engineering, 2009. CSE '09. International Conference on*, 2009, pp. 582-587.

[75] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, and S. Y. Philip, "Top 10 algorithms in data mining," *Knowledge and Information Systems,* vol. 14, pp. 1-37, 2008.

[76] M. Nasri, S. Shariati, and M. Sharifi, "Availability and Accuracy of Distributed Web Crawlers: A Model-Based Evaluation," in *Computer Modeling and Simulation, 2008. EMS '08. Second UKSIM European Symposium on*, 2008, pp. 453-458.

[77] F. Menczer, G. Pant, and P. Srinivasan, "Topical web crawlers: Evaluating adaptive algorithms," *ACM Transactions on Internet Technology (TOIT),* vol. 4, pp. 378-419, 2004.

[78] K. M. M. Risvik, R, "Scaling Internet Search Engines-Methods and Analysis," Disertation in Doctorade of degree, Norwegian University of Science and Technology, 2004.

[79] D. Yadav, A. K. Sharma, J. P. Gupta, N. Garg, and A. Mahajan, "Architecture for Parallel Crawling and Algorithm for Change Detection in Web Pages," in *Information Technology, (ICIT 2007). 10th International Conference on*, 2007, pp. 258-264.

[80] D. Horowitz and S. D. Kamvar, "The anatomy of a large-scale social search engine," presented at the Proceedings of the 19th international conference on World wide web, Raleigh, North Carolina, USA, 2010.

[81] J. M. Hsieh, S. D. Gribble, and H. M. Levy, "The architecture and implementation of an extensible web crawler," presented at the Proceedings of the 7th USENIX conference on Networked systems design and implementation, San Jose, California, 2010.

[82] A. Guerriero, F. Ragni, and C. Martines, "A dynamic URL assignment method for parallel web crawler," in *Computational Intelligence for Measurement Systems and Applications (CIMSA), 2010 IEEE International Conference on*, 2010, pp. 119-123.

Fernando Iván Camargo Sarmiento

Ingeniero de Sistemas Especialista en gestión de Sistemas y Tecnologías de la Información, estudiante MSc. Ciencias de la Información y las Comunicaciones y miembro del grupo de investigación GESDATOS en la Universidad Distrital Francisco José de Caldas en Bogotá. e-mail: ficamargo@hotmail.com

Sonia Ordóñez Salinas

Ingeniera de Sistemas, PhD. Ingeniería de Sistemas y Computación Universidad Nacional de Colombia, actualmente docente de la Maestría en Ciencias de la Información y las Telecomunicaciones y directora del grupo de investigación Gesdatos en la Universidad Distrital Francisco José de Caldas en Bogotá. e-mail: soniaords@gmail.com