CENTRO POLITÉCNICO SUPERIOR
UNIVERSIDAD DE ZARAGOZA

INGENIERÍA EN INFORMÁTICA

PROYECTO DE FIN DE CARRERA

Desarrollo de una herramienta de planificación social media en la blogosfera española

Autor:
Pablo Aragón Asenjo

Director:
Íñigo García Morte

Ponente:
Dr. Fernando Tricas García

Junio de 2010

Resumen

Con el creciente interés de las agencias publicitarias por aprovechar las posibilidades del social media para planificar las inversiones de sus clientes, la empresa Cierzo Development S.L. considera fundamental desarrollar una herramienta que sirva de base para una metodología de trabajo en este campo. El objetivo es un sistema que permita identificar las conversaciones y blogs más próximos a una serie de temáticas, no definidas apriorísticamente.

El alcance definido para este proyecto es la blogosfera española. El gran volumen, millones de páginas, que conforman la blogosfera es procesado por un diseño distribuido sobre el framework Hadoop y el servicio de computación en nube Amazon Elastic Compute Cloud.

El sistema desarrollado está compuesto de cuatro módulos que implementan la solución, en Java, a los requisitos propuestos.

En el primero se desarrolla un modulo de rastreo basado en Nutch. No obstante, el módulo incluye con mecanismos implementados para la detección del lenguaje y la expansión de la frontera de la blogosfera española. El contenido HTML de cada post es procesado por el módulo de extracción de entidades. Combinando la potencia de XPath y un enfoque de densidad de texto en HTML, el segundo módulo realiza una extracción centralizada de las entidades que definen un post: título, contenido, fecha de publicación y comentarios. Los posts y sus entidades extraídas generan un índice gestionado por el módulo de indexación. El índice sirve de fuente a la aplicación web, permitiendo al usuario realizar búsquedas que devuelven resultados en formato XML. Estos resultados son la entrada del módulo de agrupamiento que se encarga de reconocer patrones entre posts agrupándolos en conversaciones.

El producto final es un sistema altamente escalable que permite identificar las conversaciones, y los post que las forman, generadas en la blogosfera española.

Índice general

Índice de figuras

Índice de cuadros

Capítulo 1

Introducción

1.1. Contexto del proyecto

El proyecto se ha realizado dentro de la empresa Cierzo Development, compañía startup especializada en marketing online y en la gestión de la reputación corporativa. Creada en el año 2003 como fruto de las políticas de Desarrollo, Innovación y Empleo del Gobierno de Aragón y el Ayuntamiento de Zaragoza; se encuentra actualmente en el CEEI[1] de Aragón. Su presencia en el CEEI es resultado de una asignación superior del 30 % de los recursos de la compañía a labores de I+D+i relacionadas con la recuperación de información.

Cierzo Development cuenta con alta tecnología para el análisis del posicionamiento en buscadores, también denominado Search Engine Optimization o SEO. Su herramienta de análisis de posicionamiento SAPIC es el fruto de dos años de investigación apoyados por el Ministerio de Industria a través del programa PROFIT[2] en los años 2006 y 2007, y del programa de personal investigador Torres Quevedo del Ministerio de Educación y Ciencia.

Desde 2008, esta tecnología ha sido evolucionada para incluir medios digitales, blogosfera y redes sociales. Dicha evolución se conoce como la plataforma SMMART: Social Media Marketing Analysis Reporting Tool [1]. Con SMMART, el usuario accede a un sistema de monitorización de la información de los principales canales de Internet, permitiendo realizar un análisis de los resultados de su planificación publicitaria en la red.

[1] Centro Europeo de Empresas e Innovación

[2] PROgrama de Fomento de la Investigación Tecnológica

El éxito de la herramienta ha sido reconocido desde diversos sectores:

- Un porcentaje superior al 10 % de las empresas que cotizan en el IBEX35 son o han sido clientes de SMMART.
- SMMART es una de las dos herramientas de monitorización recomendadas en el octavo volumen [2] de los libros blancos de la IAB[3].
- Cierzo Development lidera proyectos del sector como 3.0 Social Media Ecosystem ó el sistema de gestión integral de marketing online para empresas de sectores tradicionales GAMO, siendo SMMART el motor principal de los mismos.

1.2. Objetivos

El objetivo del proyecto recogido en esta memoria es el desarrollo y puesta en producción de una herramienta de planificación social media en la blogosfera española.

Actualmente Cierzo Development cuenta con la plataforma SMMART para analizar la reputación online de sus clientes y el éxito de sus estrategias de marketing online. Sin embargo, la importancia que ha cobrado el social media como canal de comunicación suscita la creación de un sistema de recuperación de las conversaciones que surgen en Internet.

El cliente de la aplicación es capaz de detectar las conversaciones que surgen en la blogosfera española sobre las temáticas que demande, por ejemplo, sus productos comerciales. De esta manera, el usuario monitoriza la blogosfera española identificando los conversadores y los contenidos de las conversaciones que afectan a sus intereses, recibiendo de esta manera información privilegiada para planificar sus estrategias de marketing.

Los objetivos a realizar en este Proyecto Fin de Carrera, en adelante PFC, son:

1. Adquirir el conocimiento necesario en las tecnologías Amazon EC2, Hadoop, Nutch, Lucene y Solr.
2. Implementar un sistema de rastreo, extracción de entidades de posts e indexación de las mismas para la blogosfera española

[3]Interactive Advertising Bureau: http://www.iabspain.net/

3. Desarrollar un sistema de agrupamiento de posts basándose en su contenido.

La motivación de la empresa para impulsar este sistema no es sólo el desarrollo del producto para su explotación, sino también el diseño de una arquitectura distribuida y altamente escalable que sirva de base para la nueva versión de SMMART. De igual manera, el índice que se genera con los contenidos de la blogosfera española servirá de fuente para futuros productos a desarrollar por Cierzo Development.

1.3. Motivación personal

Existen diversas alternativas a la hora de realizar el PFC. La elección de este proyecto viene motivada por los siguientes aspectos:

I+D+i La investigación, el desarrollo y la innovación son valores fundamentales para cualquier sociedad. La posibilidad de haber desarrollado un sistema innovador en una empresa referente de un sector emergente es un estímulo constante durante el desarrollo del proyecto.

Formación El PFC debe entenderse como una aplicación real de los conceptos adquiridos durante la carrera, quedando demostradas las aptitudes del futuro ingeniero. No obstante, la continuidad del aprendizaje durante este periodo de tiempo sigue siendo un objetivo fundamental. La propuesta realizada por Cierzo Development supone una profundización en Ingeniería del Software o Sistemas de Información, y a la vez, formación en áreas con un menor grado de conocimiento inicial como Sistemas Distribuidos.

Contexto No por último menos importante, la realización del PFC en un entorno real como la empresa, complementa muy satisfactoriamente la experiencia previa como estudiante universitario. Algunos ejemplos de este enriquecimiento son la mayor exigencia en plazos, la metodología de trabajo (sistemas de gestión de proyectos y control de versiones), o el desarrollo de habilidades sociales en nuevas situaciones como el trato con el cliente.

1.4. Estructura del documento

El presente documento se compone de 5 capítulos dando comienzo el de Introducción, donde se encuentra ahora. Los demás capítulos que estructuran

la memoria son los siguientes:

- El Capítulo 2 describe el grupo de trabajo, las tareas desarrolladas por el autor de este PFC y la planificación de las mismas.
- En el Capítulo 3 se especifica el alcance del proyecto, la blogosfera española, completado con un análisis de volumen.
- El Capítulo 4 define la arquitectura desarrollada para el sistema de planificación social media desde un punto de vista tecnológico y funcional.
- Finalmente, en el Capítulo 5 se presentan los resultados obtenidos, las líneas de trabajo para el futuro y una valoración personal.

A lo largo de la memoria se referencian los apéndices que figuran al final de la misma. La memoria ha sido redactada para ser comprendida sin la información que figura en dichos apéndices. Sin embargo, se recomienda su lectura, siguiendo las referencias del texto, para profundizar todos los aspectos tratados. Los apéndices incluídos son:

- El Apéndice A extiende las características de las tecnologías base de este proyecto y lista las herramientas empleadas.
- El manual de usuario requerido por la empresa para la puesta en producción de la infraestructura en Amazon EC2 conforma el Apéndice B.
- El Apéndice C recoge el prototipado de las pantallas principales de la aplicación web junto a algunas capturas, casos de uso y el diagrama entidad/relación de la base de datos que emplea.

Capítulo 2

Planificación del proyecto

2.1. Grupo de trabajo

Dado que el proyecto se realiza dentro de la empresa Cierzo Development, son varias las personas que han formado el equipo de desarrollo. Íñigo García Morte ha desempeñado el papel de dirección de este proyecto, y Hector Horno y David de Larrea han desarrollado la aplicación web que integra el índice de la blogosfera española con el módulo de agrupamiento por conversación. Pablo Aragón Asenjo, autor de esta memoria, participa en todos los procesos de desarrollo llevando las siguientes tareas:

- Investigación y formación en:
  - Amazon EC2, servicio de computación en nube.
  - Hadoop, framework de computación distribuida.
  - Nutch, motor de rastreo.
  - Apache Lucene y Apache Solr, tecnologías de indexación.
- Estimación del volumen de la blogosfera española.
- Diseño de la arquitectura del sistema.
- Implementación de los siguientes módulos:
  - Rastreo.
  - Extracción.
  - Indexación.
  - Agrupamiento.
- Diseño y realización de las pruebas.
- Puesta en producción.
- Documentación del proyecto.

2.2. Evolución del proyecto

La figura 2.1 muestra el flujo de las tareas de desarrollo a lo largo de las etapas del proyecto.

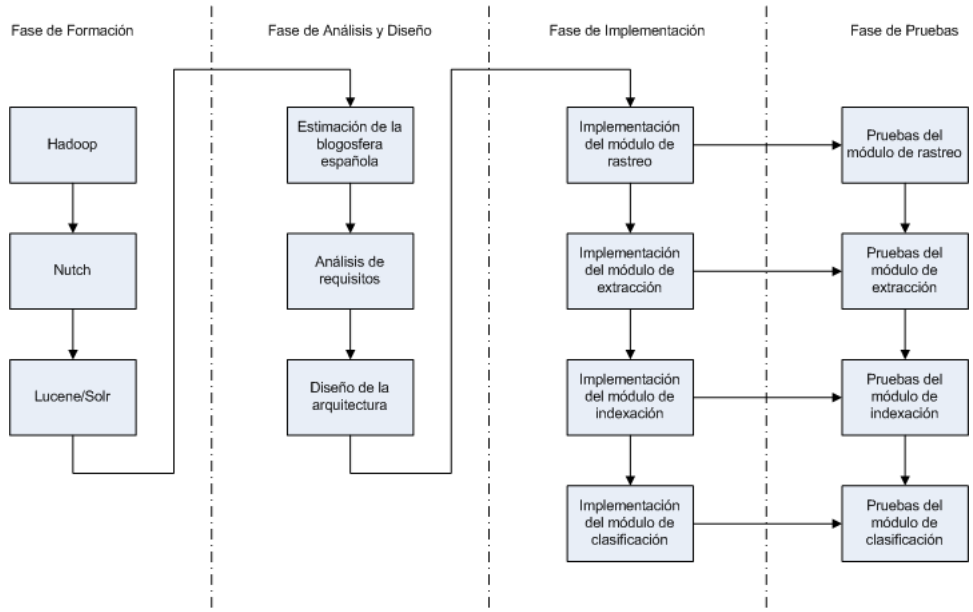


Figura 2.1: Fases del desarrollo del proyecto

En el diagrama de Gantt mostrado en la Figura 2.2 se puede observar la planificación inicial prevista para el proyecto y las desviaciones producidas tras su finalización. Estas desviaciones se han producido por...

Figura 2.2: Diagrama de Gantt del proyecto

Capítulo 3

La blogosfera española

El término blogosfera, acuñado por Brad L. Graham en 1999 y popularizado por William Quick desde 2001, representa la totalidad de los blogs que existen en Internet. Un blog, también denominado bitácora, es una página web donde un usuario o una comunidad publican una serie de contenidos ordenados cronológicamente, similar a un diario.

Uno de los cambios más significativos en Internet desde finales del siglo XX e inicios del siglo XXI, ha sido la transformación del concepto de páginas webs poco actualizadas a sitios webs colaborativos. Esta evolución bautizada por Tim O'Reilly como Web 2.0 [3], fomenta la participación de los usuarios a la hora de añadir o modificar contenidos. Los blogs conforman uno de los ejemplos más destacados dentro de este fenómeno; su carácter colaborativo queda patente en la posibilidad de comentar los artículos publicados.

Las razones del éxito de las bitácoras son muchas y siguen siendo motivo de discusión. No obstante, una de las más aceptadas es la sencillez a la hora de publicar contenidos para un usuario sin experiencia en programación web. De este modo, la accesibilidad a un mayor número de usuarios para divulgar y comentar información en Internet, promueve la aparición de conversaciones entre ellos. Una conversación es entendida como el conjunto de contenidos, ya sean posts o comentarios, publicados por varios usuarios y sobre una misma temática.

En resumen, la blogosfera se convierte no sólo en un gran repositorio de información, sino en una red donde los usuarios comparten y discuten contenidos. Un sistema capaz de identificar y valorar las conversaciones de la blogosfera se convierte en una herramienta de planificación imprescindible para cualquier agencia de publicidad en Internet.

El alcance definido para este proyecto es la blogosfera española, considerándose como el conjunto de blogs cuyo contenido se encuentra escrito en lengua castellana.

3.1. Volumen de la blogosfera española

Antes de diseñar cualquier sistema de recuperación de la información es necesario realizar una estimación del volumen de datos que va a ser gestionado. Diversas compañías especializadas en la blogosfera han realizado las siguientes estimaciones:

- Technorati contabilizó, en 2007, mas de 112 millones en su índice [4].
- Blogpulse reconoce identificar mundialmente más de 126 millones de blogs [5].
- Bitacoras.com publicó su Informe sobre el estado de la blogosfera hispana de 2010 [6] sobre los 417371 blogs de su índice.

Teniendo en cuenta que las dos primeras no ofrecen datos del porcentaje que corresponde a la blogosfera española, y que el informe de Bitacoras.com sólo recoge aquellos blogs suscritos a su red, no se obtienen valores consistentes del volumen para diseñar y configurar el proceso de rastreo. En el próximo capítulo se describe el módulo de rastreo del sistema de planificación, basado en el motor de búsqueda Nutch.

Nutch, al igual que gran parte de los motores de búsqueda, basa su funcionamiento en generar una base de datos a partir de un fichero de entrada que contiene URLs[1]. En cada iteración, extrae un segmento de URLs de la base de datos, analiza su código HTML[2], y obtiene nuevos enlaces salientes que se añaden a la base de datos de URLs.

Partiendo de un fichero semilla de 829305 blogs en español, recuperados anteriormente por Cierzo Development, se configura un proceso de rastreo de URLs que corresponden, mediante expresiones regulares, a la página principal de bitácoras pertenecientes a plataformas conocidas de la blogosfera española[3]. El proceso de expansión también valida que el contenido de las

[1] Uniform Resource Locator: secuencia de caracteres utilizada para identificar recursos en Internet

[2] HyperText Markup Language: principal lenguaje para la elaboración de páginas web

[3] Blogger, Wordpress.com, laCoctelera, Blogia, Bitacoras.com, Vox, Nireblog, Blogcindario, Obolog, Blogalia y Over-Blog

nuevas páginas aparezca en castellano.
Tras 43 iteraciones, el motor analiza todas las entradas del fichero semilla y de los blogs descubiertos en las 42 iteraciones anteriores. Así, el sistema se detiene al no haber URLs sin analizar en la base de datos alcanzando una cifra de 1.735.708 blogs.

De nuevo, y utilizando como fichero semilla las URLs de la base de datos de la prueba anterior, se realiza otro proceso de expansión. Esta expansión, más moderada ya que sólo aparecen nuevos enlaces salientes de aquellos blogs que han sufrido alguna modificacion entre ambas pruebas, finaliza tras 17 iteraciones siendo 1.819.134 el volumen de blogs de la base de datos. La evolución de los dos tests puede analizarse en la figura 3.1 y la figura 3.2 respectivamente.

Concluidas sendas pruebas, se puede afirmar que el volumen de la blogosfera española, perteneciente a las plataformas listadas, es de al menos 1.8 millones de blogs.

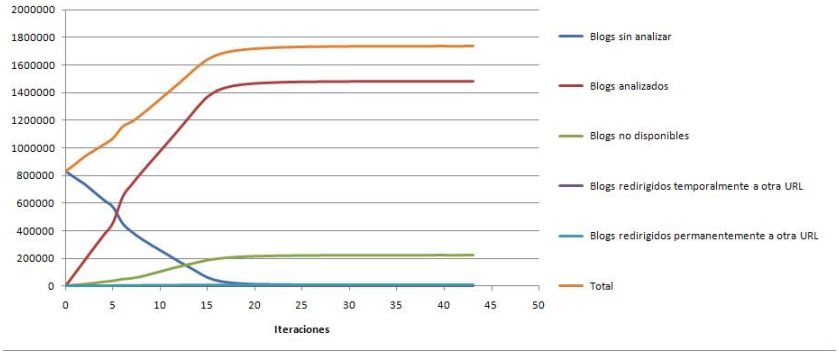


Figura 3.1: Resultados del primer análisis de la blogosfera española

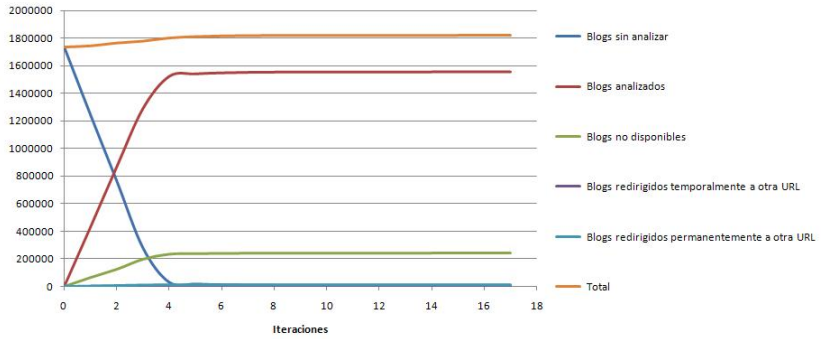


Figura 3.2: Resultados del segundo análisis de la blogosfera española

Los blogs obtenidos se agrupan según la plataforma a la que pertenecen. Como se observa en el cuadro 3.1, Blogger es la plataforma mayoritaria con el 92.77 % del total.

Plataforma	Porcentaje
Blogger	92,77
Wordpress.com	6.20
laCoctelera	0.58
Blogia	0.25
Bitacoras.com	0.06
Vox	0.05
Nireblog	0.04
Blogcindario	0.03
Obolog	0.01
Blogalia	0.01
Over-Blog	0.01

Cuadro 3.1: Porcentajes de las plataformas del segundo análisis

Capítulo 4

Arquitectura del sistema

En este capítulo se describe la arquitectura desarrollada en el sistema del planificación social media abordando su diseño distribuido, los módulos que lo componen y su integración con la aplicación web.

4.1. Diseño distribuido

Los resultados obtenidos en el análisis del volumen de la blogosfera española indican la necesidad de definir una arquitectura capaz de gestionar grandes volúmenes de información al menor coste de operación posible.

Por otro lado, el crecimiento diario del número de bitácoras y las variaciones en la frecuencia de publicación generan un alto grado de incertidumbre. Por tanto, garantizar la evolución de la plataforma en el futuro requiere la posibilidad de ajustar la capacidad de procesamiento sin detener la producción del sistema.

La resolución de ambos requisitos se lleva a cabo mediante un diseño distribuido implementado sobre Hadoop en la infraestructura de computación en nube Amazon Elastic Compute Cloud (EC2) .

4.1.1. Amazon EC2

Amazon EC2 [7] es el servicio de computación en nube perteneciente a Amazon. Dicho servicio permite instanciar servidores y discos duros adaptados a los requisitos de cada uno de los módulos del proyecto. A través de un sistema de pago por consumo, Amazon proporciona las herramientas para administrar una granja remota de servidores configurables para la implementación y

puesta en producción del sistema, obviando la gestión de su espacio físico.

Por tanto, Amazon EC2 se convierte en una plataforma escalable de almacenamiento, transferencia y computación bajo demanda, disponiendo de recursos según las necesidades del proyecto en cada momento. Los detalles de la infraestructura Amazon EC2 figuran en el primer apéndice.

4.1.2. Hadoop

Tres de los módulos que se describen en la próxima sección procesan un gran volumen de información: rastreo, extracción e indexación. La ejecución de cada una de sus tareas se optimiza a través de la utilización del framework Hadoop [8].

Hadoop es una infraestructura distribuida de procesamiento a gran escala. Su diseño permite repartir grandes cargas de trabajo a través de un clúster de máquinas implementando el paradigma de programación MapReduce [9] sobre un sistema distribuido de ficheros basado en Google File System [10].

En MapReduce, cada proceso toma como entrada un conjunto de pares <clave,valor> produciendo una colección de valores de salida. Las dos tareas que realizan esta transformación se conocen como Map y Reduce.

1. Cada tarea Map tiene un par <clave,valor> como entrada y produce una lista intermedia <clave,valor>. La lista se transforma agrupando los valores por clave y generando nuevos pares del tipo <clave,lista(valores)> que sirven de entrada a la tarea Reduce.

2. La segunda tarea, procesa los valores asociados a cada clave produciendo una lista de valores de salida.

El desarrollador implementa las funciones asociadas a las tareas Map y Reduce que transforman claves y valores, ajustándolas al problema a resolver. El Apéndice A incluye una sección que profundiza en el paradigma MapReduce incluyendo un ejemplo.

En un clúster de Hadoop, los datos se distribuyen entre todos los nodos que lo componen. El sistema de ficheros distribuido de Hadoop, conocido como Hadoop Distributed Filesystem o HDFS, divide los archivos de gran tamaño en fragmentos que son gestionados por los diferentes nodos del clúster. Además cada fragmento se replica en varios nodos, de modo que un fallo en una máquina no afecta a la disponibilidad de los datos. Ademas,

la replicación de los datos es transparante al usuario conformando un único espacio de nombres accesible desde cualquier nodo. En el Apéndice A se puede encontrar extensa documentación de la estructura y funcionamiento del HDFS.

Los principales motivos que han suscitado la elección de Hadoop como framework son:

- Alta tolerancia a fallos: La aparición de fallos en computación en gran escala debe ser un hecho asumible. La recuperación automática del sistema que ofrece Hadoop al caer un nodo es fundamental en un proyecto con una alta carga de computación.
- Memoria: Los valores intermedios de Map se suministran a la tarea Reduce través de un iterador. De esta manera el sistema gestiona largas listas de valores minimizando el consumo de memoria.
- Coste: Hadoop está diseñado para procesar grandes volúmenes de datos en un clúster de nodos de tamaño flexible, pudiendo alcanzar la cifra de más de 10000 procesadores. También se ha evaluado que varias instancias de baja gama de Amazon EC2 son inferiores en coste económico a una instancia de alta gama. Por tanto, la distribución de la carga a través de un gran clúster de baja gama basado en Hadoop ofrece los mismos resultados de computacion con mayor rentabilidad.
- Comparativas: La principal alternativa que se ha estudiado para implementar MapReduce es la plataforma de computación grid GridGrain. Esta plataforma, al basarse en tecnología grid, permite integrar diferentes tipos de recursos sin un control centralizado. No obstante, el hecho de que Hadoop ofrezca un sistema propio de ficheros tolerante a fallos y que los datos de GridGrain se estructuran en listas Java, en Hadoop forman iteradores, motiva la elección de Hadoop como framework.

4.2. Módulos del sistema

Los módulos a implementar definen los cuatro procesos principales del desarrollo de este proyecto: rastreo de la blogosfera, extracción de las entidades de un post, indexación de dichas entidades, y agrupación de posts en conversaciones. El diagrama de flujo de datos representado en la figura 4.1 muestra, a alto nivel, la evolución de la información a lo largo de cada módulo del sistema.

El diseño modular de la plataforma permite a su vez la posibilidad de incluir, en el futuro, nuevos módulos que enriquezcan el índice con metainformación. De igual manera, los parámetros que definen los módulos implementados se consultan en ficheros de configuración, permitiendo modificar su comportamiento en fase de producción.

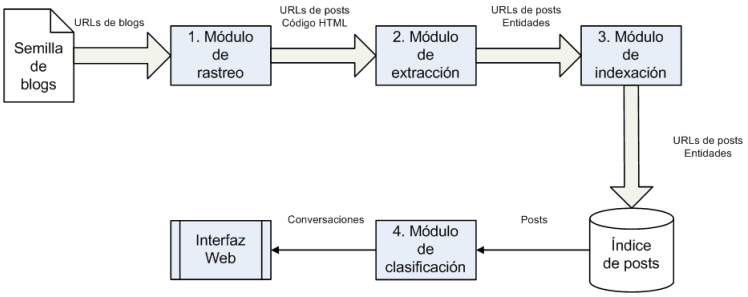


Figura 4.1: Diagrama de flujo del sistema

A continuación se describe cada uno de los módulos implementados.

4.2.1. Módulo de rastreo

El módulo de rastreo, basado en Nutch [11], es el más intensivo en recursos. Nutch es un robot y motor de búsqueda de código abierto gestionado por la Apache Software Foundation. El motor almacena la información en dos estructuras de datos:

- CrawlDb: Este registro contiene las URLs gestionadas por el sistema de rastreo. Cada URL incluye información asociada: fecha de la última consulta y el estado de la URL (consultada, no consultada... etc.).
- Segments: Cada fase de consulta genera un segmento con el contenido HTML de las URLs consultadas en dicha fase, así como los resultados del proceso de análisis.

El esquema de ejecución de Nutch, ilustrado en la figura 4.2, sigue el siguiente flujo de datos:

1. Injector: Inyeccion de las URLs de un fichero a CrawlDb.
   A partir de aquí, el motor de rastreo ejecuta la secuencia, denominada Crawl, de los siguientes 4 procesos durante un número definible de iteraciones.

2. Generate: Generación de un segmento de las URLs a consultar seleccionadas de CrawlDb.

3. Fetcher: Descarga del contenido HTML de las URLs del segmento utilizando un número definible de threads.

4. Parse: Análisis del código HTML descargado de las URLs (extracción del contenido textual, detección de idioma, extracción de enlaces salientes y asignación de puntuación).

5. Update: Actualización de CrawlDb con los resultados del proceso de análisis y las nuevas URLs descubiertas.

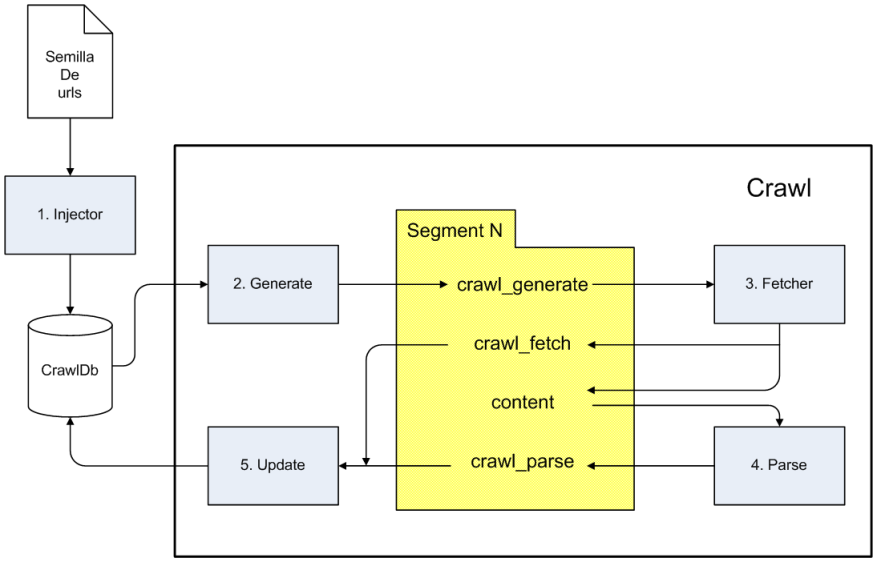


Figura 4.2: Diagrama de flujo del sistema de rastreo

Nutch ofrece diversos ficheros de configuración para optimizar el sistema de rastreo en Internet. Para atender las necesidades propias de la blogosfera se han configurado los siguientes parámetros:

- Filtros positivos y negativos para URLs basados en expresiones regulares.
- Identificación del motor en la cabecera de las tramas enviadas.
- Retardo entre consultas sucesivas a un mismo servidor.
- Volumen, retardo y concurrencia de los threads de Fetcher.

4.2.1.1. Rastreo de la blogosfera española

Nutch está diseñado para poder configurar el intervalo de tiempo que debe transcurrir para que una URL vuelva a formar parte de un proceso de Crawl. De esta manera, en cada fase de Generate, el segmento se conforma con un número definible de URLs que no han sido analizadas o cuya anterior consulta ha superado dicho intervalo. Si el número de URLs asignables supera el valor máximo indicado, se incluyen al segmento aquellas que poseen mayor puntuación. Este enfoque es válido para realizar el rastreo de un volumen de sitios web con una demanda similar de reconsulta. Sin embargo, para el alcance de este proyecto, la blogosfera española, es necesario implementar una serie de mejoras que optimicen el rastreo.

Por un lado el módulo requiere distinguir la naturaleza de la URL, es decir, si pertenece a la página principal de un blog o a un post. Esta diferenciación es obvia, una página principal necesita ser reconsultada para extraer los enlaces de las URLs de los nuevos posts, mientras que un post sólo suele ser modificado con la aparición de nuevos comentarios.

Como ya se comentó, la puntuación de cada URL viene determinada por el proceso de Fetcher. Esta puntuación tiene una correspondencia directa con el número de veces que la página será reconsultada. En la blogosfera, la página de un blog demanda una mayor puntuación por dos factores: relevancia social del blog y frecuencia de publicación. El primero viene motivado por la capacidad de influencia de un grupo pequeño de autores. Bloggers como Enrique Dans o Ignacio Escolar generan contenidos con un alto impacto en la sociedad [12], por tanto, el incremento de la puntuación de sus blogs minimiza el tiempo transcurrido entre la publicación de sus posts y su aparición en el índice del sistema. El segundo factor, la frecuencia de publicación, es un paramétro fundamental para la optimización del rastreo. Diferenciar los blogs que publican diariamente, semanalmente y mensualmente permite ajustar los parámetros de reconsulta ajustando el modelo a la realidad.

Finalmente, los posts que forman parte del índice son aquellos que pertenecen a la blogosfera española. La distinción de blogs en castellano y blogs en lengua extranjera es esencial como filtro de ruido.

Todos estos requisitos son inalcanzables con la implementación disponible de Nutch. Por tanto, siendo Nutch software libre y con el objetivo de cumplir las necesidades de la blogosfera española, se han llevado a cabo las siguientes implementaciones:

- Distinción de URLs según correspondan a la página principal de un blog o a un post.
- Inclusión en CrawlDb de campos de lenguaje y de fecha de la primera consulta de cada URL.
- Sistema de ponderación de blogs basado en la frecuencia de publicación.
- Mecanismo que inhabilita la selección de posts tras un intervalo configurable de tiempo desde la primera consulta.
- Mecanismo de detección de nuevas URLs basado en el idioma detectado y la naturaleza de la URL.
- Herramientas para listar las URLs de CrawlDb filtran por parámetros indicados por el usuario.
- Mecanismo para indicar el número de URLs de CrawlDb agrupadas por dominio.

La figura 4.3 ilustra la implementación del módulo de rastreo atendiendo a sus requerimientos y empleando las implementaciones realizadas. El módulo de rastreo ejecuta dos procesos de Crawl distintos, un primero de blogs reconocidos como relevantes y un segundo de caracter global. Así, se garantiza que las URLs de los autores más influyentes son reconsultadas en cada iteración. Por su parte, el proceso Crawl global analiza todas las URLs almacenadas en CrawlDb.

La aparición del detector de frecuencias permite agruparlas siguiendo la periodicidad de la aparicion de contenidos. La optimización conseguida por este modelo queda reflejada en los siguientes puntos:

1. Se estimula la consulta de aquellos blogs que generan un alto volumen de contenidos.
2. El tiempo de reconsulta se ajusta a la actividad concreta de cada autor.
3. Los blogs inactivos son fácilmente detectables.

Para la detección del lenguaje español se utiliza una implementación basada en la detección de n-gramas [13]. Cada idioma viene definido por una gramática que indica los posibles caracteres que pueden suceder a los anteriores. En la actualidad, fruto de los estudios realizados en el procesamiento estadístico del lenguaje natural, existen ficheros de n-gramas frecuentes de un gran

número de lenguajes. De esta manera, la detección del idioma comparando las estadísticas de acierto de un texto en cada lenguaje permite su identificación con una tasa de error casi nula para textos de más de 256 caracteres [14]. Los idiomas detectables por el módulo son: Español, Inglés, Portugués, Francés, Alemán, Italiano, Danés, Griego, Islandés, Húngaro, Holandés, Noruego, Polaco, Ruso, Sueco y Tailandés.

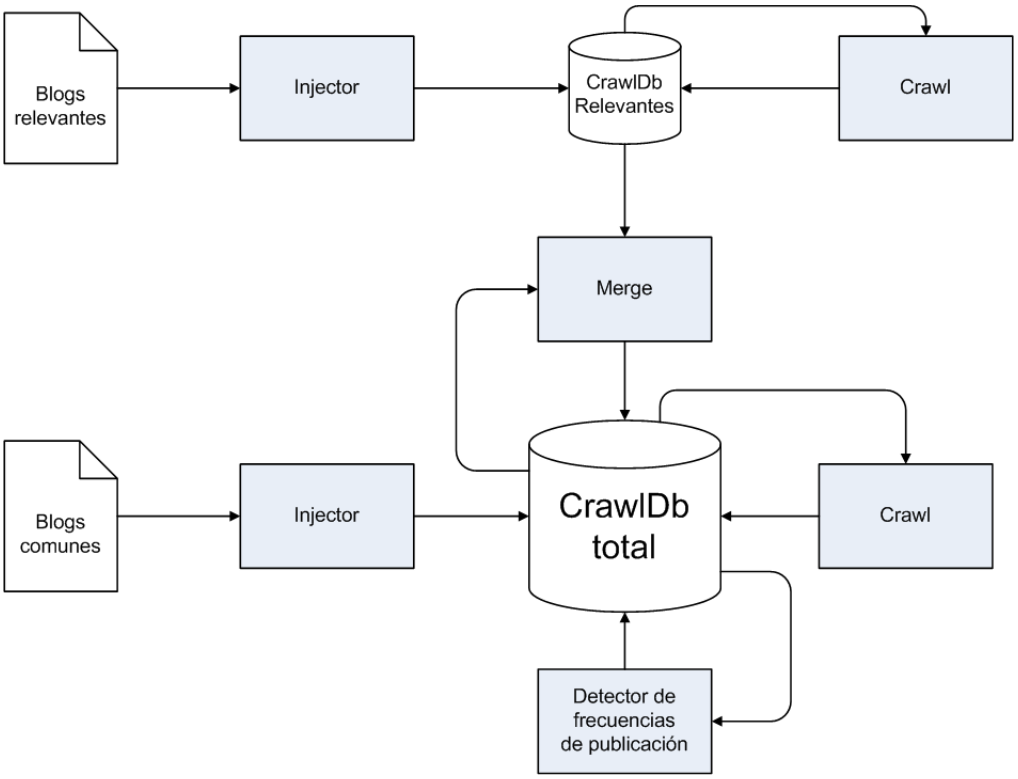


Figura 4.3: Diagrama de flujo del modulo de rastreo

4.2.2. Módulo de extracción de entidades

Cada iteración del sistema de rastreo realiza la extracción del código HTML de las URLs que componen el segmento. Dicho código queda almacenado en el subdirectorio Content dentro del directorio correspondiente al segmento.

La entrada de un blog está estructurada por una serie de entidades que lo definen: título, contenido, comentarios y fecha de publicación. El sistema de extracción de entidades realiza un análisis sintáctico sobre el código HTML de cada uno de los posts obteniendo el contenido textual diferenciado de

dichas entidades. El diseño de este análisis combina la potencia de XPath[1] [15] y un enfoque basado en la densidad del contenido textual del HTML

XPath es un lenguaje para evaluar un documento jerárquico como XML[2] o HTML a través de una serie de expresiones similares a una expresión regular. El código HTML de un blog sigue la estructura marcada por la plantilla del sistema de gestión de contenidos que lo administra. Estas plantillas incluyen metainformación en los identificadores de las etiquetas del HTML facilitando la identificación del texto que corresponde a cada entidad. Por tanto, incluyendo en un fichero de configuración expresiones XPATH que reconocen dichos identifcadores, el módulo de extracción recorre jerárquicamente el HTML y extrae los segmentos de texto deseados.

Aquellas URLs, cuyo HTML no proporcionan metainformación en los identificadores de sus etiquetas, son analizadas siguiendo un enfoque de densidad. El concepto se basa en calcular la densidad del texto y del código HTML de cada línea para validarla. Las líneas cuyo porcentaje de contenido textual, frente al del HTML, supera un valor mínimo fijado corresponden al contenido del post.

De esta manera, el sistema de extracción realiza un análisis centrado en las entidades que definen un post descartando los contenidos ajenos, como la publicidad o el blogroll. Este enfoque mixto no sólo incrementa la calidad del texto extraído, también permite enriquecer el análisis modificando únicamente las expresiones XPath del fichero de configuración .

4.2.3. Módulo de indexación

Las entidades extraídas en el sistema anterior son almacenadas en un índice para poder realizar las consultas a texto completo que generan las entradas del sistema de agrupamiento. La tecnología escogida en el sistema de indexación es Apache Solr [16]. Solr es un motor de búsqueda desarrollado en Java que utiliza la tecnología de indexación Lucene [17] ofreciendo entre muchas funcionalidades:

- Búsqueda a text completo (full-text search).
- Algoritmos de ordenación de resultados.
- Lematización (stemming) de tokens.

[1] XML Path Language

[2] eXtensible Markup Language: metalenguaje extensible de etiquetas

- Filtrado de palabras frecuentes (stop words).
- Interfaz web de administración.

Ademas, el diseño de Solr permite alcanzar un alto grado de escalabilidad proporcionando herramientas para la implementacion de búsquedas distribuidas y replicación de índices. En la actualidad existen otras tecnologías alternativas como MySQL Full Text y Sphinx. El cuadro 4.1 muestra una comparación entre ellas contrastando sus características[3].

Prestaciones	MySQL FullText	Sphinx	Apache Solr
Identificadores	Opcional	Obligatorio	Opcional
Stop words	Sí	Sí	Sí
Stemming	No	Sí	Sí
Relevance Scoring	Basada en TF-IDF	Configurable	Configurable
Faceted search	No	No	Sí
Licencia	GPL	GPL	Apache

Cuadro 4.1: Comparación de prestaciones de tecnologías de indexación

El sistema de indexación genera un documento en el índice Solr de cada uno de los posts procesados en el módulo de extracción de entidades. La figura 4.4 incluye la estructura de datos establecida para el servidor Apache Solr del módulo. Se han ocultado algunos detalles de definición de estructuras y optimización por solicitud expresa de la empresa, que desea mantener cierta confidencialidad en este aspecto.

[3]Una de las diferencias principales de una licencia GPL, frente a una Apache, es la obligación de adquirir una licencia comercial para integrar el software en productos comerciales.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>

<schema name="?" version="1.1">

    <types>
    ...
    </types>

    <fields>

        <field name="title" type="string" indexed="?" stored="?"/>
        <field name="extract" type="text" indexed="?" stored="?"/>
        <field name="content" type="string" indexed="?" stored="?"/>
        <field name="url" type="string" indexed="?" stored="?"/>
        <field name="host" type="string" indexed="?" stored="?"/>
        <field name="published" type="date" indexed="?" stored="?"/>
        <field name="collected" type="date" indexed="?" stored="?"/>
        <field name="updated" type="date" indexed="?" stored="?"/>
        <field name="language" type="string" indexed="?" stored="?"/>
        <field name="type" type="string" indexed="?" stored="?"/>
        <field name="full_text" type="text" indexed="?" stored="?"/>
        <field name="outlinks" type="string" indexed="?" stored="?" multiValued="true"/>
        <field name="comments" type="text" indexed="?" stored="?" multiValued="true"/>
        <field name="categories" type="string" indexed="?" stored="?" multiValued="true"/>

    </fields>

    <uniqueKey required="true">url</uniqueKey>
    <defaultSearchField>full_text</defaultSearchField>

</schema>
```

Figura 4.4: Esquema del índice Solr (schema.xml)

4.2.4. Módulo de agrupamiento

Actualmente existen en Internet directorios de blogs donde el usuario puede registrar su bitácora indicando las categorías que considera más cercanas. De igual manera, la bibliografía de esta memoria incluye estudios de clusterización [18], algunos [19] utilizando las categorías de estos directorios como entrada, para mapear la blogosfera. El módulo de agrupamiento implementado para el sistema de planificación social media está diseñado para agrupar un conjunto de entrada de posts en varios subconjuntos utilizando como criterio la afinidad del contenido y del título.

Para ello, el módulo realiza un proceso de minería de texto. El proceso se encarga, en primera instancia, de realizar un subproceso de etiquetado de los posts analizando las secuencias de palabras más relevantes. El algoritmo de agrupamiento contrasta el resultado del etiquetado de un post con el resto del conjunto estableciendo un grado de afinidad entre posts. Aquellos posts

con una afinidad superior a la cota mínima definida, conforman una conversación identificada con las etiquetas comunes. De esta manera, el módulo de agrupamiento no sólo realiza un proceso de clusterización basado en texto, también asigna un identificador a cada conversación de carácter semántico.

4.3. Integración de los módulos y la WUI

Los módulos de rastreo, extracción e indexación conforman una arquitectura en pipeline[4] volcando iterativamente las entidades de los posts rastreados en el índice Solr. Esta arquitectura se ejecuta en el clúster implementado sobre Hadoop ubicado en la infraestructura Amazon EC2.

El módulo de agrupamiento se despliega como un servicio en la máquina donde está localizada la interfaz web. El sistema integra la aplicación web con los módulos implementados sirviendo de puerta de entrada a las funcionalidades del sistema. A través de esta interfaz, el usuario configura sus funciones en el planificador social media. La función principal de un usuario es la definición de una serie de informes asociados a unas palabras de búsqueda, alojándose la información en un servidor de base de datos MySQL.

Una tarea programada realiza periódicamente los siguientes procesos:

1. Búsqueda de las palabras cada informe sobre el API del índice Solr.
2. Clusterizacion los posts resultantes a través del módulo de agrupamiento.
3. Almacenamiento de los posts y sus conversaciones en MySQL.

Por consiguiente, el usuario accede a una interfaz que monitoriza periódicamente las conversaciones, y sus posts, estableciendo únicamente las palabras de búsqueda de la temática que demande. La figura 4.5 describe el sistema integrado detallando las tecnologías empleadas en sus módulos. El apéndice final de esta memoria incluye los prototipados y pantallas de la interfaz web, los casos de uso más relevantes y el diagrama de entidad/relación de la base de datos de MySQL.

[4] http://es.wikipedia.org/wiki/Arquitectura_en_pipeline_%28inform%C3%A1tica%29

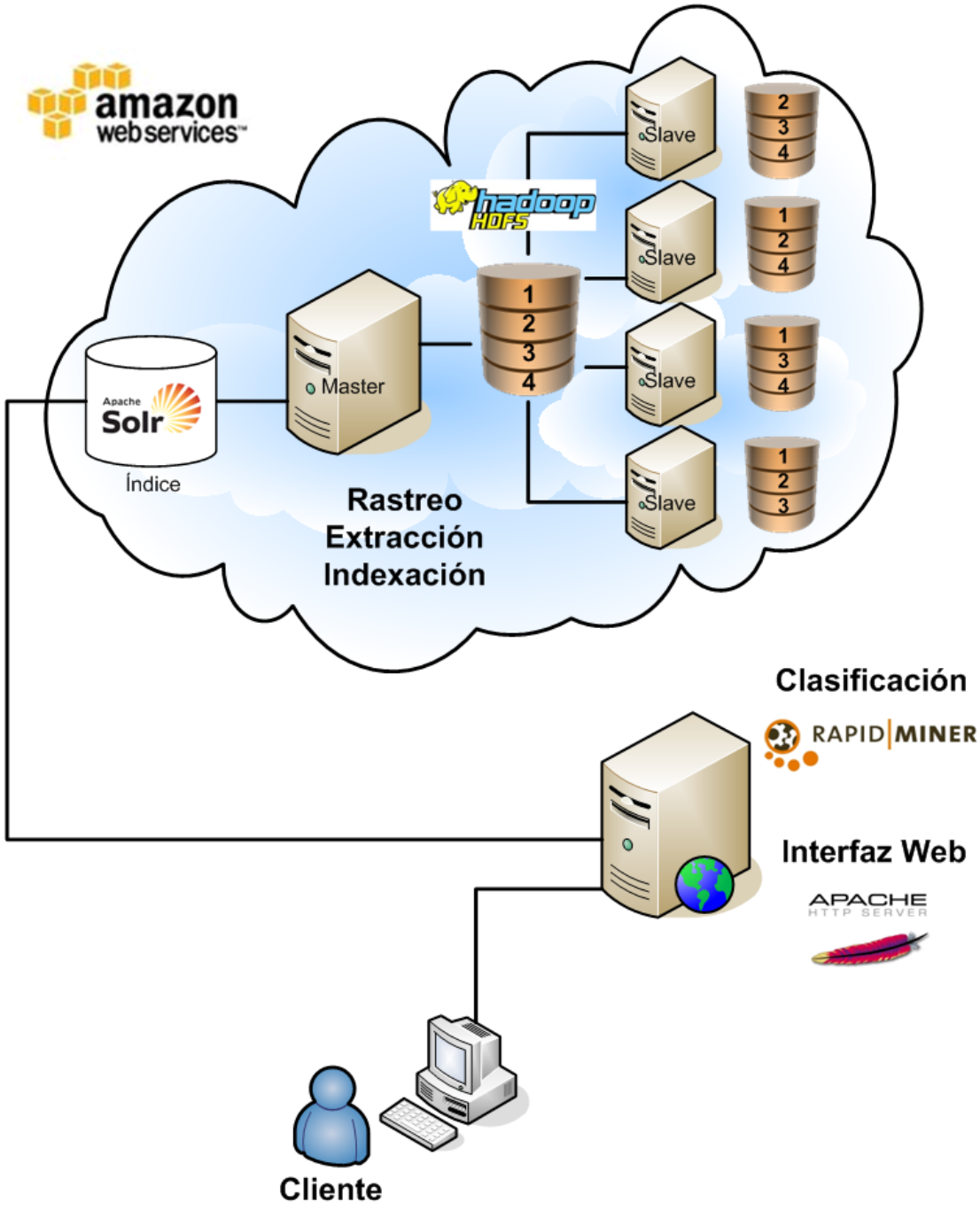


Figura 4.5: Arquitectura tecnológica del sistema

Capítulo 5

Conclusiones

5.1. Resultados obtenidos

El primer resultado satisfactorio tras el desarrollo del sistema es la aceptación y entrega del producto al cliente cumpliendo con los plazos y requisitos acordados. En esta sección se presentan, a través de capturas de pantalla, ejemplos de resultados sobre el índice y de un informe configurado con una búsqueda concreta. También se incluyen estadísticas que justifican el modelo de rastreo diseñado ajustándose el comportamiento esperado.

5.1.1. Resultados funcionales

El índice Solr que almacena los posts procesados por los módulos de rastreo, extracción e indexación alcanzan, en el momento de redacción de esta memoria, una cifra superior a X posts correspondientes a mas de X blogs. La figura 5.1 corresponde a la captura de pantalla de los resultados de la búsqueda X en el índice Solr.

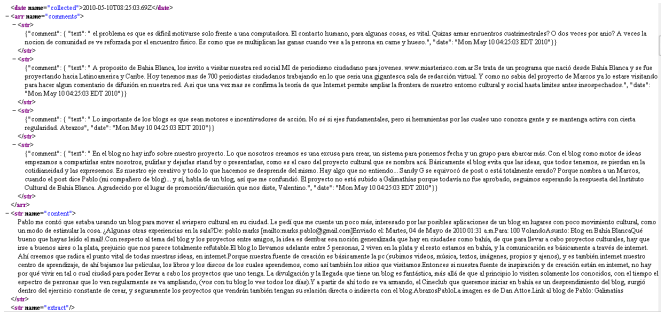


Figura 5.1: Captura del planificador social media

La integración del índice con el módulo de clasificación y la interfaz web conforma el sistema de planificación social media. A través de la aplicación web el usuario puede configurar informes asociados a las búsquedas que los definen. La figura 5.2 muestra un informe para monitorizar el impacto en la blogosfera del vertido de crudo en el golfo de México desde una torre de British Petroleum.

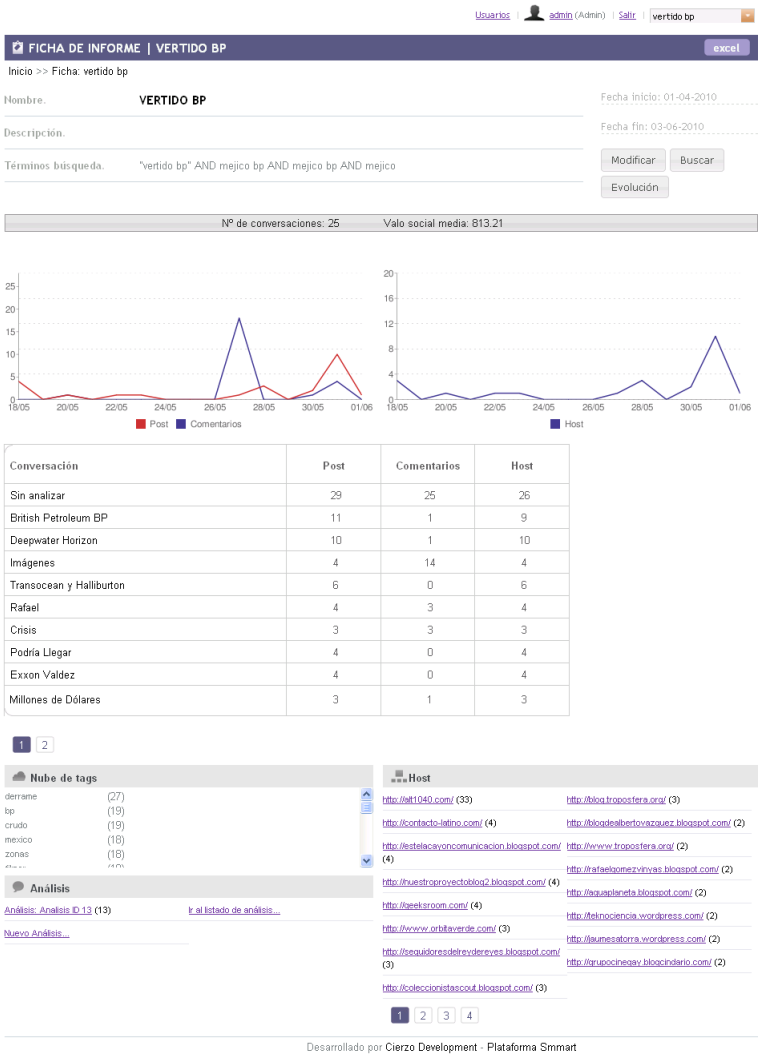


Figura 5.2: Captura de un informe del vertido de BP

5.1.2. Resultados estadísticos

La conclusión de este PFC no sólo implica la consecución de un sistema que cumple los requisitos funcionales. Las herramientas de análisis disponibles para CrawlDb y para el índice Apache Solr permiten valorar las optimizaciones implementadas para ajustar el sistema a la blogosfera española. Al tratarse de información privilegiada de la empresa Cierzo Development, los valores de los ejes de la gráficas exhibidas son simbólicos o no aparecen.

La figura 5.3 representa, en un diagrama de barras, blogs de CrawlDb ordenados por el volumen de posts publicados. La curva exponencial producida por la dispersion de las cimas de la barras explica la situación esperada de la blogosfera. Existe un pequeño grupo de blogs que generan un alto volumen de contenidos, un conjunto mayor con un volumen más reducido, y por último una gran mayoría de blogs con un número de publicaciones comparativamente ínfimo.

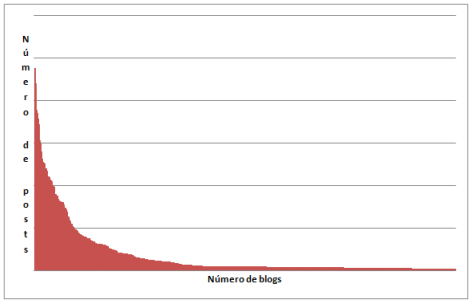


Figura 5.3: Total de blogs ordenados por el número de posts

Identificando la fecha de publicación de los contenidos, el detector de frecuencias establece una puntuación a cada blog. La figura 5.4 presenta el total de blogs ordenados por puntuación en otro diagrama de barras. La curva que describe se asemeja a la de la figura anterior, por tanto, el módulo de rastreo analiza en cada iteración las URLs con mayor posibilidad de generar contenidos y nuevos enlaces. Además, el hecho de que el modelo se ajusta a la frecuencia en cada momento, utilizando el histórico de URLs, implica una mayor sensibilidad del detector de frecuencias optimizando rendimiento del sistema.

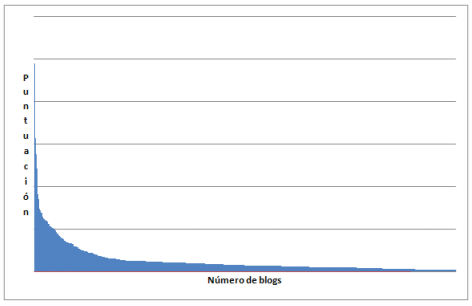


Figura 5.4: Total de blogs ordenador por puntuación

La figura 5.5 exhibe la evolución del número de URLs de cada idioma desde la puesta en producción del módulo de rastreo. La semilla utilizada es el resultado de la combinación del análisis de volumen de la blogosfera española y otras semillas producidas por Cierzo Development. El diagrama muestra un periodo de rastreo de la semilla de 6 etapas. Una vez concluido, el módulo de rastreo identifica los blogs pertenecientes a otros idiomas filtrándolos como ruido, principalmente inglés y portugués. Por tanto, a partir de la 6ª etapa, la curva de URLs en español incrementa significativamente su pendiente, mientras que el resto de idiomas aumenta de manera despreciable.

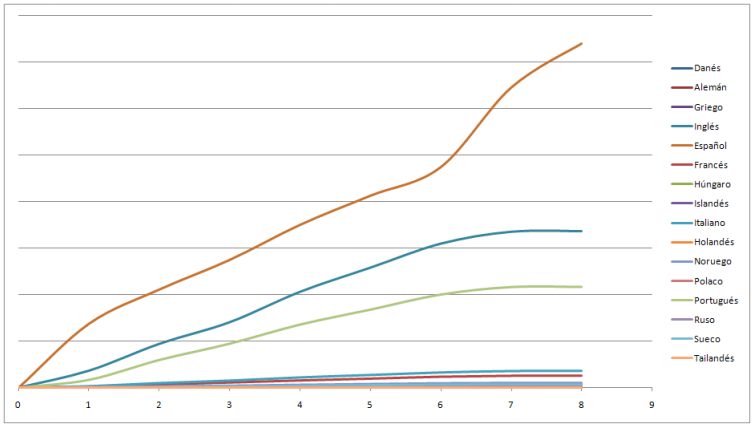


Figura 5.5: Evolución del número de URLs por idioma

5.2. Reconocimientos

El proyecto recogido en esta memoria fue seleccionado para el Meetup del congreso Apache Lucene EuroCon 2010[1] celebrado en Praga. Apache Lucene

[1] http://lucene-eurocon.org/meetup.html

EuroCon 2010 es la primera conferencia realizada en Europa sobre las tecnologías Lucene y Solr. La organización del congreso fue llevada a cabo por Lucid Imagination, primera entidad comercial dedicada exclusivamente a las tecnologías previamente citadas y que cuenta en su plantilla con los principales desarrolladores.

La presentación de este proyecto, realizada por el autor de esta memoria el pasado 20 de mayo, respalda su grado de interés al compartir espacio con ponentes de la relevancia de:

- Yonik Seeley: Creador del proyecto Apache Solr .
- Grant Ingersoll: Chair del Apache Project Management Committee.
- Erik Hatcher: Committer de los proyectos Apache Lucene/Solr y autor de la obra de referencia *Lucene in Action [17].*

5.3. Trabajo futuro

La línea tecnológica de este proyecto, implementando sus procesos en Hadoop sobre Amazon EC2, pretende ser continuada en los siguientes productos desarrollados por la empresa Cierzo Development. La arquitectura definida en este PFC permitirá alcanzar un alto grado de escalabilidad en todos los productos que procesen volúmenes de información tan elevados como la blogosfera española.

Desde un punto de vista funcional existen dos líneas de trabajo partiendo de este proyecto. Por un lado replicar este sistema para otro tipo de fuentes que también conforman el social media, por ejemplo:

- Medios digitales
- Facebook
- MySpace
- Youtube

El índice Solr de la blogosfera española y los índices generados desde las fuentes listadas conformarán el repositorio de información de los nuevos productos de Cierzo Development. La incorporación del autor de esta memoria a la plantilla de la empresa supone la implicación en el desarrollo de los siguientes productos que consumen información de dichos índices:

- Discovering Learning Suite: Mecanismo para entrenar a SMMART para detectar contenidos relevantes mediante ejemplos, en lugar del anterior sistema basado en palabras clave.

- Content Analysis and Semantic Workbench: Las empresas que se enfrentan a grandes volúmenes de información en Internet demandan una aproximación que mezcle análisis cuantitativo (semántico-automatizado) y cualitativo (humano) de los datos. SMMART integrará este módulo de análisis estadístico y semántico de contenido para obtener una visión de conjunto que permita realizar planteamientos estratégicos.

- Community Center: Generalmente la reputación online está en manos de una agencia especializada o un Community Manager dentro de la empresa. Sus principales tareas son la realización de análisis cualitativos y estratégicos de los datos, contacto con los conversadores, medición del impacto de eventos, acciones de marketing y reporting de las tareas realizadas. Todo esto lo puede hacer a través de nuestro Community Center, un completo CRM[2].

- Campaign designer: A la hora de realizar campañas de comunicación en Internet, muchas entidades demandan informacion de quiénes son los principales influenciadores. Con el módulo Campaign Designer el cliente podrá identificar los soportes y usuarios más relevantes para su ámbito de actuación.

- Realtime reporting system: Alertas, explotación gráfica de la información, exportación a Excel o Pdf y vistas personalizadas. Este sistema de reporting en tiempo real permitirá ofrecer servicios de alto valor añadido en tiempo real.

- Multimedia labs: Proximamente SMMART integrará funciones multimedia para enriquecer las técnicas de marketing en Internet. La herramienta no se limitará a reportar apariciones en medios multimedia, sino que también señalizará oportunidades de negocio al cliente.

5.4. Valoración personal

La implementación de un PFC con un un grado significativo de investigación, desarrollo e innovación es, sin duda, la mejor situación para iniciar la carrera profesional como ingeniero. Por otro lado, la formacion en las tecnologías

[2]Customer Relationship Management: modelo de gestión orientado al cliente

Hadoop, Nutch, Lucene/Solr y los conceptos de programación distribuida supone un excelente complemento a la etapa de estudiante de Ingeniería en Informática. La incorporación, previamente comentada, a la plantilla de Cierzo Development confirma la satisfacción de escoger esta empresa para la realización del PFC.

Finalmente, la selección de este proyecto para el congreso Apache Lucene EuroCon 2010 se traduce en el reconocimiento del interés del sistema implementado, la invitación a compartir experiencia con el resto de congresistas, y la introducción a una comunidad internacional de profesionales y desarrolladores afines.

Por consiguiente, la valoración personal del proyecto realizado y recogido en esta memoria no sólo satisface las motivaciones iniciales, también supone un enriquecimiento profesional y personal superior al esperado.

Apéndice A

Tecnologías y herramientas empleadas

La arquitectura diseñada para este PFC se basa en una serie de tecnologías que no se habían utilizado, hasta la fecha, en los procesos de Cierzo Development. El propósito de este apéndice es el de realizar una descripción de cada una de ellas, así como de las herramientas utilizadas en el desarrollo del proyecto.

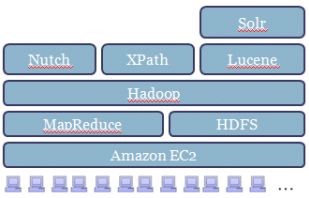


Figura A.1: Tecnologías del PFC

A.1. Tecnologías

Esta sección pretende ser una breve introducción a las tecnologías empleadas para enriquecer los conocimientos del lector.

A.1.1. Amazon EC2

Amazon EC2 es un servicio web que permite lanzar y gestionar instancias de máquinas en centros de datos de Amazon usando APIs y herramientas propias. Las instancias están disponibles en diferentes tamaños y configuraciones. Esto permite proporcionar distintos tipos de intancias que pueden utilizarse para satisfacer necesidades específicas. Por ejemplo, es posible utilizar una instancia m1.small como un servidor web, una instancia m1.xlarge como un servidor de base de datos o una instancia c1.xlarge para el procesador de aplicaciones intensivas. El cuadro A.1 muestra las diferentes tipos

de instancias que se pueden gestionar con Amazon EC2.

CPU	Memoria	Disco duro	Plataforma	I/O	id
1 procesador	1.7 GB	160 GB	32 bits	Moderada	m1.small
2 procesadores	7.5 GB	850 GB	64 bits	Alta	m1.large
4 procesadores	15 GB	1690 GB	64 bits	Alta	m1.xlarge
2 procesadores	1.7 GB	350 GB	32 bits	Moderada	c1.medium
8 procesadores	7 GB	1690 GB	64 bits	Alta	c1.xlarge
2 procesadores	17.1 GB	420 GB	64 bits	Moderada	m2.xlarge
4 procesadores	34.2 GB	850 GB	64 bits	Alta	m2.2xlarge
8 procesadores	68.4 GB	1690 GB	64 bits	Alta	m2.4xlarge

Cuadro A.1: Instancias disponibles en Amazon EC2

Amazon EC2 proporciona una serie de características para la construcción de infraestructuras y aplicaciones escalables:

- Amazon Elastic Block Store ofrece almacenamiento persistente para las instancias de Amazon EC2. Los volúmenes de datos Amazon EBS permiten que el almacenamiento de información persista independientemente de la vida de la instancia a la que se encuentra asociada.

- Amazon EC2 proporciona la capacidad de ubicar instancias en distintas areas geográficas. De esta manera se permite administrar instancias en distintas localizaciones para gestionar fallos de disponibilidad asociados y también para garantizar una menor latencia en las comunicaciones entre instancias de la misma área. Las regiones disponibles actualmente en Amazon EC2 figuran en el cuadro A.2.

- Las direcciones IP elásticas son direcciones IP estáticas diseñadas para la computación en nube. Una dirección IP elástica está asociada con la cuenta de usuario, en vez de con una instancia particular, de tal manera que el cliente gestiona sus propias direcciones IP. Así, se tiene un control de las DNS reasignando las direcciones elásticas que el usuario posee a las instancias que desee.

- Amazon Virtual Private Cloud es un puente entre las infraestructuras tecnológicas de una empresa y la nube de AWS. Amazon VPC permite conectar la infraestructura existente a un conjunto de recursos informáticos AWS a través de una red privada virtual (VPN).

- Amazon CloudWatch es un servicio web que proporciona vigilancia de los recursos de la nube AWS. Provee visibilidad en la utilización de recursos, rendimiento operativo, y métricas generales como el consumo de CPU o el tráfico de red.
- Auto Scaling permite escalar automáticamente la capacidad de Amazon EC2 hacia arriba o abajo según las condiciones que defina. Con Auto Scaling, se garantiza al usuario que el número de instancias de Amazon EC2 que está utilizando escala con los picos de demanda. De esta manera se fija el rendimiento de acuerdo a las necesidades minimizando los costes.
- Elastic Load Balancing distribuye automáticamente el tráfico de múltiples instancias de Amazon EC2. Permite alcanzar una mayor tolerancia a fallos en las aplicaciones, proporcionando la cantidad de equilibrio de carga de capacidad necesaria para responder al tráfico existente. Elastic Load Balancing detecta instancias inestables dentro de un grupo redirigiendo automáticamente el tráfico a las instancias operativas.
- Una imagen de Amazon, conocida como Amazon Machine Image o AMI, contiene la información necesaria para lanzar instancias Amazon EC2. De esta manera, se puede instanciar máquinas que incluyen por defecto el software requerido para operar, por ejemplo, Apache para un servidor web o Hadoop para un nodo de un clúster.

Región	Dominio
Este de Estados Unidos (Norte de Virginia)	ec2.us-east-1.amazonaws.com
Oeste de Estados (Norte de California)	ec2.us-west-1.amazonaws.com
Unión Europea (Irlanda)	ec2.eu-west-1.amazonaws.com
Asia - Pacífico (Singapur)	ec2.ap-southeast-1.amazonaws.com

Cuadro A.2: Regiones geográficas de Amazon EC2

A.1.2. Java

Java es un lenguaje de programación orientado a objetos y multiplataforma diseñado por Sun Microsystems en la década de los 90. Al haber sido la tecnología escogida para un amplio número de asignaturas de Ingeniería en Informática y su uso frecuente en los procesos de Cierzo Development, esta subsección no pretende describirlo sino listar las librerías utilizadas para el desarrollo del PFC:

- Hadoop-Core: Funcionalidades que implementan una plataforma distribuida donde ejecutar aplicaciones bajo el paradigma de MapReduce.
- Jaxen: Librería Java que permite el análisis y la manipulación de códigos HTML a través de la sintaxis de XPath.
- JTidy: Herramientas necesarias para verificar si un código HTML está bien formado y corregir los posibles errores sintácticos.
- Apache SolrJ: Librería con las funcionalidades para manejar la tecnología Solr
- Lucene Core: Conjunto de herramientas para gestionar un índice Lucene
- Librerías propias de Nutch: Colección de librerías para implementar el motor búsqueda Nutch

A.1.3. MapReduce

A.1.3.1. Descripcion

MapReduce es el paradigma de programación diseñado para procesar grandes volúmenes de datos en paralelo, dividiendo el trabajo en un conjunto de tareas independientes. Para ello es necesario dividir la carga de computación a través de un gran número de máquinas.

Este paradigma tiene su origen en los combinadores map y reduce de los lenguajes funcionales como Lisp. En Lisp, un combinador map toma como entrada una función y una secuencia de valores aplicando dicha función a cada uno de los valores. Un combinador reduce combina todos los elementos de la secuencia utilizando una operación binaria.

En MapReduce, el usuario define la función asociada a las tarea map que se aplica a una entrada de pares (clave,valor) produciendo otro conjunto intermedio de pares (clave, valor). MapReduce agrupa todos los valores intermedios por clave sirviendo de entrada a la tarea reduce.

En la tarea reduce, el usuario también define una función que se aplica al conjunto intermedio de pares (clave,valor) generando el conjunto de valores de salida.

Un ejemplo típico para ilustrar MapReduce es el conteo de cada una de las palabras que aparece en una serie de documentos. Su implementación a alto nivel sería la siguiente:

Algorithm A.1 Map de conteo de palabras

```
map(Cadena clave, Cadena valor):
// clave: nombre del documento
// valor: contenido del documento

    para cada palabra w en valor
        EmiteParIntermedio(w, 1);
```

Algorithm A.2 Reduce de conteo de palabras

```
reduce(String clave, Iterator valores):
// clave: palabra
// valores: lista de apariciones

    entero result := 0;
    para cada v en valores
        result := result+1;
    Emite(result);
```

En este ejemplo, la function map emite el valor 1 para cada palabra encontrada. La función reduce recibe una lista intermedia con pares del tipo (palabra,lista<1,1,1...>). La longitud de la lista coincide con el número de apariciones de la palabra en el total de documentos.

A.1.3.2. Ejecución de un proceso MapReduce

Las tareas Map se distribuyen a través de múltiples máquinas particionando automáticamente los datos de entrada en un conjunto de M nodos. Los fragmentos de entrada pueden ser procesados de forma paralela en diferentes máquinas. Por su parte, las tareas Reduce son distribuidas en el clúster agrupando las claves intermedias en R fragmentos.

La figura A.2 muestra el flujo de un proceso MapReduce a través de los siguientes pasos:

1. La librería MapReduce divide los archivos de entrada en M fragmentos. A continuación, se realizan varias copias del programa en un clúster de máquinas.

2. Uno de los nodos del clúster asume el rol de maestro, mientras el resto son clientes con unas tareas asignadas por el maestro. En total existen M tareas map y R tareas reduce para asignar. El maestro gestiona la disponibilidad de cada nodo cliente para ir asignándole tareas.

3. Un nodo cliente al que se le ha asignado una tarea map lee los contenidos de entrada correspondientes. Analiza los pares (clave,valor) y genera nuevos pares según la función Map definida por el usuario. Estos pares intermedios (clave,valor) producidos se almacenan temporalmente en memoria.

4. Periódicamente, los pares en memoria se escriben en disco, indicando al maestro la ubicación de los mismos. El maestro es responsable de transmitir estas direcciones a los nodos que realizan tareas reduce.

5. Cuando un nodo está disponible para realizar una tarea reduce, recibe una señal del maestro con la ubicación de los datos intermedios a procesar agrupándolos por clave.

6. El nodo cliente ejecuta la tarea reduce aplicando la función definida por el usuario y generando los valores finales que son almacenados en el fichero de salida.

Cuando todas las tareas map y reduce se completan, el maestro informa de la finalización del proceso MapReduce, y por tanto, los datos de salida están disponibles.

Para detectar fallos, el maestro realiza un ping periódicamente a cada máquina. Si no se recibe respuesta de un nodo en un determinado periodo de tiempo, el maestro marca el nodo como fallido. Cualquier tarea realizada por un nodo fallido queda anulada y se reasigna a otro.

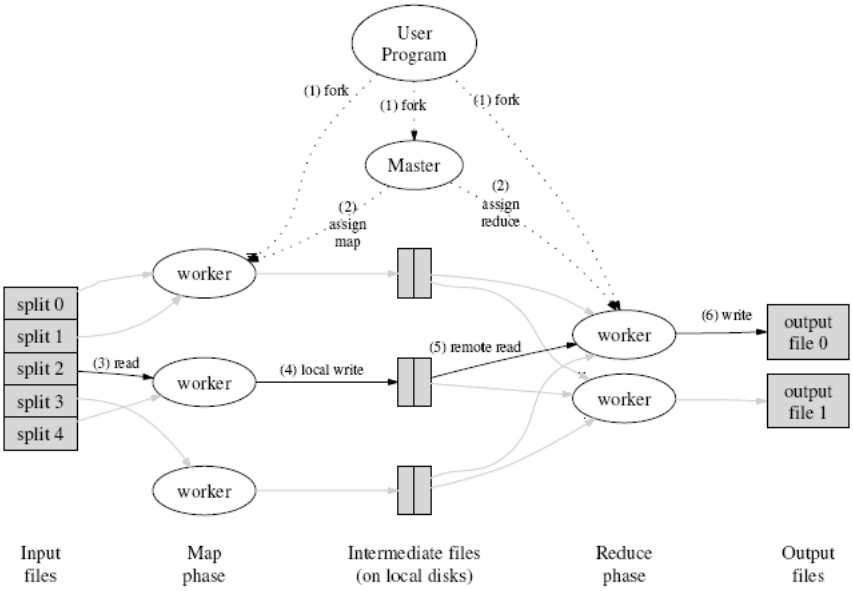


Figura A.2: Esquema de MapReduce

A.1.4. HDFS

Hadoop Distributed File System es el sistema de ficheros distribuido implementado para el framework Hadoop. Tiene muchas similitudes con los actuales sistemas distribuidos de archivos. Sin embargo, las diferencias con otros sistemas son significativas. HDFS es altamente tolerante a fallos y está diseñado para ser implementado en hardware de bajo coste. HDFS proporciona acceso de alto rendimiento y es adecuado para aplicaciones que tienen grandes conjuntos de datos. También, es flexible con algunos requisitos de POSIX para permitir el acceso vía streaming a los archivos del sistema. Fue construido originalmente como infraestructura para el proyecto Apache Nutch.

Los errores de hardware en computación distribuida deben son la norma y no la excepción. Un clúster HDFS puede consistir en cientos o miles de servidores, cada uno almacenando parte de los datos del sistema de ficheros. La existencia de un alto número de nodos y que cada componente tiene una probabilidad no trivial de fracaso implica que algún nodo de HDFS puede dejar de funcionar en cualquier momento. Por lo tanto, la detección de fallos y la recuperación automática de ellos es un objetivo básico de la arquitectura HDFS.

A.1.4.1. NameNode y DataNodes

HDFS posee un arquitectura cliente/servidor. Un clúster HDFS consiste en un único NameNode: un proceso que gestiona el espacio de nombres del sistema de ficheros y que regula el acceso a los datos de los nodos cliente. Además, hay un número de DataNodes, generalmente uno por nodo cliente en el clúster, que gestionan de almacenamiento conectándose con su nodo correspondiente. HDFS expone un espacio de nombres que almacena los datos en ficheros. Internamente, un fichero es divido en uno o varios bloques que son almacenados en un conjunto de DataNodes. El NameNode ejecuta las operaciones del sistema de ficheros como la apertura, cerradura o modificación de ficheros y directorios. También determina el direccionamiento de los bloques hacia los DataNodes. A su vez, los DataNodes son responsables de enviar y recibir peticiones desde los clientes del sistema de ficheros.

El NameNode y el Datanode son componentes software diseñados para ser ejecutados en máquinas de bajo coste, generalmente en un sistema operativo GNU/Linux. No obstante, HDFS está implementado en Java, por lo que cualquier máquina que soporte este lenguaje es capaz de ejecutar el Namenode y el DataNode. Una arquitectura típica consta de una máquina dedicada al NameNode y el resto del clúster ejecutando una instancia de DataNode por nodo. Esto no impide la posibilidad de ejecutar múltiples DateNodes en una misma máquina, aunque esta implementación es poco frecuente.

La existencia de un único Namenode en el clúster simplifica considerablemente la arquitectura del sistema. El NameNode es el árbitro y el repositorio de la metainformación del HDFS. El sistema está diseñado de tal manera que los datos fluyen a través del Namenode.
También es posible la ejecución de un Secondary NameNode, que a pesar de su nombre, no actua como un NameNode. Su rol principal es mezclar periódicamente el espacio de nombres con la edición de logs simplificando los históricos. El Secondary Namenode se ejecuta generalmente en una máquina separada ya que requiere un consumo considerable de CPU. Mantiene una copia del espacio de nombres para gestionar la posible caída del nodo que alberga el NameNode. En este caso, se realiza una copia de los archivos del NameNode registrados en el Secondary NameNode recuperándose el sistema.

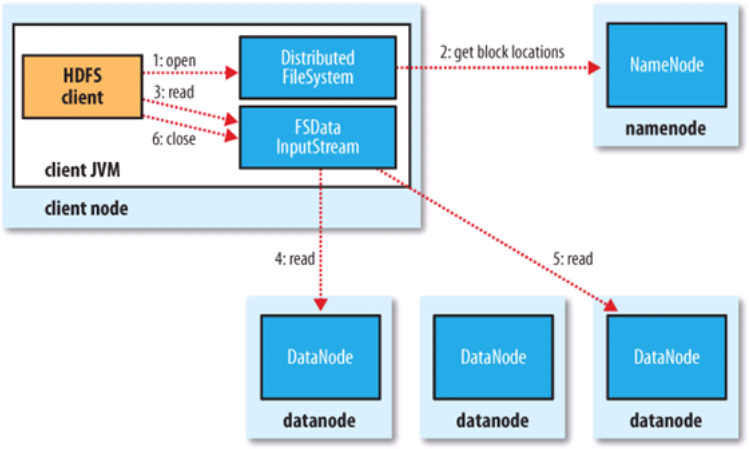


Figura A.3: Esquema de Hadoop Distributed Filesystem

A.1.4.2. El espacio de nombres del HDFS

HDFS soporta una organización jerárquica tradicional de ficheros. Un usuario o una aplicación pueden crear directorios y almacenar archivos dentro. La jerarquía del espacio de nombres del sistema de ficheros es similar a la de la mayoría de sistemas: se pueden crear y eliminar archivos, moverlos de un directorio a otro, o renombrarlos. Sin embargo carece de algunas funcionalidades como los permisos de acceso, los accesos directos o las cuotas de usuario.

A.1.4.3. Replicación de datos

HDFS está diseñado para almacenar, de manera fiable, ficheros de gran tamaño en un clúster compuesto por un alto número de nodos. Cada fichero se almacena como una secuencia de bloques del mismo tamaño. Los bloques de un fichero son replicados para incrementar la tolerancia a fallos. El tamaño de bloque y el factor de replicación son configurables por el usuario.

El NameNode asume las decisiones con respecto a la replicación de bloques y su distribución en el clúster. Recibe periódicamente un ping de cada uno de los DataNodes del clúster informando de la disponibilidad de sus bloques. De esta manera, administra cada uno de los nodos de clúster y la información que corresponde a la distribución de los ficheros.

Por su parte, los DataNodes almacenan la información del HDFS en archivos

dentro su sistema local de ficheros. El DataNode no tiene conocimiento alguno de la distribución de los ficheros HDFS, únicamente almacena cada bloque asignado del HDFS en ficheros locales. Así, utiliza una heurística para determinar el número óptimo de ficheros por directorio.

A.1.5. Nutch

Nutch es el motor de búsqueda gestionado por la Apache Software Foundation, desarrollado bajo la tecnología Lucene, e implementado para ejecutarse en un clúster de nodos de Hadoop. Dependiendo del propósito de los datos, y la forma en que se accede una vez que se ha creado, las estructuras siguen un esquema basado en archivos MapReduce o en ficheros secuenciales. Dado que los datos se procesan siguiendo el paradigma MapReduce, ejecutándose varias tareas map y tarea reduce, esta configuración corresponde a los formatos de salida MapFileOutputFormat y SequenceFileOutputFormat. Las estructuras empleadas son las siguientes:

- CrawlDb: Almacena el estado actual de cada URL como un fichero map <url,CrawlDatum>, donde url es una clase Text y CrawlDatum una clase definida en Nutch. Para proveer de un acceso rápido a los registros, este conjunto de pares se almacena en ficheros map en lugar de ficheros secuenciales. Crawldb se crea inicialmente utilizando la clase Injector que convierte un listado de URLs de un fichero al formato descrito. La información de estado, CrawlDatum, es actualizada durante la ejecución de Nutch.

- LinkDb: Esta base de datos almacena el link que apunta a cada URL conocida en Nutch. Se trata de un fichero map del tipo <url,Inlinks> donde Inlinks es una lista de las URLs que apuntan a la clave.

- Segments: Los segmentos en Nutch corresponden a cada proceso de Fetch y Parse de URLs. Un segmento, que a su vez es un directorio del sistema de ficheros, contiene los siguientes subdirectorios:

    - content: Contiene el contenido HTML de cada página parseada con un formato <url,Content>. Nutch utiliza un fichero map para optimizar el acceso a esta información.
    - crawl_generate: Almacena la lista de URLs a ser consultadas, así como el estado del CrawlDb, con un formato secuencial <url,CrawlDatum>.
    - crawl_fetch: Contiene el estado resultante de la consulta de cada URL almacenándose en un fichero map <url, CrawlDatum>.

- crawl_parse: La lista de enlaces salientes de todas las URLs que han sido consultadas satisfactoriamente. Estos enlaces son los que utiliza Nutch para expandir la frontera de la semilla del sistema de rastreo.
- parse_data: Metadatos recogidos durante la fase de parse, entre otros, la lista de enlaces salientes.
- parse_text: Texto extraído por el análisis sintáctico de cada URL. El formato de esta estructura es un fichero map del tipo <url,ParseText>.

La figura A.4 describe los procesos MapReduce que se ejecutan durante un proceso de rastreo en Nutch.

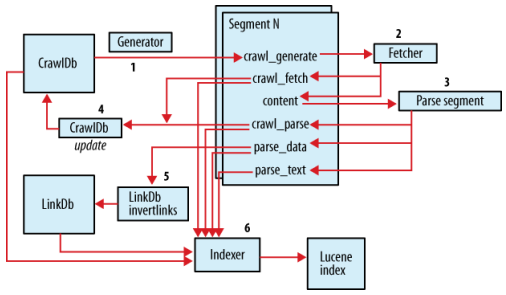


Figura A.4: Diagrama de flujo de Nutch

A.1.6. XPath

XPath es un lenguaje que permite el procesamiento de datos ajustándose al modelo de datos definido en [XQuery/XPath Data Model (XDM)]. Se considera el resultado de un esfuerzo por proporcionar una semántica común y la sintaxis de las funciones que comparten XSLT[1] y XPTR[2]. El objetivo principal de XPath es hacer frente a todas las partes de un documento XML. En apoyo de este objetivo principal, también proporciona servicios básicos para la manipulación de números, cadenas y booleanos. XPath utiliza una sintaxis que facilita su uso dentro de una URI[3] y de los valores de los atributos de un fichero XML. XPath opera sobre la estructura abstracta, lógica de un documento XML, en lugar de su sintaxis superficial.

[1] eXtensible Stylesheet Language Transformations: estándar para transformar documentos XML

[2] XML Pointer Language: lenguaje base para identificar un fragmento mediante una referencia URI

[3] Uniform Resource Identifier: identificador uniforme de recurso

La sintaxis del lenguaje XPath definida por la 3WC[4] puede consultarse en documento [15] referenciado en la bibliografía de esta memoria.

A.1.7. Lucene

Lucene es librería para implementar motores de búsqueda de alto rendimiento desarrollada por Doug Cutting. El proyecto Lucene se encuentra actualmente bajo la gestión de la Apache Software Foundation.

Al tratarse de una librería de código, Lucene no es un servidor ni tampoco es un sistema de rastreo como lo son Solr o Nutch respectivamente. Lucene permite realizar tareas de indexación y consulta con prestaciones como:

- Un índice, basado en frecuencia inversa de términos, almacenado persistentemente para optimizar la extracción de los documentos que concuerdan con las consultas.
- Un amplio conjunto de analizadores para transformar una cadena de texto en una serie de términos (tokens), que son las unidades fundamentales de indexado y consulta.
- Una sintaxis y una variedad de tipos de consulta para obtener resultados de búsquedas difusas (fuzzy matches).
- Un algoritmo de puntuación basado en teorías de Recuperación de Información para mostrar los algoritmos ordenados por relevancia.
- Correctores ortográficos y detectores de lenguaje

[4] World Wide Web Consortium

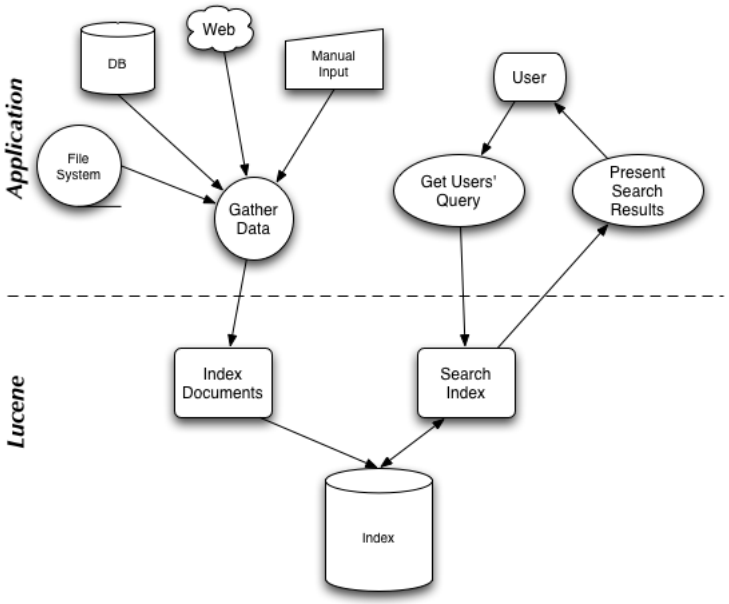


Figura A.5: Esquema de Lucene

La API proporcionada por la librería Lucene utiliza un conjunto reducido de clases Java. Las principales para operar con ella son:

- Document: es la unidad de búsqueda e indexación. Un índice consiste en uno o más Documents que son indexados o recuperados por las clases IndexWriter e IndexSearch respectivamente.
- Field: corresponde a cada uno de los campos del Document. Un Field es un par simple de valores donde el primero corresponde al nombre del campo y el segundo a su contenido.
- IndexWriter: es la encargada de generar el índice a través de la método addDocument(). El constructor de IndexWriter requiere la dirección del directorio para almacenar el índice y el analizador del contenido de los Documents.
- Analyser: proporciona un analizador estándar. Esta clase es responsable de analizar el texto y filtrar ciertas palabras comunes (stop words) a definir, por ejemplo, preposiciones.
- IndexSearcher: incluye las funcionalidades de búsqueda en el índice gestionando los resultados de cada consulta

A.1.8. Apache Solr

Apache Solr es la implementación de un servidor de motor de búsqueda en Java y basado en Lucene. Sin embargo, no es un simplemente un wrapper alrededor de las bibliotecas de Lucene, también aporta nuevas prestaciones como:

- Procesamiento de solicitudes HTTP[5] para la indexación y consulta de documentos.
- Sistemas de caché para minimizar la latencia de las consultas
- Una interfaz de administración web que incluye:
  - Tiempo de ejecución incluidas las estadísticas de rendimiento de aciertos de caché
  - Un formulario de consulta para buscar en el índice.
  - Un navegador con histogramas de términos populares, junto con algunas estadísticas.
  - Desglose detallado de métricas de relevancia y fases de análisis de texto.
- Búsqueda Facetada (Faceted Search) que permite añadir listas de criterios prácticos que ayudan al usuario a optimizar los resultados.
- Fichero XML de configuración que define la estructura del índice, el tipo de cada campo e información adicional.

[5]Hypertext Transfer Protocol: protocolo empleado en la World Wide Web

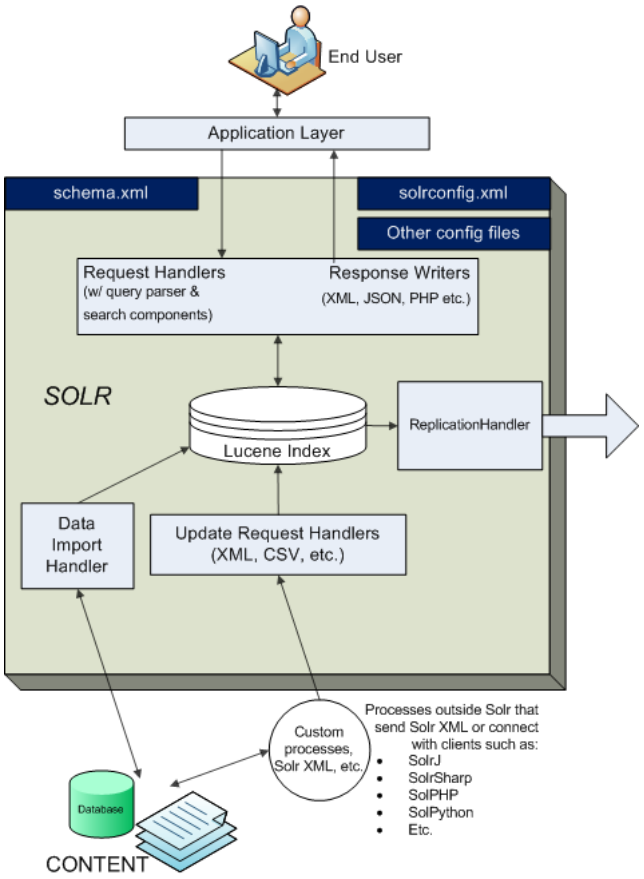


Figura A.6: Esquema de Solr

En Solr y Lucene, un índice se construye de uno o más Documents y un Document consta de uno o más Fields. Un Field puede ser descrito mediante una serie de opciones que indican a Solr cómo tratar el contenido durante la indexación y búsqueda. El cuadro A.3 muestra los principales atributos de un Field.

Nombre	Descripción
indexed	Los campos indexados se pueden buscar y ordenar. También pueden ser objeto de procesos de análisis sobre Solr para alterar el contenido y para mejorar o cambiar los resultados
stored	El contenido de un campo almacenados se guardan en el índice. De esta manera su contenido se mostrará en pantalla pero no podrán ser objeto de búsqueda.

Cuadro A.3: Atributos de field

Solr permite ejecutar un proceso de análisis para modificar el contenido antes de indexarlo. En Solr y Lucene, la clase Analyzer consiste en un clase Tokenizer y uno o más TokenFilters. La clase Tokenizer es responsable de producir tokens, que en la mayoría de los casos corresponden a las palabras a indexar. La clase TokenFilter extrae y filtra dichos tokens. Por ejemplo, la clase WhitespaceTokenizer de Solr divide las el texto en tokens separados por espacios en blanco, y la clase StopFilter elimina palabras de uso común en los resultados de búsqueda. Otros tipos de análisis incluyen derivados o la expansión de sinónimos.

Solr acepta mensajes HTTP GET y HTTP POST para las consultas que se procesan mediante la clase SolrRequestHandler. El cuadro A.4 muestra los principales parámetros disponibles en Solr.

Parámetro	Descripción	Ejemplo
start	Especifica el desplazamiento de partida en el conjunto de resultados, util para la paginación a través de resultados	0
rows	Número máximo de documentos para volver	10
fq	Proporcionar una consulta opcional de filtrado	
hl	Resultados destacados de la consulta.	true
fl	Especifica, con una lista separada por comas, el conjunto de campos que deben ser devueltos en los resultados del documento	*

Cuadro A.4: Principales parámetros

A.2. Herramientas

Durante el desarrollo de este PFC se han utilizado las siguientes herramientas:

Eclipse 1.2
Entorno de desarrollo integrado que a través de una interfaz de usuario permite desarrollar un proyecto Java ofreciendo funcionalidades como facilitar la configuración del proyecto, visualizar errores de compilación en tiempo real o realizar depuración de código.

Apache Ant 1.8.0
Herramienta usada para realizar tareas de compilación y construcción del proyecto.

TortoiseSVN 1.4.4
Cliente de Subversion para realizar una gestión del sistema de control de versiónes.

Mozilla Firefox 3.0.17
Navegador web utilizado para visualizar los blogs que se han empleado como muestra para configurar los filtros de extracción de entidades. Para facilitar esta tarea se ha añadido el plugin Firebug 1.4.4, para visualizar de manera instantánea y jerarquizada el código HTML.

Putty
Cliente SSH[6] para Windows empleado para acceder a las máquinas con sistema operativo Fedora y alojadas en el servidor de Amazon.

WinSCP
Cliente gráfico SSH para Windows, que configurado con Notepad++ como editor de texto, permite redactar y modificar ficheros de manera más accesible con funcionalidades como sintaxis coloreada de lenguajes, multivista de ficheros o buscar/reemplazar expresiones regulares.

Cygwin
Colección de herramientas que proporcionan la simulación de un sistema UNIX en Windows empleadas en fases iniciales del proceso desarrollo.

[6]Secure SHell: protocolo de acceso a máquinas remotas

Microsoft Office 2007 (MS Excel y MS Visio)
Herramientas de ofimática utilizadas para la elaboración de diagramas y gráficas de la memoria.

LyX y LaTeX
Editor de textos en LaTeX para la generación de toda la documentación escrita.

JabRef y BibTeX
Herramientas de gestión y mantenimiento de referencias bibliográficas.

Apéndice B

Manual de Usuario

Este manual ha sido desarrollado con la intención de que el lector adquiera el conocimiento necesario para la puesta en producción del sistema de rastreo en instancias de Amazon EC2. Incluye los pasos para gestionar tanto su cuenta en Amazon AWS, como el propio sistema de crawling.

El autor asume que el usuario dispone de una cuenta de Amazon así como los archivos correspondientes a la clave privada y el certificado.

B.1. Herramientas en línea de comandos

Por convención, todos los textos de línea de comandos se preceden con una línea de comandos genérica PROMPT>.

También se utiliza el símbolo $ para indicar comandos Linux/UNIX y C:\> para un comando específico de Windows. Las herramientas funcionan correctamente en Mac OS X con los comandos de Linux/UNIX.

B.1.1. Ajuste de la variable Java Home

Las herramientas en línea de comandos de Amazon EC2 requieren la versión Java 5 o posterior, o bien, una instalación de JRE o JDK. Descargue JRE para una gama de plataformas, incluyendo Linux/UNIX y Windows, accediendo a http://java.sun.com/j2se/1.5.0/.

Las herramientas en línea de comandos dependen de una variable de entorno (JAVA_HOME) para localizar Java Runtime. Esta variable de entorno debe establecerse en la ruta completa del directorio que contiene un subdirectorio llamado bin, que a su vez, contiene los ejecutables java (en Linux/UNIX) o java.exe (en Windows). Si no desea realizar este ajuste contínuamente, añada este directorio en la variable PATH. Asegúrese de no incluir el directorio

bin en el PATH, error común de algunos usuarios.

Nota Si está utilizando Cygwin, EC2_HOME, EC2_PRIVATE_KEY y EC2_CERT, debe utilizar rutas Linux/UNIX (por ejemplo, /usr/bin en lugar de C:\usr\bin). Sin embargo, JAVA_HOME debe tener una ruta de acceso de Windows. Además, EC2_HOME no puede contener espacios en blanco.

Ejemplo de cómo configurar la variable JAVA_HOME en Linux/UNIX :

$ export JAVA_HOME=<PATH>

Ejemplo de la sintaxis en Windows:

C:\> set JAVA_HOME=<PATH>

Puede confirmar la configuración verificando la salida del siguiente comando:

$JAVA_HOME/bin/java -version java version "1.5.0_09"
Java(TM) 2 Runtime Environment, Standard Edition (build 1.5.0_09-b03)
Java HotSpot(TM) Client VM (build 1.5.0_09-b03, mixed mode, sharing)

La sintaxis en Windows es diferente aunque la salida es similar:

C:\> %JAVA_HOME%\bin\java -version
java version "1.5.0_09"
Java(TM) 2 Runtime Environment, Standard Edition (build 1.5.0_09-b03)
Java HotSpot(TM) Client VM (build 1.5.0_09-b03, mixed mode, sharing)

B.1.2. Configuración de las herramientas

Para utilizar Amazon EC2, es necesario descargar las herramientas en línea de comandos y configurarlas.

B.1.2.1. Obtención de las herramientas

Las herramientas en línea de comandos están disponibles como un archivo ZIP en el Centro de Recursos de Amazon EC2[1]. Estas herramientas, implementadas en Java, incluyen scripts de shell para Windows 2000/XP y Linux/UNIX/Mac. El archivo ZIP es autónomo, no requiere instalación. Simplemente descárguelo y descomprímalo.

Es necesario configurar algunos parámetros para que las herramientas utilicen sus credenciales de cuenta de AWS. Estos se discuten a continuación.

[1] http://developer.amazonwebservices.com/connect/entry.jspa?externalID=351&categoryID=88

B.1.2.2. Configuración de los parámetros de entorno

Las herramientas en línea de comandos dependen de una variable de entorno (EC2_HOME) para localizar las bibliotecas de apoyo. Tendrá que configurarla indicando la ruta del directorio en el que se descomprimieron las herramientas. Este directorio se denomina EC2-api-tools-AB-nnnn (A, B y n corresponden a la versión), y contiene subdirectorios bin y lib.

En Linux y UNIX, se puede ajustar de este modo:

$ export EC2_HOME=<path-to-tools>

En Windows la sintaxis es ligeramente diferente:

C:\> set EC2_HOME=<path-to-tools>

Además es probable que desee agregar el directorio bin al PATH de su equipo. De hecho, el resto del manual supondrá esta afirmación.

En Linux y UNIX, puede actualizar su PATH de la siguiente manera:

$ export PATH=$PATH:$EC2_HOME/bin

En Windows la sintaxis es ligeramente diferente:

C:\> set PATH= %PATH %; %EC2_HOME %\bin

B.1.2.3. Configuración de los parámetros de cuenta Amazon AWS

Las herramientas en línea de comandos necesitan tener acceso a la clave privada y a un certificado X.509 generados tras registrarse en Amazon EC2.

Es posible utilizar dos variables de entorno que apunten a su clave privada y certificado. Si estas variables de entorno se establecen, las herramientas emplean sus valores para encontrar las credenciales pertinentes. La variable de entorno EC2_PRIVATE_KEY debe hacer referencia a su archivo de clave privada, y a su vez, la variable EC2_CERT a su certificado X.509.

En Linux/UNIX, fije estas variables de entorno de esta manera:

$ export EC2_PRIVATE_KEY=~/.ec2/pk-HKZYKTAIG2ECM.pem
$ export EC2_CERT=~/.ec2/cert-HKZYKTAIG2ECMXYIBH3.pem

En Windows la sintaxis es ligeramente diferente:

C:\> set EC2_PRIVATE_KEY=c:\ec2\pk-HKZYKTAIG2ECM.pem
C:\> set EC2_CERT=c:\ec2\cert-HKZYKTAIG2ECMXYIBH3.pem

B.1.2.4. Configuración de los parámetros de región

Por defecto, las herramientas de Amazon EC2 utilizan la zona este de los Estados Unidos (us-east-1), con ec2.us-east-1.amazonaws.com como terminación de la URL del servicio. Si desea ver las regiones disponibles introduzca:

PROMPT> ec2-describe-regions
REGION ap-southeast-1 ec2.ap-southeast-1.amazonaws.com

REGION eu-west-1 ec2.eu-west-1.amazonaws.com
REGION us-east-1 ec2.us-east-1.amazonaws.com
REGION us-west-1 ec2.us-west-1.amazonaws.com

Las instancias para el sistema de rastreo se localizan en Europa Occidental así que modifique el valor de la variable EC2_URL:

- Para Linux y UNIX: $ export EC2_URL=https://ec2.eu-west-1.amazonaws.com
- Para Windows: C:\> set EC2_URL=https://ec2.eu-west-1.amazonaws.com

B.1.2.5. Configuración de los puertos

Antes de acceder a la instancia, debe abrir los puertos necesarios para su conexión. De igual manera el sistema de rastreo necesitará tener configurados puertos específicos.

Utilice los siguientes comandos para abrir el puerto 22 (SSH, SCP, SFTP):

PROMPT> ec2-authorize default -p 22
GROUP default
PERMISSION default ALLOWS tcp 22 22 FROM CIDR 0.0.0.0/0

Repita este comando para el puerto HTTP (80) y los puertos del sistema de rastreo (50030, 50060, 50070, 50075, 50090, 9000 y 9001).

B.1.3. Generación un par de claves SSH

Cuando se pone en marcha una instancia de una AMI pública, en lugar de contraseña, se necesita un par de claves pública/privada para acceder a la instancia. La mitad de este par de claves está embebida en la instancia. Después de aprender a crear sus propias imágenes, puede elegir otros mecanismos que le permiten iniciar sesión de forma segura a todas las instancias nuevas. Cada par de claves que generan requiere un nombre. Asegúrese de escoger un nombre que sea fácil de recordar.

Para generar un par de claves usando gsg-keypair introduzca la siguiente información

PROMPT> ec2-add-keypair gsg-keypair

Amazon EC2 devuelve un par de claves similares a la del ejemplo.

KEYPAIR gsg-keypair 1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f
—–BEGIN RSA PRIVATE KEY——
MIIEoQIBAAKCAQBuLFg5ujHrtm1jnutSuoO8Xe56LlT+HM8v/xkaa39Est
HungXQ29VTc8rc1bW0lkdi23OH5eqkMHGhvEwqa0HWASUMll4o3o/IX+
5AU52EQfanIn3ZQ8lFW7Edp5a3q4DhjGlUKToHVbicL5E+g45zfB95wIyy
ebIUlq1qTbHkLbCC2r7RTn8vpQWp47BGVYGtGSBMpTRP5hnbzzuqj3itki
i8BygR4s3mHKBj8l+ePQxG1kGbF6R4yg6sECmXn17MRQVXODNHZbAg

91CXirkYGuVfLyLflXenxfI50mDFms/mumTqloHO7tr0oriHDR5K7wMcY/
ZNUJs7rw9gZRTrf7LylaJ58kOcyajw8TsC4e4LPbFaHwS1d6K8rXh64o6W
3wcfgt5ecIu4TZf0OE9IHjn+2eRlsrjBdeORi7KiUNC/pAG23I6MdDOFEQR
SWS4dMbrpb9FNSIcf9dcLxVM7/6KxgJNfZc9XWzUw77Jg8x92Zd0fVhHO
tE8C3p9bbU9VGyY5vLCAiIb4qQKBgQDLiO24GXrIkswF32YtBBMuVgL
jUE5IpzRjTedc9I2qiIMUTwtgnw42auSCzbUeYMURPtDqyQ7p6AjMujp9E
xW9MC0dtV6iPkCN7gOqiZXPRKaFbWADp16p8UAIvS/a5XXk5jwKBgQ
iDCiK6JBRsMvpLbc0v5dKwP5alo1fmdR5PJaV2qvZSj5CYNpMAy1/EDN
rdLNLDL4+TcnT7c62/aH01ohYaf/VCbRhtLlBfqGoQc7+sAc8vmKkesnF7C
gC0iZzzNAapayz1+JcVTwwEid6j9JqNXbBc+Z2YwMi+T0Fv/P/hwkX/ype
DQbsz7LcY1HqXiHKYNWNvXgwwO+oiChjxvEkSdsTTIfnK4VSCvU9Bx
rBYvChJZF7LvUH4YmVpHAoGAbZ2X7XvoeEO+uZ58/BGKOIGHByHB
gK+8zp4L9IbvLGDMJO8vft32XPEWuvI8twCzFH+CsWLQADZMZKSsBa
JZKjTSu3i7vhvx6RzdSedXEMNTZWN4qlIx3kR5aHcukCgYA9T+Zrvm1F
P8TTvW/6bdPi23ExzxZn7KOdrfclYRph1LHMpAONv/x2xALIf91UB+v5o
2ERKKdwz0ZL9SWq6VTdhr/5G994CK72fy5WhyERbDjUIdHaK3M849JJ
—–END RSA PRIVATE KEY—-

La clave privada devuelta debe ser guardada en un archivo local para que pueda usarse más tarde. Cree un archivo denominado id_rsa-gsg-keypair y pegue toda la clave generada en el paso anterior, incluyendo las siguientes líneas.

"—-BEGIN RSA PRIVATE KEY—-"
"—-END RSA PRIVATE KEY—-"

Confirme que el contenido del archivo es similar al del ejemplo y guarde el archivo. Recuerde que si no opta por almacenarlo en el directorio actual, se debe especificar la ruta completa al usar comandos que requieren que el par de claves.

En este punto, ya está preparado para usar Amazon EC2.

B.2. Gestión de imágenes de máquinas, instancias y volúmenes

Esta sección abarca lo relacionado con todas las acciones para administrar AMIs, instancias y volúmenes de Amazon EC2.

B.2.1. Creación de una instancia

La AMI (Amazon Machine Instance) diseñada por Cierzo para el sistema de rastreo se identifica como ami-CIERZO. Para crear una nueva instancia de esta AMI use el comando ec2-run-instances:

PROMPT> ec2-run-instances ami-CIERZO -k gsg-keypair -t m1.large

Amazon EC2 devolverá una salida similar a la siguiente:

RESERVATION r-f25e6f9a 999988887777 default
INSTANCE i-85b435ee ami-CIERZO pending gsg-keypair 0
m1.large 2010-03-30T08:01:36+0000
eu-west-1a aki-94c527fd ari-96c527ff monitoring-disabled ebs

B.2.2. Detención y arranque de una instancia

En esta sección describe cómo iniciar y detener instancias que utilizan volúmenes Amazon como dispositivos raíz. Una instancia parada permitirá al usuario conservarla sin generar costes mientras se mantenga detenida.

Para detener una instancia use el comando ec2-stop-instances:

PROMPT> ec2-stop-instances i-85b435ee

Amazon EC2 devuelve una salida similar a la siguiente.

IMAGE i-85b435ee running stopping

Si desea ponerla en marcha, emplee el commando ec2-start-instances:

PROMPT> ec2-start-instances i-85b435ee

La salida será similar a la del ejemplo.

IMAGE i-85b435ee stopped pending

B.2.3. Borrado de una instancia

Cuando ya no se requiere la utilización de una instancia puede borrarse con el comando ec2-terminate-instances:

PROMPT> ec2-terminate-instances i-85b435ee

La salida será similar a la del ejemplo.

IMAGE i-85b435ee running shutting-down

B.2.4. Creación y asociación de volúmenes de datos

Si la instancia creada no corresponde al AMI creado por Cierzo Development necesitará un volumen de datos EBS asociado a ella.

Introduzca el siguiente commando:

PROMPT> ec2-create-volume –size 1000 -z eu-west-1b

Amazon EBS devuelve una información del volumen similar a la del ejemplo.

VOLUME vol-c7f95aae 1000 eu-west-1b creating 2010-03-30T13:54:37+0000

Por ultimo vincule el volumen con la instancia en el dispositivo /dev/sdg:

ec2-attach-volume -d /dev/sdg -i i-85b435ee vol-c7f95aae

B.2.5. Disociación de volúmenes de datos

En ocasiones un volumen desea desconectarse de una instancia manteniéndose su informacion, por ejemplo para adjuntarla a otra. Para ello utilice el siguiente comando:

ec2-dettach-volume -d /dev/sdg -i i-85b435ee vol-c7f95aae -f

B.2.6. Borrado de volúmenes de datos

Al igual que con las instancias, puede borrar volúmenes con el comando ec2-delete-volume:

PROMPT> ec2-delete-volume vol-4282672b
VOLUME vol-4282672b

B.2.7. Creación de un AMI

Para crear un AMI utilice el comando ec2-create-image:

PROMPT> ec2-create-image -n "My AMI" i-eb977f82

Amazon EBS devolverá la información sobre el la AMI creada:

IMAGE ami-8675309.

B.3. Conexión a las instancias

Para conectar con una instancia Unix/Linux de Amazon se requiere un par de claves SSH. Esta sección también incluye la conexión a instancias Linux desde Windows a través de PuTTY.

B.3.1. Conexión a una instancia desde Windows con PuTTY

PuTTY es un cliente SSH gratuito para Windows. La suite de PuTTY también incluye aplicacions como PuTTYgen, un programa de generación de claves.

B.3.1.1. Conversión del formato de clave privada

PuTTY no soporta nativamente el formato de la clave privada generada por Amazon EC2. Afortunadamente, PuTTYgen puede convertir claves a su formato interno. Para configurar PuTTY, ejecute PuTTYgen y cargue id_rsa-gsg-keypair. PuTTYgen debería mostrar el siguiente mensaje:

PuTTYgen muestra una gran cantidad de información sobre la clave que se ha cargado, como la clave pública, la frase de paso clave, el tipo y el número de bits en la clave generada. Las claves generadas por Amazon EC2 son de claves SSH-2 RSA de 1024 bits. Con esta clave el acceso con PuTTY a las instancias no requiere contraseña.

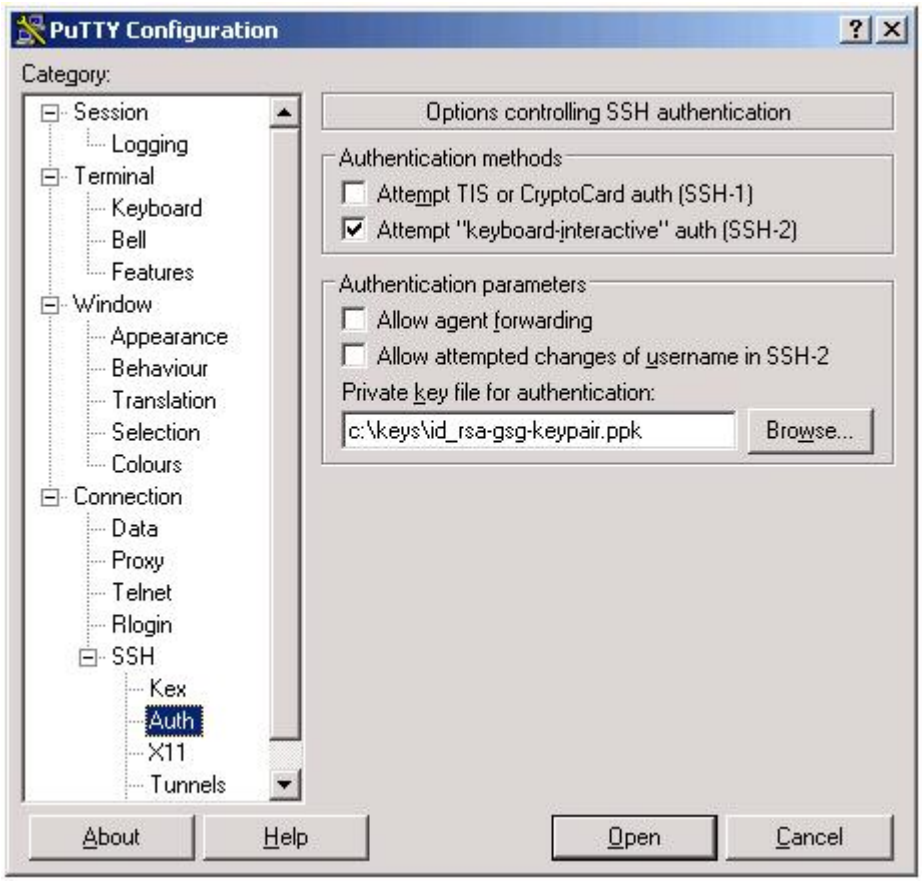
Puede seleccionar Guardar en el menú Archivo o haga clic en Guardar clave privada. Asígnele un id_rsa-gsg-keypair.ppk como nombre de archivo. Cuando PuTTYgen le pide que guarde la clave sin contraseña, haga clic en Sí.

El archivo se puede utilizar con PuTTY para conectarse a su host de Amazon EC2 como se describe a continuación.

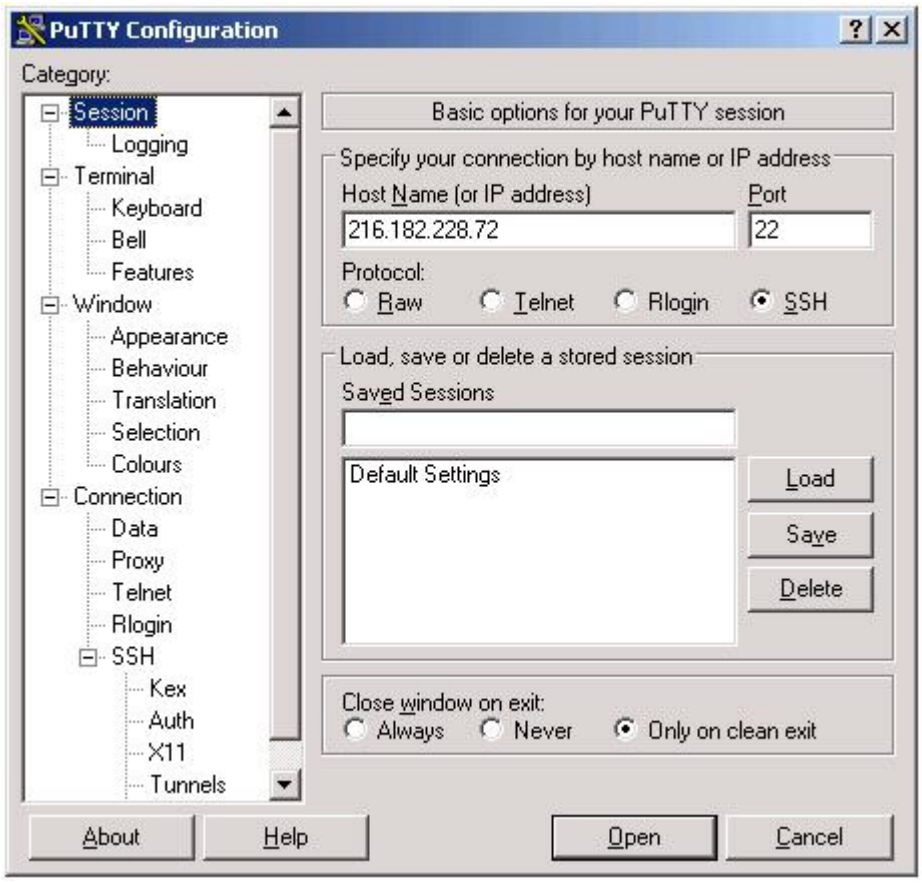
B.3.1.2. SSH con PuTTY

En esta sección supone que ha convertido la clave privada generada por Amazon EC2 archivo a una clave privada para PuTTY, y también, ha levantado una instancia.

Ejecute PuTTY y verá una interfaz como la de la imagen. Haga clic en Connection, SSH y seleccione Auth. Con el botón Browse podrá seleccionar el archivo de la clave privada para PuTTY generada anteriormente (id_rsa-gsg-keypair.ppk en este manual)



En Session, introduzca root@hostname or root@ip_address. Haga clic en Open para conectar con su instancia de Amazon.



B.4. Configuración de las instancias de rastreo

Si las instancias levantadas corresponden al AMI específico definido por Cierzo Development, únicamente habrá que realizar el montaje del volumen EBS y confirmar el acceso sin contraseña.

B.4.1. Formateo del Volumen de datos

Simplemente tendrá que formatear /dev/sdg y crear el punto de montaje:
$ yes | mkfs -t ext3 /dev/sdg
$ mkdir /nutch

B.4.2. Montaje del Volumen de datos

Tanto si está utilizando la AMI diseñada por Cierzo, como si ha utilidad otro tipo de AMI y un volumen de datos, deberá realizar el siguiente montaje:

$ mount /dev/sdg /nutch

B.4.3. Configurar el fichero de hosts

La configuración de Nutch requiere indicar qué máquinas serán esclavo y qué máquina será el maestro. Por cuestiones de legibilidad se recomienda asignar nombres, en cada instancia, de las máquinas que conforman el clúster. Para ello, se edita el fichero /etc/hosts de la siguiente manera:

127.0.0.1 localhost.localdomain localhost
XXX.XXX.XXX.XXX maestro
XXX.XXX.XXX.XXX slave1
XXX.XXX.XXX.XXX slave2
XXX.XXX.XXX.XXX slave3
...

B.4.4. Configurar el acceso SSH sin contraseña

Se accede al directorio de SSH y se crear el archivo identificador de clave pública:

$ touch /root/.ssh/id_rsa

Una vez creado se edita con el contenido de la clave privada (gsg-keypair). Es probable que haya que modificar los permisos del fichero con el siguiente comando:

$ chmod 600 /root/.ssh/id_rsa

B.4.5. Confirmación del acceso sin contraseña

Una vez configurado el acceso sin contraseña y el fichero con los hosts del clúster, la comunicación SSH se realizará acorde con las exigencias del rastreo. Sin embargo, la primera conexión entre dos máquinas exige una confirmación (yes/no) por parte del usuario.

Ya que la comunicación en el clúster es maestro/esclavo, es deseable conectar desde el maestro con todos los esclavos, y viceversa, para no tener que realizar dicha confirmación en la primera ejecución del rastreo (también habrá que realizar una conexión maestro-maestro).

B.4.6. Instalación del Java Development Kit (JDK)

Para poder utilizar las herramientas basadas en Java es necesario instalar Java Development Kit.

El ejemplo de este manual se utiliza la versión Java SE Development Kit 6u20 for Linux, Multi-language. Si existen versiones superiores simplemente hay que sustituir los nombres de los archivos de los pasos de este manual .

En la web oficial de Sun puede encontrar un link[2] de descarga del JDK para Linux con el nombre jdk-6u20-linux-i586-rpm.bin. La instalación se realiza de la siguiente manera:

$ wget <link_descarga>
$ mv <archivo_descargado> jdk-6u20-linux-i586-rpm.bin
$ chmod 755 jdk-6u20-linux-i586-rpm.bin
$ ./jdk-6u20-linux-i586-rpm.bin
[seguir la instalación]

Si desea que la variable de entorno JAVA_HOME, esté establecida cada vez que inicie sesión, ha de introducir la siguiente línea en el fichero /etc/profile:

export JAVA_HOME=/usr/java/jdk1.6.0_20

Por último se borran archivos generados durante la instalación

$ rm -rf jdk*
$ rm -rf sun*

B.4.7. Instalación de Apache ANT

La instalación de Apache Ant permite la programación de tareas en Java. De esta manera, se simplifica la compilación y ejecución de las herramientas del sistema. El siguiente ejemplo muestra como instalar la versión 1.8:

$ wget http://www.apache.org/dist/ant/binaries/apache-ant-1.8.0-bin.tar.gz
$ mkdir /usr/local/ant
$ mv apache-ant-1.8.0-bin.tar.gz /usr/local/ant/
$ cd /usr/local/ant
$ tar -xzvf apache-ant-1.8.0-bin.tar.gz

Por último, al igual que con JDK, se recomienda establecer las variables de Ant incluyendo las siguientes líneas en el archivo /etc/profile:

ANT_HOME="/usr/local/ant/apache-ant-1.8.0/"
PATH="$PATH:/usr/local/ant/apache-ant-1.8.0/bin"
export ANT_HOME
export PATH

Apache Ant estará disponible la próxima vez que se inicie sesión

[2] http://java.sun.com/javase/downloads/widget/jdk6.jsp

B.4.8. Instalación de Subversion

Con Yum, herramienta de software libre de gestión de paquetes para sistemas Linux, instale Subversion de la siguiente manera:

$ yum install svn
[seguir la instalación]

B.4.9. Instalación de GNU Screen

Ya se ha visto anteriormente como conectar con las instancias de Amazon a través de PuTTY. Sin embargo, debe poder cerrar el terminal sin cerrar la sesión, y también, reconectar con una sesión previa. Para ello, descargue el proyecto GNU Screen:

$ yum install screen
[seguir la instalación]

Con esta herramienta se simplifica la gestión de múltiples sesiones desde un mismo terminal. Los comandos básicos son los siguientes:

Abrir una nueva sesión:

$ screen

Desconectar de una sesión sin cerrarla

$ <CTRL+A+D>

Listar las sesiones previas:

$ screen -list

Reconectar con una sesion previa:

$ screen -r [pid.]tty.host

B.4.10. Instalación de MySQL y MySQL Server

La instalación de MySQL y MySQL Server también se realiza a través de Yum. Una vez instalados solo hay que configurar los usuarios, contraseñas, schemas y tablas del servidor.

$ yum install mysql mysql-server
[seguir la instalación]
$ chkconfig –level 2345 mysqld on; service mysqld start
$ mysql -u root
mysql> delete from mysql.user where not (host="localhost" and user="root");
mysql> flush privileges;
mysql> set password for 'root'@'localhost' = password('1234');
mysql> flush privileges;
mysql> create schema pfc;
mysql> use pfc;

mysql> create table tbsegment (segment char(255) primary key,parsed int,started datetime,ended datetime);
mysql> exit

B.4.11. Instalación del sistema de rastreo

Para obtener la última versión del sistema de rastreo simplemente hay que descargarla del repositorio de Subversion con el siguiente comando:

$ cd /nutch
$ svn checkout http://87.238.90.178:8000/svn/nutch

Nota: El usuario y la contraseña son XXXX y 'XXXX' respectivamente

B.5. Configuración y gestión del sistema de rastreo

En esta última sección se explican los detalles para que sea capaz de gestionar el sistema de rastreo.

B.5.1. Configuración

Tanto si se ha utilizado la AMI diseñada por Cierzo Development, como si se ha configurado una AMI convencional, el sistema de rastreo se encontrara en la ruta /nutch/nutch.

En el subdirectorio conf se encuentran los ficheros con los parámetros que definen el crawling. La versión que figura en la AMI es la misma que la que puede encontrar en el repositorio de Subversion, ya configurada con los párametros adecuados. Muchos de ellos son los propios del proyecto Nutch, sin embargo, otros se han modificado para ajustar la configuración del clúster y la configuración del sistema de rastreo.

B.5.1.1. Configuración del clúster

fichero: core-site.xml
fs.default.name hdfs://master:9000/
fichero: hadoop-env.sh
HADOOP_HOME /nutch/nutch
JAVA_HOME /usr/java/jdk1.6.0_20
HADOOP_LOG_DIR ${HADOOP_HOME}/logs
HADOOP_SLAVES ${HADOOP_HOME}/conf/slaves

HADOOP_HEAPSIZE 1024
HADOOP_NUTCH 1024
fichero: hdfs-site.xml
dfs.name.dir /nutch/filesystem/name
dfs.data.dir /nutch/filesystem/data
dfs.replication 3
fichero: mapred-site.xml
mapred.job.tracker hdfs://master:9001/
mapred.map.tasks 2 x nº slaves
mapred.reduce.tasks 2 x nº slaves
mapred.system.dir /nutch/filesystem/mapreduce/system
mapred.local.dir /nutch/filesystem/mapreduce/local
mapred.child.java.opts -Xmx1536m
mapred.task.timeout 6000000
fichero: masters
master
fichero: slaves
slave1
...
slave n

B.5.1.2. Configuración del sistema de rastreo

crawl-urlfilter.txt
Expresiones regulares que definen las urls del crawling
nutch-default.xml
http.agent.name cierzo-development
db.fetch.interval.default 259200
db.parsemeta.to.crawldb lang
fetcher.server.delay 2.0
fetcher.server.min.delay 0.0
fetcher.threads.fetch 500
fetcher.threads.per.host 5

B.5.2. Gestión del cluster

Si ha configurado los parámetros del clúster correctamente, la creación de éste es muy seimple. Introduzca los siguientes comandos para crear el HDFS y posteriormente iniciar Hadoop:

$ cd /nutch/nutch
$ bin/hadoop namenode -format

$ bin/start-all.sh

Una vez creado Hadoop ofrece herramientas para gestionar el clúster. Para conocer todos los comandos disponibles ejecute el comando general de Hadoop:

$ bin/hadoop

Por ejemplo, acciones como pasar ficheros de local al HDFS y viceversa se ejecutan de este modo:

$ bin/hadoop dfs -put ficherolocal ficheroHDFS
$ bin/hadoop dfs -get ficheroHDFS ficherolocal

B.5.3. Ejecución

El sistema de rastreo se pone en funcionamiento con el siguiente script:

$ cd /nutch/nutch
$ bin/nutch crawl

El sistema informará del uso del comando teniendo que indicar el fichero con las URLs a inyectar y pudiendo establecer el directorio donde esta la base de datos del crawl, el número de iteraciones, el número de threads o el máximo de URLs a tratar por iteracion. Por ejemplo:

$ bin/nutch crawl urls -dir crawl -depth 3 -threads 500 -topN 2000000

Con este comando el sistema realizará 3 iteraciones con 500 threads de 2 millones de URLs por iteración provenientes del fichero/directorio urls y de la base de datos almacenada en la carpeta crawl.

Apéndice C

Interfaz Web

C.1. Mapa de navegación

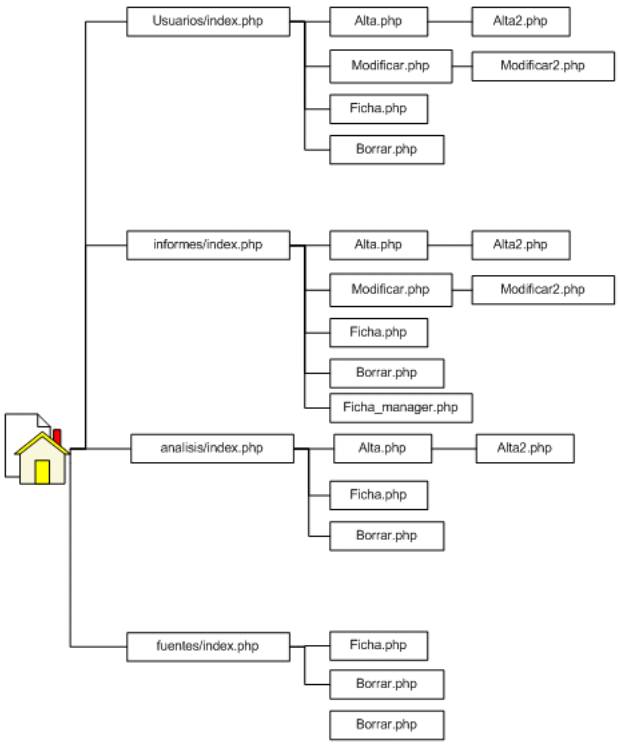


Figura C.1:

C.2. Pantallas web

C.2.1. Bienvenido a Planificador (//index.php)

C.2.2. Zonas comunes

C.2.3. Cabecera.php

Debe contener las siguientes entidades:

- Mail del usuario
- Acceso a 'Mi cuenta' (//usuarios/ficha.php)
- Salir (//modulos/salir.php)
- Informes (select con los informes del usuario que cambia el informe en la sesión)

C.2.4. Pie.php

Debe contener las siguientes entidades:

- Desarrollado por Cierzo Development (http://www.cierzo–development.com)
- Plataforma Smmart (http://www.analisisdemedios.es)

C.2.5. Acceso.php

Archivo que verifica la presencia del usuario en la base de datos. Si el usuario es el administrador redirige a usuarios/index.php, si el usuario está en la base de datos redirige a informes/index.php y si no lo está al login con confirmación del error.

C.2.6. Listado de usuarios (//usuarios/index.php)

Usuarios del sistema

- Nº
- Id
- Mail (//informes/index.php)
- Nº de informes

- Checkbox de borrado

Botonera

- Nuevo usuario (//usuarios/alta.php)
- Borrar (//usuarios/borrar.php)

C.2.7. Alta de un usuario (//usuarios/alta.php)

Formulario de alta de un usuario

- Email
- Verificar email (comprueba que el campo es similar a mail)
- Contraseña
- Nuevo usuario (formulario por post a //usuarios/alta2.php)

C.2.8. Borrar usuarios (//usuarios/borrar.php)

Proviene de index.php y borra los usuarios seleccionados. Redirige a index.php con mensaje en un layer (desaparece tras 5 segundos -> prototype, motolos, etc).

C.2.9. Ficha de un usuario (//usuarios/ficha.php)

Datos del usuario

- Email
- Contraseña
- Nuevo usuario (formulario por post a //usuarios/modificar2.php)

Informes del usuario (//informes/index.php)

- Nombre (//informes/ficha.php)
- Descripción
- Fecha inicio
- Fecha fin
- Borrar (//informes/borrar.php ? verificación javascript)

C.2.10. Listado de informes (//informes/index.php)

Recibe el usuario de la sesión y recupera todos sus informes. Informes disponibles:

- Nº
- Id
- Nombre (//informes/ficha.php)
- Descripción
- Fecha inicio
- Fecha fin
- Nº Posts – nº hosts
- Borrar (//informes/borrar.php)

Botonera

- Nuevo usuario (//informes/alta.php)
- Borrar (//informes/borrar.php)

Alta de un informe (//informes/alta.php)

A continuación se muestra de forma simplificada el pre diseño.

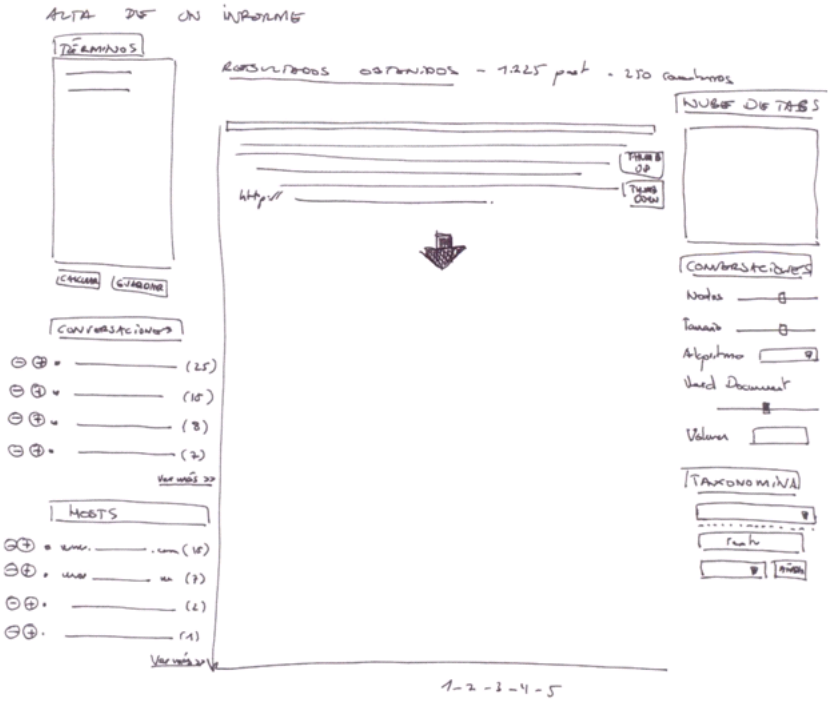


Figura C.2:

Esta pantalla es muy importante puesto que debe definir los parámetros de configuración del informe y dependiendo de estos vamos a extraer una información u otra. A continuación vamos a explicar cada uno de los módulos que la componen de izquierda a derecha.

- Términos, este textarea nos permite incorporar los términos de búsqueda textual o fulltext search que el sistema va a lanzar contra el API del smmart. Debajo aparecen los botones 'rastrear', que ejecuta una nueva búsqueda demo (limitada a 200 documentos) que nos permite comprobar la configuración y 'guardar' que almacena la configuración en la base de datos y lanza los sistemas de reporting para la extracción de información.

- Conversaciones (izq), componente donde aparecen las conversaciones detectadas y el volumen de cada una de ella. Pulsando sobre el nombre de la conversación se efectúa un filtrado ajax de los resultados. Además

aparecen los botones +- que permiten incorporar el nombre de la conversación a los términos de búsqueda para que al recalcular sea tenido en cuenta.

- Hosts, volumen agregado por host, al igual que en conversaciones podemos añadir filtro positivo (sólo entradas de un host) o negativo (todas las entradas menos las de un host concreto) a los criterios de búsqueda.
- Resultados de búsqueda, una vez hemos pulsado 'calcular' y el sistema ha extraído la información, ésta es mostrada en este listado. Los campos que aparecen son título, fecha, descripción (300ch max.) y la url destino.
- Nube de tags, tokenización y conteo de los términos más importantes que aparecen en el informe. Este sistema no estará compuesto por agregación de los tokens solamente sino que tendrá que incorporar términos de sugerencia semántica a través de las Apis (adwords, smmart, yahoo boss, etc).
- Conversaciones (dcha.), componente que define los parámetros del sistema detector de conversaciones.

Cada vez que se realice el sistema de cálculo el sistema mostrará un layer con las diferencias con el parámetro anterior en volumen de posts, comentarios y hosts y número de conversaciones. Además de guardar la configuración anterior para poder 'deshacer' y recalcular según los parámetros anteriores.

C.2.11. Borrar informe (//informes/borrar.php)

Realiza el proceso de borrado de toda la información referente al informe y vuelve a //informes/index.php

C.2.12. Ficha de un informe (//informes/ficha.php)

Se muestra en forma de pre diseño.

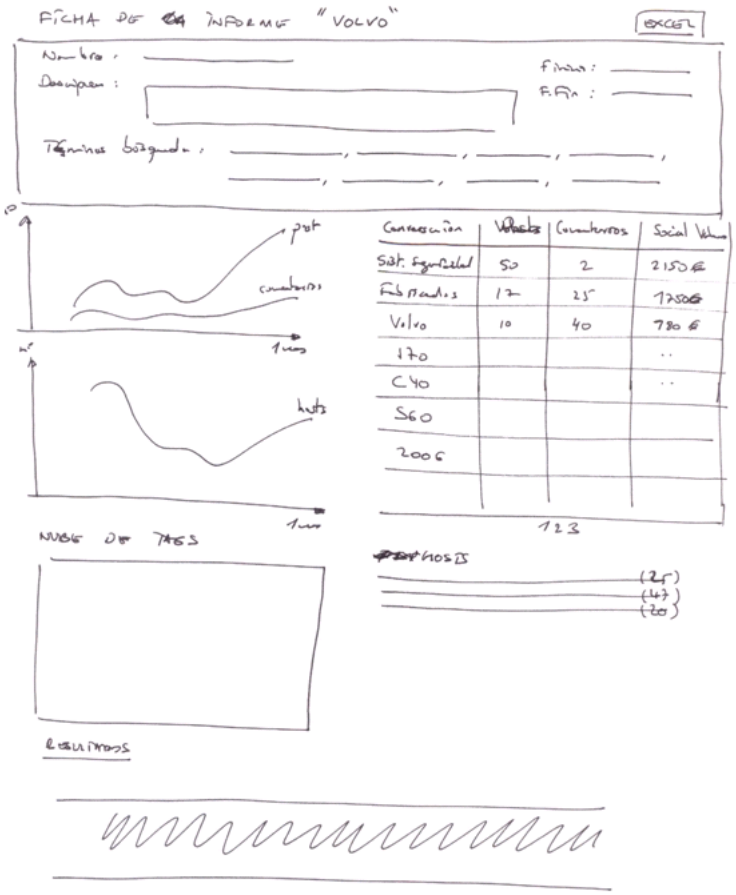


Figura C.3:

A continuación se detalla la información que va a contener esta pantalla:

- Ficha del informe, en esta pestaña mostraremos la información general del informe, es decir los parámetros de configuración. Para hacer más sencilla la navegación a través de una opción debemos poder contraer este layer.
- Gráfica con el volumen general de posts y comentarios en el último mes
- Gráfica con el número de host aparecidos en el último mes
- Tabla con las conversaciones aparecidas y resultados de cada una de ellas por posts y comentarios así como el valor social media detectado. La tabla debe permitir filtrados al pulsar sobre una de las columnas.
- Nube de tags (fuente el texto de las entradas y comentarios)

- Listado del volumen por hosts
- Listado de las entradas aparecidas

La siguiente imagen muestra a modo de ejemplo como quedarían los resultados agregados por conversaciones de uno de los informes que actualmente tenemos en curso (algoritmo I):

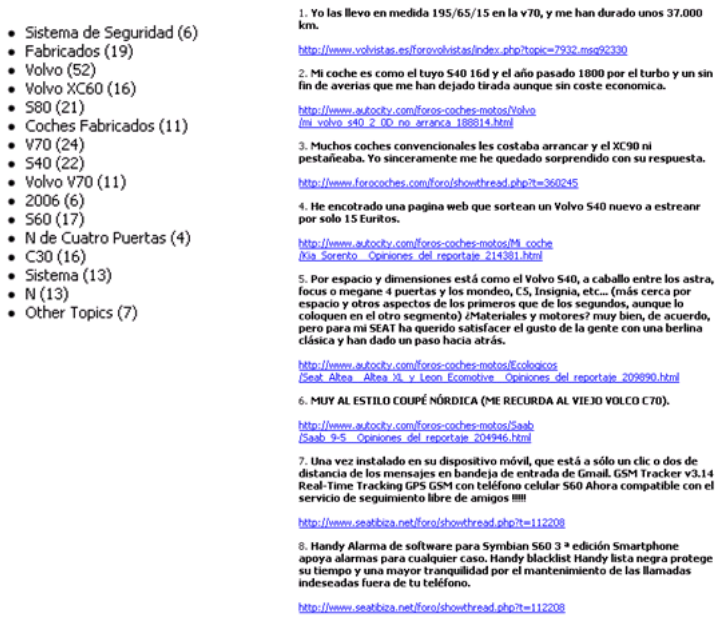


Figura C.4:

A continuación se muestran las conversaciones que sobre ese mismo conjunto de datos se obtienen con el algoritmo detector II.

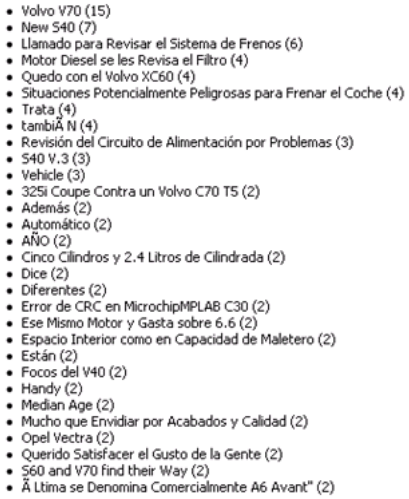


Figura C.5:

Como sea podido apreciar el primer algoritmo ofrece menos resultados pero con una agregación más exacta (como si fuesen categorías) el segundo por el contrario ofrece más resultados, menos agregados pero mucho más descriptivos de cara al etiquetado de conversaciones.

C.2.13. Ficha de un informe (//informes/ficha_manager.php)

Esta ficha corresponde a los managers para que puedan consultar la información aparecida en un informe. El sistema debe permitir el filtrado por fechas, fuente, dominio y analizar el contenido a partir de un término de búsqueda. La tabla debe permitir ordenación.

fuente	dominio	url	fecha	texto	etiqueta	cluster
Post	www.xxx.com	www.xxx.com/posts	12-05-2010	Texto texto texto...	motor	Xs60
Foro	www.xxx.com	www.xxx.com/posts	12-05-2010	Texto texto texto...	motor	Xs60
Post	www.xxx.com	www.xxx.com/posts	12-05-2010	Texto texto texto...	motor	Xs60
post	www.xxx.com	www.xxx.com/posts	12-05-2010	Texto texto texto...	motor	Xs60

Figura C.6:

C.2.14. Listado de análisis (//analisis/index.php)

Esta información aparecerá en un componente en la zona izquierda de la navegación (similar a la zona de filtrado de las búsquedas de google).

- Fecha inicio
- Fecha fin
- Nombre
- Borrar (//análisis/borrar.php)
- Nuevo (//análisis/alta.php)

C.2.15. Alta de un análisis (//analisis/alta.php)

Como hemos indicado previamente un análisis refleja una foto fija del proceso de extracción de información que hemos realizado y que queremos mantener en nuestra base de datos para su posterior consulta. El proceso de alta de un análisis nos debe permitir por tanto, definir una serie de parámetros de búsqueda en los datos almacenados y guardar los resultados obtenidos. La pantalla de definición es similar a la de alta de un informe pero se tiene que tener en cuenta que es un filtrado sobre la información guardada en nuestra base de datos)

Figura C.7:

C.2.16. Borrar análisis (//analisis/borrar.php)

C.2.17. Ficha de un análisis (//analisis/ficha.php)

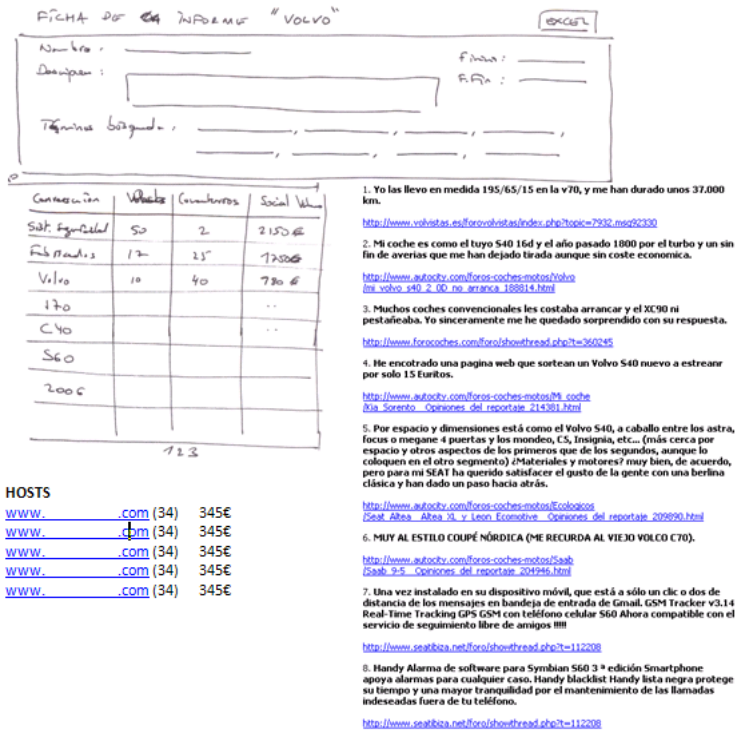


Figura C.8:

C.2.18. Ficha de una conversación (//conversación/ficha.php)

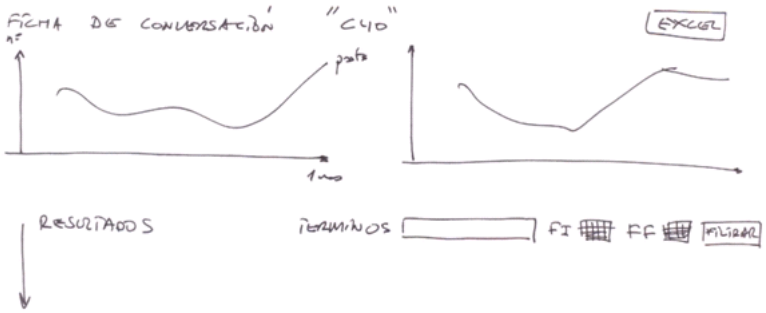


Figura C.9:

Bibliografía

[1] C. de la Figuera Suárez, "Smmart: Social media marketing analysis reporting tool," Master's thesis, Universidad de Zaragoza, 2009.

[2] I. A. B. (IAB-Spain), *Libro Blanco Vol. 8 - Comunicación en Medios Sociales*, 2009. [Online]. Available: http://www.iabspain.net/descargas/descarga.php?id=124

[3] T. O'Reilly, "What is web 2.0: Design patterns and business models for the next generation of software." [Online]. Available: http://papers.ssrn.com/sol3/papers.cfm?abstract_id=1008839

[4] L. Fernández, "Entrevista: David sifry, creador de technorati." [Online]. Available: http://www.elpais.com/articulo/radio/television/blogs/tienen/espacio/enorme/crecer/elpepirtv/20071120elpepirtv_3/Tes

[5] Mangasverdes.es, "Estado de internet a diciembre de 2009," 2009. [Online]. Available: http://mangasverdes.es/2010/01/23/estado-internet-diciembre-2009/

[6] Bitacoras.com, "Informe sobre el estado de la blogosfera hispana bitacoras.com 2010," 2010. [Online]. Available: http://bitacoras.com/informe/10/

[7] *Amazon Elastic Compute Cloud: Developer Guide*. [Online]. Available: http://awsdocs.s3.amazonaws.com/EC2/latest/ec2-gsg.pdf

[8] T. White, *Hadoop: The Definitive Guide*, M. Loukides, Ed. O'Reilly Media, Inc., 2009.

[9] J. Dean and S. Ghemawat, "Mapreduce: Simplified data processing on large clusters," 2004. [Online]. Available: http://static.googleusercontent.com/external_content/untrusted_dlcp/labs.google.com/es//papers/mapreduce-osdi04.pdf

[10] S. Ghemawat, H. Gobioff, and S.-T. Leung, "The google file system," 2003. [Online]. Available: http://static.googleusercontent.com/external_content/untrusted_dlcp/labs.google.com/es//papers/gfs-sosp2003.pdf

[11] *Nutch Wiki*. [Online]. Available: http://wiki.apache.org/nutch/

[12] A. Fraguas, "¿una palabra?: decepción," 2009. [Online]. Available: http://www.elpais.com/articulo/cultura/palabra/decepcion/elpepucul/20091203elpepucul_1/Tes

[13] M. J. Cafarella and O. Etzioni, "A search engine for natural language applications," 2005. [Online]. Available: http://www.eecs.umich.edu/~michjc/papers/be_www2005.pdf

[14] "Language identifier benchs." [Online]. Available: http://wiki.apache.org/nutch/LanguageIdentifierBenchs

[15] *XML Path Language (XPath). Version 1.0*. [Online]. Available: http://www.w3.org/TR/1999/REC-xpath-19991116

[16] D. Smiley and E. Pugh, *Solr 1.4. Enterprise Search Server*. Packt Publishing, 2009.

[17] O. Gospodnetic and E. Hatcher, *Lucene in Action*. Manning, 2005.

[18] J. Kelly and B. Etling, "Mapping iran's online public: Politics and culture in the persian blogosphere," 2008. [Online]. Available: http://cyber.law.harvard.edu/publications/2008/Mapping_Irans_Online_Public

[19] N. Agarwal, M. Galan, and H. Liu, "Clustering blogs with collective wisdom," 2008. [Online]. Available: http://www.public.asu.edu/~huanliu/papers/icwe08.pdf

[20] N. Agarwal, S. Kumar, H. Liu, and M. Woodward, "Blogtrackers: A tool for sociologists to track and analyze blogosphere," vol. 2009. [Online]. Available: http://www.public.asu.edu/~nagarwa6/BlogTrackers-ICWSM09.pdf

[21] R. Agrawal, T. Imielinski, and A. Swami, "Mining association rules between sets of items in large databases." [Online]. Available: http://rakesh.agrawal-family.com/papers/sigmod93assoc.pdf

[22] A. Avendaño and C. Abad, "Sistema de agrupamiento y búsqueda de contenidos de la blogosfera de la espol, utilizando hadoop como plataforma de procesamiento masivo y escalable de datos," 2009. [Online]. Available: http://192.188.59.56/bitstream/123456789/7621/1/Sistema%20de%20Agrupamiento%20y%20b%C3%BAsqueda%20de%20Contenidos%20de%20la%20Blogosfera%20de%20la%20ESPOL.pdf

[23] C. H. Brooks and N. Montanez, "An analysis of the effectiveness of tagging in blogs," 2006. [Online]. Available: http://www.aaai.org/Papers/Symposia/Spring/2006/SS-06-03/SS06-03-002.pdf

[24] T. Couto, C. Ribeiro, and S. Nunes, "Characterizing the portuguese blogosphere," 2009. [Online]. Available: http://www.aaai.org/ocs/index.php/ICWSM/09/paper/viewFile/168/492

[25] P. S. Dodds and C. M. Danforth, "Measuring the happiness of large-scale written expression: Songs, blogs, and presidents," 2009. [Online]. Available: http://www.springerlink.com/content/757723154j4w726k/fulltext.pdf

[26] K. E. Gill, "How can we measure the influence of the blogosphere?" 2004. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.124.2509&rep=rep1&type=pdf

[27] I. H.Witten and E. Frank, *Data Mining: Practical Machine Learning*. Elsevier Inc, 2005.

[28] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Springer, 2008.

[29] S. C. Herring, I. Kouper, J. C. Paolillo, L. A. Scheidt, M. Tyworth, P. Welsch, E. Wright, , and N. Yu, "Conversations in the blogosphere: An analysis "from the bottom up"," 2005. [Online]. Available: http://csdl2.computer.org/comp/proceedings/hicss/2005/2268/04/22680107b.pdf

[30] R. del Hoyo, I. Hupont, F. J. Lacueva, and D. Abadía, "Hybrid text affect sensing system for emotional language analysis."

[31] A. Java, "Tracking influence and opinions in social media," 2006. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.95.775&rep=rep1&type=pdf

[32] A. Java, P. Kolari, T. Finin, and T. Oates, "Modeling the spread of influence on the blogosphere," 2006. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.120.9214&rep=rep1&type=pdf

[33] P. Kolari, T. Finin, and A. Joshi, "Svms for the blogosphere: Blog identification and splog detection," 2006. [Online]. Available: http://www.aaai.org/Papers/Symposia/Spring/2006/SS-06-03/SS06-03-019.pdf

[34] P. Kolari, A. Java, T. Finin, T. Oates, and A. Joshi, "Detecting spam blogs: A machine learning approach," 2006. [Online]. Available: http://www.aaai.org/Papers/AAAI/2006/AAAI06-212.pdf

[35] X. Llorà, N. I. Yasui, and D. E. Goldberg, "Analyzing trends in the blogosphere using human-centered analysis and visualization tools," 2007. [Online]. Available: http://www.icwsm.org/papers/4--Llora-Imafuji-Goldberg.pdf

[36] G. Mishne and N. Glance, "Leave a reply: An analysis of weblog comments," 2006. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.61.5982&rep=rep1&type=pdf

[37] R. M. C. Morales, "Clasificación automática de textos considerando el estilo de redacción," Master's thesis, Instituto Nacional de Astrofísica, Óptica y Electrónica, 2007.

[38] *Hadoop on Windows with Eclipse*. [Online]. Available: http://v-lad.org/Tutorials/Hadoop/00%20-%20Intro.html

[39] *GridGain 2.1 Documentation Center*. [Online]. Available: http://www.gridgainsystems.com/wiki/display/GG15UG/Table+Of+Contents

[40] *Solr Application Development Workshop*. Lucid Imagination, 2010.

[41] *Lucidworks for Solr, Certified distribution reference guide*. Lucid Imagination, 2009.