Sistema de Agrupamiento y Búsqueda de Contenidos de la Blogosfera de la ESPOL, Utilizando Hadoop como Plataforma de Procesamiento Masivo y Escalable de Datos

Allan Avendaño, Cristina Abad. MsC.
Facultad de Ingeniería en Electricidad y Computación
Escuela Superior Politécnica del Litoral
Campus "Gustavo Galindo V.", Km 30.5, Vía Perimetral, Guayaquil, Ecuador
aavendan@espol.edu.ec, cabadr@espol.edu.ec

Resumen

En la actualidad, la proliferación de las redes sociales en la Web permite crear redes de contactos confiables y establecer espacios de colaboración digital. Estos espacios han proporcionado nuevas estrategias para la difusión de los contenidos que se generan en empresas o instituciones educativas.

En este contexto, en la Escuela Superior Politécnica del Litoral se ha adaptado el mecanismo de publicación de información en blogs para la conformación de una red digital compuesta por los miembros de la comunidad politécnica, y que además funcione como un medio de divulgación de los contenidos que la comunidad genere.

En el presente documento se plantea la implementación de un sistema de búsqueda y recomendación de entradas de la blogosfera politécnica, que a más de ser una herramienta para la difusión de los contenidos de la comunidad politécnica, también ayude a mejorar la visibilidad del dominio de ESPOL en la Web.

Palabras Claves: *Hadoop, Map/Reduce, Information Retrieval, Blogs, ESPOL*

Abstract

Nowadays, the proliferation of social networks on the Web enables the creation of networks of contacts and the establishment of reliable and digital collaboration spaces. This spaces provide new strategies for the dissemination of information generated by different institutions.

The Escuela Superior Politécnica del Litoral has adapted the mechanism of blogging for the formation of a digital network consisting of the community members, and also as a means to disseminate information that the community generates.

In this paper, we present an implementation of a system that allows to users to search entries within the ESPOL blogosphere.

1. Introducción

Desde el 2007, la Escuela Superior Politécnica del Litoral se ha planteado el objetivo de mejorar la visibilidad del contenido que se genere en su dominio Web, a fin de posicionarse en los primeros lugares de la lista de sitios Webs de universidades en el Ecuador y a nivel mundial.

Una de las medidas adoptadas para lograr este propósito consistió en crear la blogosfera[1] politécnica, cuya comunidad de autores estaría conformada por estudiantes, profesores y personal relacionado a la ESPOL.

Si bien es cierto, que con el desarrollo de la blogosfera politécnica se ha logrado diversificar los temas con los que se relaciona la ESPOL y así aportar al cumplimiento del objetivo planteado inicialmente; también resulta necesario analizar otros factores implícitos de la comunidad, como la relación de lectura [9], que podrían aportar a la mejora de la visibilidad del contenido de la ESPOL en la Web.

En la actualidad, el directorio de blogs de la ESPOL [1] funciona como punto de entrada a la comunidad al mostrar los blogs recientemente actualizados. No obstante, resulta imposible realizar tareas sencillas como la búsqueda de entradas por términos específicos o que hayan sido publicadas en una fecha determinada.

En este documento se describe el proceso de desarrollo del "Sistema de Agrupamiento y Búsqueda de Contenidos de la Blogosfera de la ESPOL, Utilizando Hadoop como Plataforma de Procesamiento Masivo y Escalable de Datos", diseñado como una herramienta de ayuda para la difusión de los contenidos que se genera en la blogosfera politécnica.

2. Motivación

Según Porter [2], las aplicaciones sociales en la Web se fundamentan en siete características que motivan la participación activa de los miembros en la comunidad.

Dos de estas características, están estrechamente relacionadas al comportamiento de los usuarios, las cuales, son el sentido de eficacia y de pertenencia, con las que se mantiene el esquema colaborativo de las comunidades Web. El sentido de eficacia se refiere a la percepción positiva de las aportaciones de los miembros de la comunidad; y el sentido pertenencia es expresado por el compromiso de los usuarios con la información que se maneja en la comunidad. Debido a estos criterios, cada miembro de la blogosfera desarrolla una reputación, expresada por el número de citaciones en la blogosfera o por la cantidad de comentarios por cada entrada.

Para lograr esta notoriedad, es necesario visibilizar el contenido publicado en los blogs, a través de sitios como Technorati [5], o al permitir el tracking[2] de la información publicada.

Es por esto, que para aplicaciones sociales que involucran la expresión de los usuarios a través de sus publicaciones en la Web, que resulta imprescindible un medio de difusión de los contenidos que se genera en la comunidad, como motores de búsqueda o directorios de contenidos.

En el presente trabajo se considera la implementación de un sistema de búsqueda y recomendación de entradas, como una herramienta para la difusión de los contenidos que se genera en la blogosfera politécnica; y que además, servirá de para mejorar la visibilidad del dominio de ESPOL en la Web.

3. Fundamentación Teórica

3.1. Paradigma Map/Reduce

Desde el año 2000 el equipo de Google, realiza cientos de transacciones computacionales para resolver problemas que comprenden numerosas unidades de procesamiento y grandes conjuntos de datos, por ejemplo: Escaneo de patrones de texto, Conteo de unidades ó Problemas de indexación términos.

Es por esto, que decidieron implementar una plataforma que permita paralelizar tareas y distribuir los datos a procesar en un clúster de máquinas de propósito general. De acuerdo al diseño original, la plataforma debía permitir la planificación, seguimiento y reporte de tareas por cada nodo del clúster; además, debía detectar y recuperarse de las fallas comunes para este tipo de equipos [3].

Map/Reduce es el modelo de programación desarrollado por Google para resolver sus tareas de procesamiento de datos a larga escala, inspirado en las operaciones que implementan lenguajes funcionales como Lisp [7].

---

[1] *Colección de blogs en la Web.*

[2] E*xtracción de información de sitios externos.*

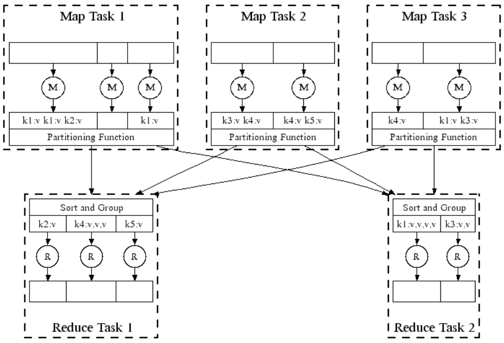

Figura 1. Ejecución de un proceso Map/Reduce [8]

En la Figura 1, se muestra el esquema de ejecución de un proceso Map/Reduce. La función map procesa pares del tipo clave/valor para generar un conjunto intermedio de pares del mismo tipo, los cuales son parcialmente agrupados y ordenados. Esto pares intermedios son procesados con funciones llamadas reduce, las cuales emiten un archivo final con cada claves y sus respectivos valores asociados.

3.2. Hadoop: Plataforma de procesamiento masivo de datos

Hadoop [4] es un proyecto de Apache Software Foundation, que provee una plataforma para el procesamiento en distribuido y masivo de datos en computadores de propósito general, basada en el estilo de programación Map/Reduce desarrollado por Google.

Esta plataforma, se presenta como una solución de código abierto para los programadores sin experiencia en desarrollo de aplicaciones para ambientes distribuidos, ya que oculta la implementación de detalles propios de estos sistemas: paralelismo de tareas, tolerancia a fallos, administración de procesos y balanceo de carga [3][10].

3.3. Recuperación de Información (Information Retrieval)

En la era de la información, ha aumentado la generación de documentos y, de igual manera, la capacidad de almacenamiento de los dispositivos electrónicos, aunque sigue siendo mínima la extracción de información contextual de los documentos.

El proceso de extracción de información consiste en representar, almacenar, organizar y acceder a documentos relevantes tomados a partir de una colección de documentos sin estructurar (generalmente en lenguaje natural), con el objetivo de satisfacer las necesidades de los usuarios [11].

Mediante estos sistemas de extracción de información permite a los usuarios obtener una visión más detallada de las características que posee una colección de documentos, sin realizar un análisis minucioso.

4. Metodología de Desarrollo

4.1. Diseño

El sistema planteado en el presente trabajo está compuesto por los siguientes componentes: un módulo de agrupamiento de contenidos y un módulo de indexación y búsqueda por términos.

El módulo de agrupamiento de contenidos es el encargado de procesar y agrupar las entradas de blogs de acuerdo a la similitud en su contenido.

El módulo de indexación y búsqueda por términos funciona como un motor de búsquedas sobre las entradas extraídas de la blogosfera politécnica a partir de los términos recibidos desde la interfaz Web; además, provee los blogs relacionados a los resultados de las búsquedas. Estas recomendaciones de entradas son obtenidas a partir de los resultados del módulo de agrupamiento de contenidos.

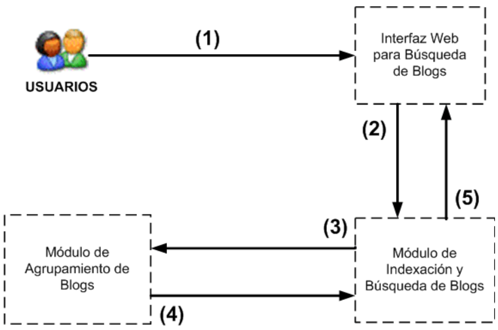

Figura 2. Diseño modular del sistema

En la Figura 2, se describe la interacción de los módulos del sistema: Primero, los usuarios ingresan los términos de búsqueda desde la interfaz Web del sistema (1), dichos términos son enviados al módulo de indexación y búsqueda para obtener los blogs relevantes a dichos términos (2).

Luego, a partir de las entradas resultantes de la búsqueda (3) se obtienen los blogs relacionados (4). Finalmente, el resultado es procesado y visualizado en la interfaz Web (5).

4.2. Implementación

4.2.3. Módulo de agrupamiento de contenidos.

El contenido de las entradas es agrupado de acuerdo a su similaridad, mediante siete fases Map/Reduce que implementan ciertas técnicas de extracción de información.

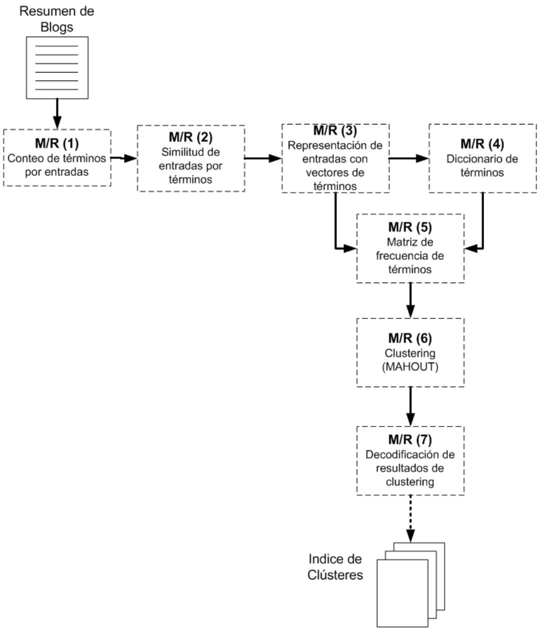

Figura 3. Diseño del Módulo de Agrupamiento de contenidos

Tal como se indica en la Figura 3, se describen cada uno de las fases secuencialmente ejecutadas, que componen este módulo:

1. **Conteo de términos.** Es una adaptación de un proceso Map/Reduce de la clase de CS/Info4300 de la Universidad de Cornell [11][12].
   En esta fase, las palabras del corpus son convertidas en términos. Primero, las palabras son extraídas del texto, por la separación existente entre ellas, sea por un espacio simple, signos de puntuación y/o caracteres especiales. Luego, las palabras más comunes son eliminadas (tanto del idioma español, como del corpus), y finalmente, se extraen las raíces de las palabras comunes. El resultado de esta fase, está compuesto por cada término del corpus relacionado con el identificador de cada blog que contiene dicho término.

2. **Similitud de entradas por términos.** Para asegurar la confiabilidad de las recomendaciones del sistema, en esta fase se realiza un agrupamiento simple de entradas. El resultado de esta fase son pequeños grupos, conformados por entradas asociadas de acuerdo a la similaridad de los términos que se encuentren en el contenido [13].

3. **Representación de entradas con vectores de términos.** En esta fase se analizan cada uno los grupos de entradas de la fase anterior. Cada entrada es representada como un vector de términos con su respectivo factor de de relevancia en el corpus (TF-IDF).

4. **Diccionario de términos.** En esta fase se extraen los términos que componen el léxico del corpus.

5. **Matriz de frecuencia de términos.** Aquí se construye una matriz de frecuencia términos de *m* filas (entradas en el grupo) y *n* columnas (términos en el léxico), cuya intersección corresponde al factor de relevancia del término en la entrada.

6. **Clustering (MAHOUT).** En esta fase se utiliza el algoritmo de agrupamiento, K-means, implementado en la librería Mahout [14] bajo el paradigma Map/Reduce.

7. **Decodificación de resultados del clustering.** En esta fase se decodifican e indexan los resultados del agrupamiento previamente realizado.

4.2.3. Módulo de indexación y búsqueda por términos.

Principalmente, está compuesto por un índice invertido de entradas extraídas de la blogosfera politécnica entre los meses de Febrero y Marzo del 2009.

Este índice es utilizado para realizar las búsquedas de entradas que contengan los términos ingresados desde la interfaz Web del sistema.

Luego, con el resultado de la búsqueda, se obtienen las entradas relacionadas a partir del índice obtenido del módulo de agrupamiento.

Finalmente, las entradas resultantes son procesadas por el API de la librería Carrot[2] [15], con la que se producen grupos de entradas generadas en línea.

4.3. Pruebas del Sistema

Durante la implementación del sistema, se realizaron pruebas de rendimiento a las fases Map/Reduce, que por el volumen de datos procesar requieren de mayor tiempo y recursos.

Para la realización de estas pruebas, se utilizaron los recursos bajo demanda de Amazon con los Amazon Web Services, compuestos por diez nodos configurados con la plataforma de Hadoop, instalados con la distribución Fedora de Linux.

Las pruebas se realizaron desde el proceso Map/Reduce de obtención de entradas similares por términos (2) hasta el procesamiento de Clustering-MAHOUT (6) del Módulo de Agrupamiento de Contenidos, descritos previamente.

En la Figura 4, se muestra que existe una relación lineal entre los grupos de blogs más representativos y el tiempo de procesamiento. Esto muestra la presencia de escalabilidad lineal en la plataforma Hadoop, tal y como ha sido reportado por sus desarrolladores.

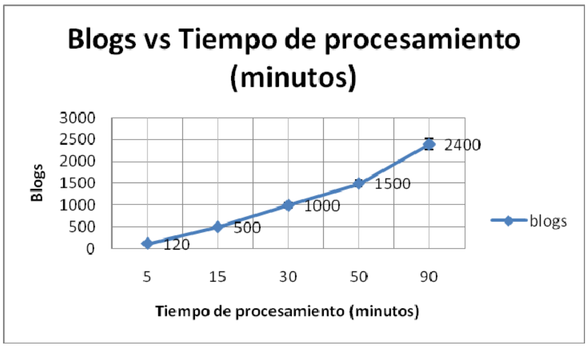

Figura 4. Gráfico comparativo de grupos de blogs analizados versus tiempo de procesamiento (minutos)

5. Conclusiones

Al finalizar el proceso de desarrollo del sistema descrito en el presente documento, se logró identificar que debido a que el paradigma de procesamiento masivo de datos Map/Reduce, está diseñado para procesar datos con una estructura definida, resulta fácil implementar tareas que permitan extraer información relevante de un conjunto de documentos escritos en lenguaje natural.

Mediante la visualización de grupos etiquetados de entradas de blogs, es posible realizar inspecciones rápidas de los resultados de búsquedas.

6. Trabajo Futuro

Debido a que a que el alcance del presente estudio se limitó al desarrollo del sistema de búsqueda y recomendaciones de entradas, es aconsejable que se realice un estudio comparativo en el que se analice el aporte del sistema desarrollado a la visibilidad del dominio de ESPOL. Además, es aconsejable que para futuras investigaciones se analice el impacto del módulo de recomendaciones en la comunidad de autores de blogs politécnicos para la conformación de redes de confiabilidad.

7. Agradecimientos

De manera especial, agradecemos por la colaboración del Centro de Servicios Informáticos de la ESPOL por permitirnos realizar parte del desarrollo del sistema, descrito en el presente trabajo.

8. Referencias

[1] BLOG DE ESPOL DIRECTORIO, Julio 2009, Disponible en http://blog.espol.edu.ec/directorio/.

[2] PORTER, JOSUA, Designing for the Social Web, NEW RIDERS, 2008, p. 97.

[3] DEAN, JEFF Y GHEMAWAT, SANJAY, Map/Reduce: Simplified Data Processing on Large Clusters, en Sixth Symposium on Operating System Design and Implementation (OSDI'04), San Francisco CA, Diciembre 2004.

[4] HADOOP, Julio 2009, Disponible en http://hadoop.apache.org/.

[5] TECHNORATI. Julio 2009, Disponible en http://www.technorati.com.

[6] PAUL GRAHAM, Julio 2009, Disponible en http://www.paulgraham.com/lisp.html.

[7] WEISS DAWID. Massive Distributed Processing using Map-Reduce, Institute of Computing Science, Pozna University of Technology. Enero 2007.

[8] FURUKAWA T., ISHIZUKA M., MATSUO Y., OHMUKAI I. Y UCHIYAMA K, Analyzing Reading Behavior by Blog Mining. Association for the Advancement of Artificial Intelligence. 2007.

[9] VENNER JASON. Pro Hadoop, APRESS, 2009, 4-6 pp.

[10] MANNING CHRISTOPHER, RAGHAVAN PRABHAKAR, SCHÜTZE HINRICH,

Introduction to Information Retrieval, Cambridge University Press, 2008, 1-10 pp.

[11] WILLIAM Y. ARMS Y BLAZEJ J. KOT, Indexer, Octubre 2008, http://www.infosci.cornell.edu/courses/info4300/2008fa/Indexer.txt.

[12] WILLIAM Y. ARMS AND BLAZEJ J. KOT, WordinDoc, Octubre 2008, http://www.coursehero.com/file/1550104/WordinDoc/.

[13] ELSAYED T., LIN J., DOUGLAS W., Pairwise Document Similarity in Large Collections with Map/Reduce, en Proceedings of ACL-08: HLT, Short Papers (Companion Volume), Columbus, Ohio, USA, Junio 2008, 265-268 pp.

[14] MAHOUT, Apache Mahout – Overview, Septiembre 2009, http://lucene.apache.org/mahout/.

[15] CARROT^2 – OPEN SOURCE SEARCH RESULTS CLUSTERING ENGINE, Carrot^2, Septiembre 2009, http://project.carrot2.org/.