Gestión de Información en la Web
Master en Ingeniería Informática

Práctica 3: Desarrollo de un Sistema de Recuperación de Información

Luis Alberto Segura Delgado
DNI: 45922174-Y
segura2010@correo.ugr.es

Miércoles 4 de Mayo de 2016

Índice

1 Introducción

El objetivo de esta práctica es conocer las distintas partes que forman un sistema de recuperación de información y la funcionalidad asociada a cada una de ellas. Para ello, se propone desarrollar un sistema de recuperación de información utilizando una de las bibliotecas más utilizadas, **Lucene**[1].

De esta forma, aprenderemos como implementar un sistema de recuperación de información de forma rápida y sencilla gracias a la utilización de la biblioteca Lucene. Así, si en algún momento necesitamos desarrollar un sistema de recuperación de información conoceremos las herramientas que hay disponibles, como funcionan y como usarlas.

2 Trabajo Realizado

Se ha implementado una aplicación sencilla en lenguaje **Java**, que utilizando la biblioteca de Lucene, permite indexar y buscar los documentos de la agencia EFE proporcionados. El software desarrollado es una única aplicación que dispone de varias pestañas, una para indexación, otra para seleccionar ciertos parámetros y otra con una breve indicación sobre como buscar documentos. A continuación veremos cada una de ellas y brevemente la parte técnica detrás del sistema.

2.1 Manual de Usuario

2.1.1 Pestaña Opciones

Es la pestaña más importante y la que siempre debemos configurar al ejecutar la aplicación. En esta sencilla pestaña debemos indicar el directorio en el que se encuentra el indice que hayamos creado anteriormente o en el que queramos que se cree un nuevo índice.

También debemos seleccionar en ella el fichero de palabras vacías que corresponda. Si no seleccionamos alguna o ninguna de estás dos opciones, ni el buscador ni el indexador funcionarán.

[1] `http://lucene.apache.org/`

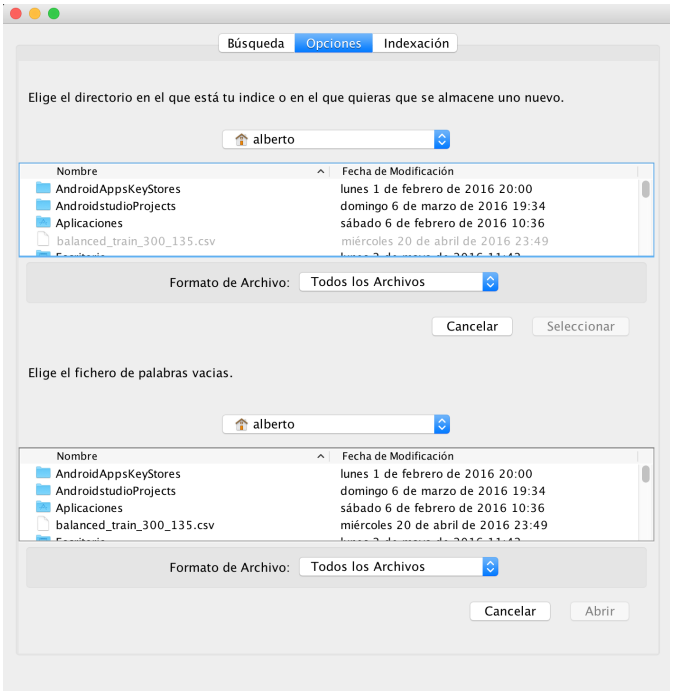

Figura 1: Pestaña de Opciones

2.1.2 Pestaña Indexación

En esta pestaña, el usuario podrá seleccionar el documento XML de la agencia EFE que desee indexar. Una vez seleccionado, se avisará con un mensaje cuando los documentos asociados a dicho fichero XML se hayan indexado.

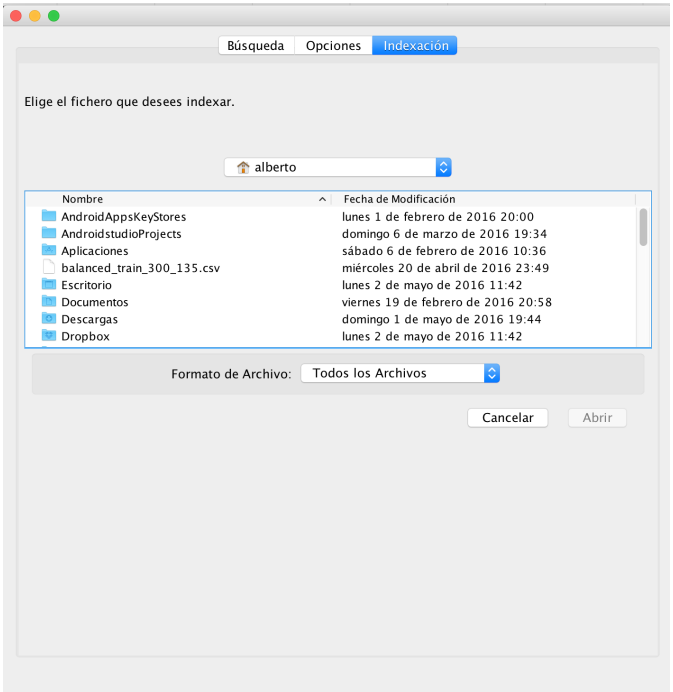

Figura 2: Pestaña de Indexación

2.1.3 Búsqueda

En la pestaña de búsqueda se simplemente se incluye un mensaje que nos indica que debemos acceder a la dirección `http://localhost:8000/` con nuestro navegador Web para poder buscar los documentos que deseemos.

Figura 3: Web para la búsqueda

Como podemos ver en la figura 3, es posible buscar cadenas de texto en el título o en el contenido de la noticia. Y al buscar, nos aparecerá el listado de noticias recuperadas por el sistema.

2.1.4 Como ejecutar

Para ejecutar la aplicación es necesario ejecutar el ejecutable "GIW-P3.jar". Hay que asegurarse de que junto al ejecutable se encuentra la carpeta "lib" que incluye las bibliotecas necesarias de Lucene, y la carpeta "public_html" que contiene el fichero "index.html", que contiene el código HTML y JavaScript para la parte de la búsqueda en la web.

2.2 Desarrollo

2.2.1 Indexación

Para la indexación, se ha creado una clase llamada "Indexer", que realiza la indexación a través del método "index". Este método debe recibir como parámetros:

- El fichero XML de documentos EFE
- El fichero de palabras vacías
- El directorio en el que está o se creará el índice

De forma que leerá el fichero XML, obteniendo el título y texto para indexarlo con Lucene. El código de indexación, como se puede ver en el código fuente, es muy similar al proporcionado en el guión, pero con la diferencia principal de usar específicamente el "Spanish Analyzer". Este analizador nos permite indexar los documentos realizando stemming con el "Stemmer" por defecto.

2.2.2 Búsqueda

Para la búsqueda de los documentos previamente indexados, como hemos visto en el manual de usuario, se ha desarrollado una aplicación web sencilla que nos permite realizar búsquedas sobre nuestro índice, ya sea por título o por contenido.

Para hacer funcionar la web, se ha utilizado la clase "Webserver" de Java, de forma que muestra el fichero HTML correspondiente cuando se accede a la URL de la web. Y para hacer la búsqueda se han utilizado peticiones HTTP GET (en la dirección http://localhost:8000/api/search), realizadas con Angular[2]. De esta forma, lo que se ha hecho ha sido montar una API sencilla para recuperar los resultados de la búsqueda en formato JSON y después mostrarlos en la web.

Para realizar la búsqueda con Lucene, se ha creado la clase "Searcher", que mediante los métodos "searchByTitle" y "searchByText", se encargan de preparar la llamada final a la función "search" que es la que contiene todas las llamadas necesarias a la librería Lucene. El código es similar al código de ejemplo que se proporciona en el guión, pero al igual que con el "Indexer" se utiliza el Analyzer español y la lista de palabras vacías para realizar la búsqueda sobre el índice.

Entre la clase que se encarga de gestionar los eventos de las peticiones HTTP para la API de búsqueda y la clase "Searcher", se ha desarrollado una clase Singleton (SearcherController) que contiene un objeto Searcher para realizar las búsquedas. De esta forma evitamos tener varios objetos searcher innecesarios consumiendo recursos.

[2] `https://angularjs.org`

Notas Finales sobre la Entrega

Junto a esta documentación se incluye el código fuente Java y en la carpeta "dist" del proyecto se encuentra el ejecutable de la aplicación, las librerías en la carpeta "lib" y la parte web en la carpeta "public_html". Para probar la aplicación se recomienda usar el contenido de la carpeta "dist", pero si se desea compilar el proyecto, será necesario incluir la carpeta "public_html" junto al ejecutable generado. En caso de no hacerlo, no funcionará la web, y por tanto, no se podrán realizar búsquedas.