



ugr

Universidad
de Granada

GESTIÓN DE INFORMACIÓN EN LA WEB
MÁSTER PROFESIONAL EN INGENIERÍA INFORMÁTICA

Desarrollo de un Sistema de Recuperación de Información con Lucene

Autor

Ernesto Serrano Collado

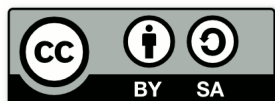
Profesor

Juan Manuel Fernández Luna



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍAS INFORMÁTICA Y DE
TELECOMUNICACIÓN

Granada, 2 de marzo de 2018



Desarrollo de un Sistema de Recuperación de Información con Lucene

Ernesto Serrano Collado

Resumen

Palabras clave: *software libre, recuperación información, lucene*

Los objetivos de esta práctica son:

1. Conocer las partes principales que tiene un sistema de recuperación de información y qué funcionalidad tiene cada una.
2. Implementar un sistema de recuperación de información.
3. Emplear la biblioteca **Lucene** para facilitar dicha implementación.

Yo, **Ernesto Serrano Collado**, alumno de la titulación **Máster Profesional en Ingeniería Informática** de la **Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación de la Universidad de Granada**, autorizo la ubicación de la siguiente copia de mi Trabajo (*Desarrollo de un Sistema de Recuperación de Información con Lucene*) en la biblioteca del centro para que pueda ser consultada por las personas que lo deseen.

Además, este mismo trabajo está publicado bajo la licencia **Creative Commons Attribution-ShareAlike 4.0**, dando permiso para copiarlo y redistribuirlo en cualquier medio o formato, también de adaptarlo de la forma que se quiera, pero todo esto siempre y cuando se reconozca la autoría y se distribuya con la misma licencia que el trabajo original. El documento en formato **LaTeX** se puede encontrar en el siguiente repositorio de **GitHub**: https://github.com/erseco/ugr_gestion_informacion_web/tree/master/p3/.

Fdo: Ernesto Serrano Collado

Granada, a 2 de marzo de 2018

Índice general

1. Introducción	1
2. Implementación	3
2.0.1. Instalación de PyLucene	3
2.0.2. Desarrollo de los programas requeridos	4
3. Manual	7
4. Bibliografía	9

Capítulo 1

Introducción

En esta práctica se construirá un sistema de recuperación de información empleando la biblioteca Lucene, compuesto de dos programas:

1. Un **indexador**, el cual recibirá como argumentos la ruta de la colección documental a indexar, el fichero de palabras vacías a emplear y la ruta donde alojar los índices, y llevará a cabo la indexación, creando los índices oportunos y ficheros auxiliares necesarios para la recuperación. Esta aplicación se ejecutará en la línea de mandatos y no tendrá ningún componente gráfico. Este software realizará las tareas de tokenización, eliminación de palabras vacías y extracción de raíces antes de crear el índice.
2. Un **motor de búsqueda**, que al ejecutarse recibirá como argumento la ruta donde está alojado el índice de la colección y permitirá que un usuario realice una consulta de texto y obtenga el conjunto de documentos relevantes a dicha consulta. En este caso, el programa sí será gráfico. Sobre la consulta se realizarán los mismos procesos que sobre los documentos en el indexador.

Capítulo 2

Implementación

Para la realización de esta práctica se ha optado por utilizar el lenguaje de programación **Python** mediante la librería **PyLucene**. Dicha librería lo que hace es mediante **jcc** crear un envoltorio (*wrapper*) de **Lucene** que puede ser invocado desde **Python**. La librería **PyLucene** no es un desarrollo tan activo y con tanta documentación como el propio **Lucene** por lo que para usar muchas cosas se ha tenido que indagar en el propio código fuente y a base de prueba y error.

2.0.1. Instalación de PyLucene

Para el despliegue se ha provisionado una máquina Ubuntu 16.04 LTS en **AWS**. Para la compilación de *PyLucene* ha sido necesario una máquina bastante potente con al menos 2GB de memoria RAM, pero una vez compilado *PyLucene* se ha podido cambiar el tipo de instancia a una t2.micro con solo 0.5GB de RAM.

Instalación de dependencias

```
1 sudo apt-get update
2 sudo apt-get install -y ant g++ python-dev python-setuptools python-
  pip default-jdk-headless
3 sudo pip install --upgrade pip
4 sudo pip install setuptools --upgrade
```

Descarga y extracción de ‘PyLucene’

```
1 wget http://apache.rediris.es/lucene/pylucene/pylucene-4.10.1-1-src.
  tar.gz
2 tar -zxvf pylucene-4.10.1-1-src.tar.gz
```

Compilación e instalación de ‘jcc’

```
1 cd pylucene-4.10.1-1/jcc
2 sed -i s/java-7-openjdk-amd64/java-8-openjdk-amd64/ setup.py
3 python setup.py build
4 sudo python setup.py install
5 cd ..
```

Edición del ‘Makefile’ de ‘PyLucene’

```
1 # Linux      (Ubuntu 11.10 64-bit, Python 2.7.2, OpenJDK 1.7,
   setuptools 0.6.16)
2 # Be sure to also set JDK['linux2'] in jcc's setup.py to the JAVA_HOME
   value
3 # used below for ANT (and rebuild jcc after changing it).
4 PREFIX_PYTHON=/usr
5 ANT=JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64 /usr/bin/ant
6 PYTHON=$(PREFIX_PYTHON)/bin/python
7 JCC=$(PYTHON) -m jcc --shared
8 NUM_FILES=8
```

Compilación e instalación de ‘PyLucene’

```
1 make
2 sudo make install
```

2.0.2. Desarrollo de los programas requeridos

Indexador

Se ha desarrollado un indexador básico utilizando la clase `SpanishAnalyzer`, el mayor problema encontrado a la hora de utilizar dicha clase ha sido para indicarle la lista de palabras a ignorar, ya que esperaba un tipo de dato especial, tras muchas pruebas se ha resuelto utilizando la clase `CharArraySet`.

Para parsear los ficheros XML se ha utilizado la librería `etree` The ElementTree XML API haciendo uso de XPath para facilitar el extraer elementos del árbol DOM.

Se ha mejorado el ejemplo proporcionado agregando más elementos de los XML como pueden ser: **tipo sesion**, **organo presidente**, **fecha**, y **tipo epigrafe**.

Motor de búsqueda

Se ha desarrollado un motor de búsqueda que mediante el micro framework `Flask` expone una web que nos permite realizar búsquedas de una

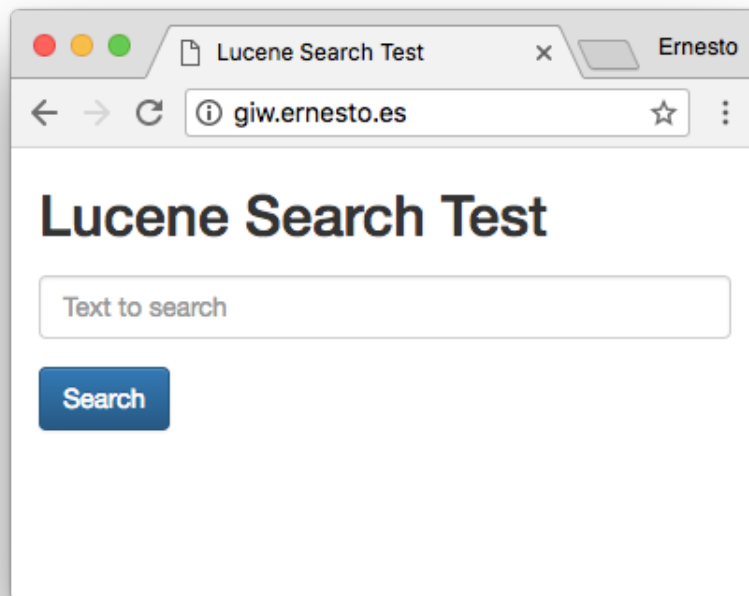


Figura 2.1: Interfaz

forma sencilla e intuitiva como se puede ver en las siguientes capturas.

Se ha utilizado **Bootstrap** para darle un mejor aspecto visual, además se ha agregado una funcionalidad para resaltar en el texto la cadena buscada.

En el caso de que se devuelvan muchos resultado solo se muestran los 10 primeros para evitar sobrecargar al servidor.

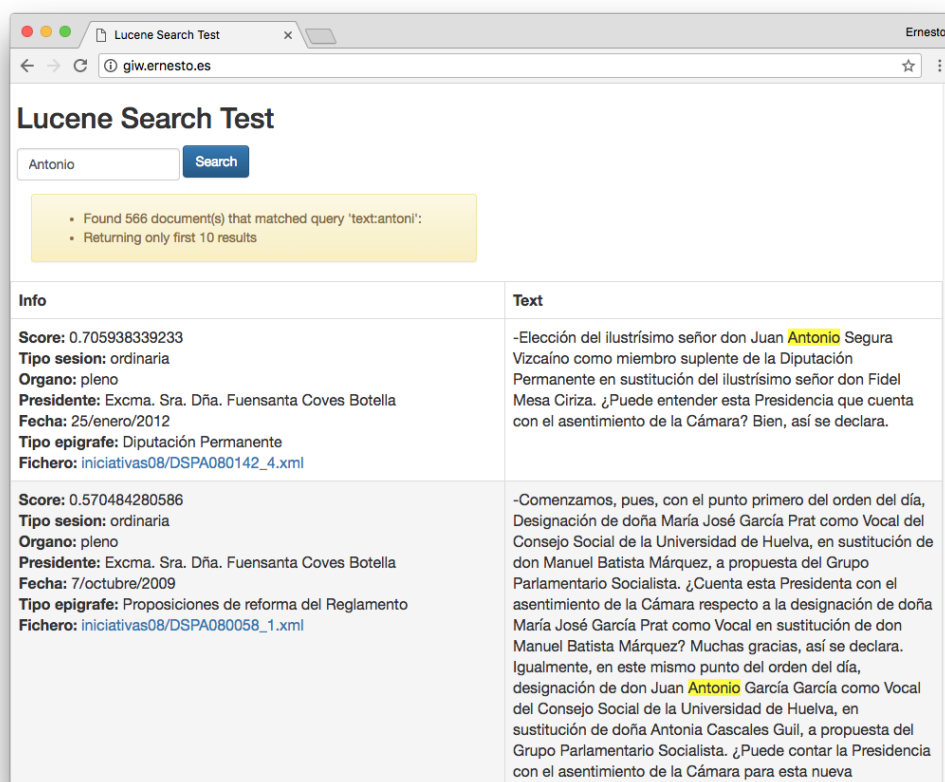


Figura 2.2: Resultados

Capítulo 3

Manual

Indexador

Para realizar la indexación hay que ejecutar el programa `index.py` con los parámetros que exigen los requisitos de la práctica, se adjunta un ejemplo de como se debería de lanzar:

```
1 ./indexer.py iniciativas08 palabras_vacias_utf8.txt index
```

Si no se le especifican los parámetros requeridos el programa mostrará un mensaje de ayuda.

Motor de Búsqueda

Para lanzar el motor de búsqueda hay que ejecutar el programa `search.py` con los parámetros que exigen los requisitos de la práctica, se adjunta un ejemplo de como se debería de lanzar:

```
1 ./search.py index
```

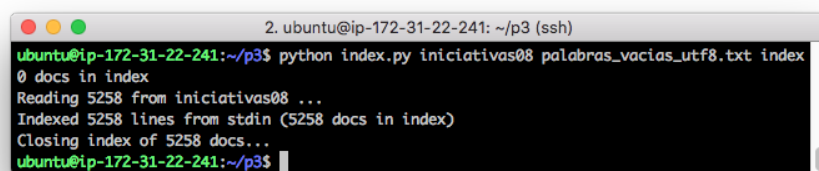
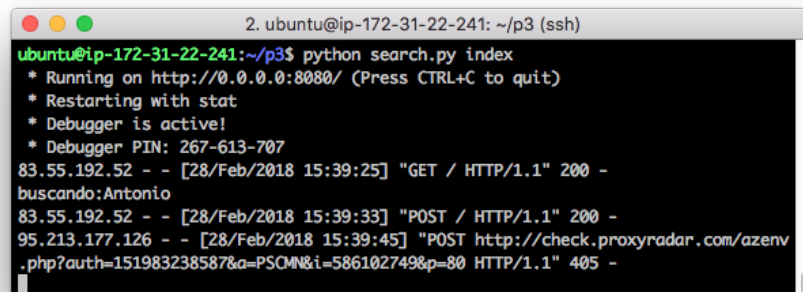


Figura 3.1: Indexador

A terminal window titled '2. ubuntu@ip-172-31-22-241: ~/p3 (ssh)' displays the output of running 'python search.py index'. The script starts by announcing it's running on 'http://0.0.0.0:8080/' and is restarted with 'stat'. It then shows a debugger is active with PIN '267-613-707'. Log entries show a GET request from 83.55.192.52 at 15:39:25 returning 200, followed by a POST request from the same IP at 15:39:33 also returning 200. A final log entry shows a POST request from 95.213.177.126 at 15:39:45 to 'http://check.proxymadar.com/azenv.php?auth=151983238587&a=PSOMN&i=586102749&p=80' returning 405.

```
2. ubuntu@ip-172-31-22-241: ~/p3 (ssh)
ubuntu@ip-172-31-22-241:~/p3$ python search.py index
* Running on http://0.0.0.0:8080/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger PIN: 267-613-707
83.55.192.52 - - [28/Feb/2018 15:39:25] "GET / HTTP/1.1" 200 -
buscando:Antonio
83.55.192.52 - - [28/Feb/2018 15:39:33] "POST / HTTP/1.1" 200 -
95.213.177.126 - - [28/Feb/2018 15:39:45] "POST http://check.proxymadar.com/azenv
.php?auth=151983238587&a=PSOMN&i=586102749&p=80 HTTP/1.1" 405 -
```

Figura 3.2: Motor de búsqueda

Si no se le especifican los parámetros requeridos el programa mostrará un mensaje de ayuda.

Una vez lanzado el programa podremos interactuar con el buscador entrando con el navegador en la url `http://localhost:8080`

Capítulo 4

Bibliografía

- Documentación de Lucene: <http://lucene.apache.org>
- Documentación de PyLucene: <http://lucene.apache.org/pylucene>
- Documentación de XML Etree: <https://docs.python.org/2/library/xml.etree.elementtree.html>
- StackOverflow: <https://stackoverflow.com>
- Creative Commons Share Alike 4.0: <https://creativecommons.org/licenses/by-sa/4.0/>

