FACULTAD DE CIENCIAS DE LA DOCUMENTACIÓN

GRADO EN INFORMACIÓN Y DOCUMENTACIÓN

SISTEMAS DE RECUPERACIÓN DE INFORMACIÓN DE CÓDIGO ABIERTO: SOLR, INDRI, TERRIER

CUADERNO DE TRABAJO

Nº 19

Profesor

Juan Antonio Martínez Comeche

Colección “Cuadernos de Trabajo”, n º 19

Grado en Información y Documentación

Coordinador de la Titulación: Fátima Martín Escudero

Coordinador de la Colección: José Luis Gonzalo Sánchez-Molero

Marzo 2018

ISBN-13: 978-84-697-9855-3

Depósito Legal: M - 5792 - 2018

Edita: Facultad de Ciencias de la Documentación

Universidad Complutense

C/ Santísima Trinidad, 37

28010 Madrid

Contenido

INTRODUCCIÓN

El presente cuaderno de trabajo tiene como principal objetivo completar el conocimiento de los modelos de Recuperación de la Información abordados en la asignatura 'Búsqueda y Recuperación de Información', realizando un primer acercamiento a algunos de los programas de código abierto que desarrollan Sistemas de Recuperación de Información. Esta presentación inicial de los motores de búsqueda no pretende ahondar en las capacidades de administración y gestión de las colecciones o profundizar en las diversas posibilidades de recuperación que poseen, sino dar una visión general de sus características fundamentales. El análisis preliminar de su modo de funcionamiento que se encontrará en estas páginas permitirá al alumno observar cómo se materializan los algoritmos y procedimientos habituales de la Recuperación de Información en tres sistemas manejados cotidianamente en empresas e instituciones de muy diversa índole.

Así pues, los alumnos de la asignatura 'Búsqueda y Recuperación de Información', asignatura obligatoria del Grado en Información y Documentación, podrán en estas páginas comprobar cómo se articulan en la práctica los conceptos analizados previamente en el aula y, a la vez, tener un primer contacto con sistemas reales de Recuperación de Información. Con ello adquieren una formación de carácter tecnológico más próxima a un posible desarrollo profesional futuro, aspectos demandados cada vez en mayor medida tanto académica como profesionalmente.

Para la elaboración de este cuaderno se ha optado por los motores de búsqueda Solr, Indri y Terrier, en cuya selección se han tenido en cuenta las siguientes características:

- Un alto nivel de utilización, tanto en empresas como en organismos públicos e instituciones académicas. Solr, por ejemplo, es un sistema de recuperación que permite integrar diversas fuentes de información empleadas a nivel empresarial (intranet, bases de datos, documentos locales, correos electrónicos…), siendo utilizado también en organismos públicos como la Biblioteca Nacional de España. Por su parte, los sistemas Indri y Terrier son ampliamente empleados en proyectos académicos vinculados a la enseñanza y la investigación en Recuperación de Información.

- Son programas de código abierto, lo que permite no solamente acceder al código para conocer con exactitud cómo se han implementado los distintos procesos involucrados en el programa, sino que además son gratuitos y, por tanto, de mayor accesibilidad que los programas comerciales.

- Son programas multiplataforma, estando disponible una versión para sistema operativo Windows, una versión para sistema operativo Linux/Unix y una versión para sistema operativo Mac OS X. Esta característica garantiza la posibilidad de instalación y uso en el futuro, sea cual sea la plataforma utilizada en la institución u organismo donde los alumnos puedan desarrollar su labor profesional. Dado que en la Facultad de Ciencias de la Documentación se emplea el sistema operativo Windows, en este cuaderno nos centraremos en dicha plataforma. En la documentación de los respectivos sistemas encontrará el lector información relativa a otros sistemas operativos.

- Existencia de documentación complementaria y otras fuentes de información (wikis, foros, sitios web, tutoriales) que permiten ahondar en la gestión y administración de estos sistemas y resolver dudas en caso necesario.

El cuaderno se organiza mostrando, para cada uno de los tres sistemas abordados, al menos los siguientes aspectos generales:

- Los prerrequisitos para su funcionamiento en Windows
- La instalación del programa en Windows
- La apertura y cierre del programa en Windows
- Indexación de una colección
- Eliminación de palabras vacías
- Stemming o reducción morfológica
- Recuperación de documentos

Tras la exposición de cada uno de estos aspectos el cuaderno incluye, intercalados a lo largo del texto relativo a los distintos sistemas de recuperación, varios ejercicios con su solución, de manera que el alumno pueda ir comprobando si ha llevado a cabo correctamente las diversas acciones básicas afrontadas con cada sistema. Con ello cumplimos una doble finalidad: por una parte, si incita al alumno a instalar y probar estos sistemas de recuperación, pues puede comprobar paso a paso si ha realizado correctamente las sucesivas acciones básicas abordadas en este manual; en segundo lugar, le permite evaluar progresivamente su nivel de conocimiento y pericia en el manejo y gestión de cada uno de los sistemas.

Finalmente, en apéndices, se incluyen los pasos que deben seguirse para realizar tareas básicas frecuentes, de carácter informático, necesarias para la utilización de estos programas, además del modo de resolver los principales errores que pueden surgir durante su instalación y empleo.

SOLR

Prerrequisitos de Solr en Windows

Solr es un programa de código abierto de búsqueda y recuperación de información, basado en la librería Lucene.

Solr exige para su funcionamiento la instalación previa de Java (JRE) versión 1.8 o superior en nuestro sistema. Por tanto, debemos asegurarnos de tener instalado Java en nuestro ordenador antes de instalar el programa Solr. Para comprobar si Java (JRE) está instalado en Windows, hacemos clic en 'Inicio' (el icono de Windows en el margen inferior izquierdo de la pantalla), luego en 'Panel de control' y posteriormente en 'Programas':

Inicio > Panel de control > Programas

Si no apareciese el programa 'Java' en 'Inicio > Panel de control > Programas', podemos hacer otra comprobación previa en la siguiente ruta:

Inicio > Panel de control > Programas > Programas y características

Allí debe incluirse una entrada del tipo 'Java 8 Update 131 (64-bit)'. Esta entrada, por ejemplo, corrobora que en nuestro ordenador está instalada la versión '1.8.131' del programa Java. Si no disponemos en nuestro ordenador del programa Java (JRE), debemos instalarlo (ir al apéndice'Instalación de Java (JRE) en Windows', más adelante en este mismo documento).

No solamente es necesario tener instalado el programa Java (JRE). De igual forma, debemos asegurarnos de que la versión instalada en nuestro sistema es la 1.8 o superior (como en este caso, la versión 1.8.131).

Si ya disponemos del programa, en 'Programas' debería aparecer 'Java' junto a su icono (una taza de café). Si es así, comprobaremos la versión de Java instalada. Para ello, hacemos clic en 'Java'. En la pantalla que aparece, hacemos clic en la pestaña 'General' y a continuación hacemos clic en el botón 'Acerca de'. En la pantalla resultante

figurará un mensaje del tipo: ‘Versión 8 Actualización 131’. Ello indica que tenemos instalada la versión ‘1.8.131’. Debemos asegurarnos de que la versión instalada en nuestro sistema es la 1.8 o superior (como en este caso, la versión 1.8.131). Si no disponemos en nuestro ordenador de una versión 1.8 o superior, debemos instalarla (ir al apéndice ‘Instalación de Java (JRE) en Windows’, más adelante en este mismo documento).

También podemos comprobar qué versión de Java tenemos instalada en nuestro ordenador abriendo una CONSOLA, TERMINAL O LÍNEA DE COMANDOS. Para ello, basta ir a ‘Inicio’ (el icono en el margen inferior izquierdo de la pantalla con la bandera de Windows), y en el buscador que aparece en la parte inferior de la pantalla con la frase ‘Buscar programas y archivos’ introducimos ‘cmd’ [sin las comillas simples; solamente las letras cmd]. Aparecerá una pantalla con fondo negro y una última línea semejante a:

C:\Users\Juan>_

Hasta el signo > el sistema nos está indicando en qué subdirectorio nos encontramos (en este caso, en el subdirectorio ‘Juan’, dentro del directorio ‘Users’, dentro de la unidad ‘C’ de nuestro ordenador). El prompt (_) está parpadeando en espera de que introduzcamos un comando (de donde la abreviatura ‘cmd’ introducida anteriormente). Hemos abierto, pues, una CONSOLA, TERMINAL O LÍNEA DE COMANDOS, que nos permite introducir directamente comandos al sistema para su ejecución. Si introducimos el siguiente comando:

java -version

El sistema nos devolverá un mensaje como el siguiente, siempre que tengamos instalado Java:

java version “1.8.0_131”

Java <TM> SE Runtime Environment <build 1.8.0_131-b01>

Java HotSpot <TM> 64-Bit Server VM <build 25.144-b01, mixed mode>

Si no está instalado el programa Java (JRE) o la versión instalada es inferior a la 1.8, procederemos a instalar Java (JRE) antes de proceder a instalar el programa Solr, siguiendo para ello los pasos que se indican en el apartado 'Instalación de Java (JRE) en Windows' dentro de los Apéndices.

Instalación del programa Solr en Windows

En el momento de redactar este taller, la última versión del programa Solr (en nuestro caso, la 7.2.1) puede descargarse desde la página

http://www.apache.org/dyn/closer.lua/lucene/solr

Enlazando allí con el sitio web recomendado para la descarga, en nuestro caso:

http://apache.uvigo.es/lucene/solr

Una vez allí, basta hacer clic en la carpeta con la versión más reciente de Solr, en nuestro caso la versión 7.2.1. En la nueva pantalla encontraremos diversas posibilidades de archivos comprimidos de esta versión (src.tgz, tgz, zip…). Nosotros emplearemos la versión 'zip', por lo que haremos clic en 'solr-7.2.1.zip'. Este archivo es el que se empleará para su instalación, tanto en Mac como en Windows. Una vez que hayamos hecho clic en 'solr-7.2.1.zip', el archivo se descargará en nuestro ordenador. Tened en cuenta que el archivo ocupa alrededor de 150 MB, por lo que puede tardar varios minutos en realizar esta tarea.

Habitualmente los archivos se descargan en la carpeta 'Descargas', pero posteriormente se puede mover el archivo 'solr-7.2.1.zip' desde allí al directorio de trabajo que se desee. Nosotros, por ejemplo, lo hemos descargado finalmente en el Escritorio (Desktop).

Allí descomprimiremos ese archivo con un programa compresor-descompresor (WinRAR en nuestro caso; si no se dispone del programa, puede consultarse el apéndice 'Instalación de WinRAR en Windows' al final de este documento). Para ello, con el botón derecho desplegamos el menú contextual y elegimos la opción 'Extraer aquí'. Tras unos segundos se mostrará en pantalla la carpeta con la versión más reciente, en nuestro caso 'solr-7.2.1'. El programa Solr no necesita más tareas para su instalación. Ya podemos empezar a emplearlo.

En adelante denominaremos $SOLR_INSTALL a la ruta donde hayamos situado la carpeta con el programa; en nuestro caso, la ruta hasta llegar a la carpeta 'solr-7.2.1'. Por tanto, de ahora en adelante:

En Mac, $SOLR_INSTALL es equivalente a:

/Users/juan/Desktop/solr-7.2.1

En Windows, $SOLR_INSTALL es equivalente a

C:\Users\juan\Desktop\solr-7.2.1

Instalación del programa Solr en Mac

La última versión del programa Solr (en nuestro caso, la 7.2.1) puede descargarse desde la página

http://www.apache.org/dyn/closer.lua/lucene/solr/7.2.1

Enlazando allí con el sitio web:

http://ftp.cixug.es/apache/lucene/solr/7.2.1

Una vez allí, basta hacer clic en 'solr-7.2.1.zip'. Este archivo es el que se empleará para su instalación, tanto en Mac como en Windows. El archivo se descargará en nuestro ordenador. Se puede mover el archivo zip al directorio de trabajo que se desee. Nosotros, por ejemplo, lo hemos descargado en el Escritorio (Desktop).

Allí descomprimiremos ese archivo, resultando la carpeta 'solr-7.2.1'. El programa Solr no necesita más tareas para su instalación. Ya podemos empezar a emplearlo.

En adelante denominaremos $SOLR_INSTALL a la ruta donde hayamos situado la carpeta con el programa; en nuestro caso, la ruta hasta llegar a la carpeta 'solr-7.2.1'. Por tanto, de ahora en adelante:

En Mac, $SOLR_INSTALL es equivalente a:

/Users/juan/Desktop/solr-7.2.1

En Windows, $SOLR_INSTALL es equivalente a

C:\Users\juan\Desktop\solr-7.2.1

Apertura y cierre del programa Solr en Windows

Si se ha trabajado con una cierta colección de ejemplo en Solr ('techproducts', por ejemplo) en ocasiones anteriores, y se desea iniciar el programa partiendo de la **versión modificada** en la última sesión del archivo de configuración, **NO ES NECESARIO** eliminar el subdirectorio correspondiente dentro de $SOLR_INSTALL\example ($SOLR_INSTALL\example\techproducts, siguiendo con el ejemplo) con todo su contenido.

Para iniciar el programa, en una CONSOLA, TERMINAL o LÍNEA DE COMANDO (para la apertura de una consola, terminal o línea de comandos, ver el apartado 'Apertura de una consola, terminal o línea de comandos'), teclear:

cd $SOLR_INSTALL\bin

[[esto es, ir al subdirectorio 'bin' dentro del directorio de instalación; en nuestro caso:

cd C:\Users\juan\Desktop\solr-7.2.1]]

Una vez en $SOLR_INSTALL\bin, teclear:

solr start –e techproducts

[[inicia Solr con la colección de ejemplo 'techproducts']]

La interfaz de administración del programa está en la página

http://localhost:8983/solr/

(debe esperarse unos segundos para que el programa se cargue correctamente)

Siempre que se desee se puede cerrar el programa Solr. Para ello, basta con ir al terminal desde donde se lanzó Solr y teclear: **Ctrl +C,** o cerrar el terminal (el método más seguro), o bien teclear en un terminal (para la apertura de un terminal o línea de comando, ver el apartado 'Apertura de una consola, terminal o línea de comandos'):

cd C:\Users\juan\Desktop\solr-7.2.1\bin

Una vez en $SOLR_INSTALL\bin, teclear:

solr stop -all

Una vez que ha iniciado el programa con la colección de ejemplo 'techproducts', **AL CERRAR EL PROGRAMA, LA COLECCIÓN 'TECHPRODUCTS' PERMANECE ALMACENADA Y DISPONIBLE.**

Si se ha acabado una sesión con el programa Solr habiendo instalado al arrancar la colección 'techproducts', se puede iniciar de nuevo Solr y simultáneamente cargar la colección de ejemplo 'techproducts', empleando para ello la misma orden vista anteriormente, sin que sea preciso realizar ninguna modificación en los archivos del programa (la eliminación del subdirectorio $SOLR_INSTALL/example/techproducts era necesaria en versiones anteriores).

Sin embargo, si al arrancar de nuevo el programa recibe un mensaje de error, pruebe a solucionar el problema eliminando previamente dicha colección. En resumen,

CADA VEZ QUE ACABE UNA SESIÓN CON EL PROGRAMA SOLR HABIENDO INSTALADO AL ARRANCAR LA COLECCIÓN 'TECHPRODUCTS', NO DEBE ELIMINAR EL DIRECTORIO '$SOLR_INSTALL\example\techproducts' EN SU INTEGRIDAD ANTES DE VOLVER A ARRANCAR EL PROGRAMA CON LA COLECCIÓN DE EJEMPLO 'techproducts'.

AHORA BIEN, SI AL ARRANCAR DE NUEVO EL PROGRAMA RECIBE UN MENSAJE DE ERROR, PRUEBE A ELIMINAR EL DIRECTORIO '$SOLR_INSTALL\example\techproducts' EN SU INTEGRIDAD ANTES DE VOLVER A ARRANCAR EL PROGRAMA CON LA COLECCIÓN DE EJEMPLO 'techproducts'. SIEMPRE QUE DICHO DIRECTORIO SE HAYA ELIMINADO, SE PUEDE VOLVER A ARRANCAR EL PROGRAMA CON LA MISMA ORDEN INDICADA ANTERIORMENTE.

Apertura y cierre del programa Solr en Mac

Para iniciar el programa, en una CONSOLA, TERMINAL o LÍNEA DE COMANDO (para la apertura de una consola, terminal o línea de comandos, ver el apartado 'Apertura de una consola, terminal o línea de comandos'), teclear:

cd /Users/juan/Desktop/solr-7.2.1/bin

[[esto es, ir al subdirectorio 'bin' dentro del directorio de instalación]]

Una vez en $SOLR_INSTALL/bin, teclear:

solr start –e techproducts

[[inicia Solr con la colección de ejemplo 'techproducts']]

La interfaz de administración del programa está en la página

http://localhost:8983/solr/

(debe esperarse unos segundos para que el programa se cargue correctamente).

Siempre que se desee se puede cerrar el programa Solr. Para ello, basta con ir al terminal desde donde se lanzó Solr y teclear: **Ctrl +C,** o cerrar el terminal (el método más seguro), o bien teclear en un terminal (para la apertura de un terminal o línea de comando, ver el apartado 'Apertura de una consola, terminal o línea de comandos'):

cd /Users/juan/Desktop/solr-7.2.1/bin

Una vez en $SOLR_INSTALL/bin, teclear:

solr stop -all

Una vez que ha iniciado el programa con la colección de ejemplo 'techproducts', **AL CERRAR EL PROGRAMA, LA COLECCIÓN 'TECHPRODUCTS' PERMANECE ALMACENADA Y DISPONIBLE.**

Si se ha acabado una sesión con el programa Solr habiendo instalado al arrancar la colección 'techproducts', se puede iniciar de nuevo Solr y simultáneamente cargar la

colección de ejemplo ‘techproducts’, empleando para ello la misma orden vista anteriormente, sin que sea preciso realizar ninguna modificación en los archivos del programa (la eliminación del subdirectorio $SOLR_INSTALL/example/techproducts era necesaria en versiones anteriores).

Sin embargo, si al arrancar de nuevo el programa recibe un mensaje de error, pruebe a solucionar el problema eliminando previamente dicha colección. En resumen,

CADA VEZ QUE ACABE UNA SESIÓN CON EL PROGRAMA SOLR HABIENDO INSTALADO AL ARRANCAR LA COLECCIÓN ‘TECHPRODUCTS’, NO DEBE ELIMINAR EL DIRECTORIO ‘$SOLR_INSTALL/example/techproducts’ EN SU INTEGRIDAD ANTES DE VOLVER A ARRANCAR EL PROGRAMA CON LA COLECCIÓN DE EJEMPLO ‘techproducts’.

AHORA BIEN, SI AL ARRANCAR DE NUEVO EL PROGRAMA RECIBE UN MENSAJE DE ERROR, PRUEBE A ELIMINAR EL DIRECTORIO ‘$SOLR_INSTALL/example/techproducts’ EN SU INTEGRIDAD ANTES DE VOLVER A ARRANCAR EL PROGRAMA CON LA COLECCIÓN DE EJEMPLO ‘techproducts’. SIEMPRE QUE DICHO DIRECTORIO SE HAYA ELIMINADO, SE PUEDE VOLVER A ARRANCAR EL PROGRAMA CON LA MISMA ORDEN INDICADA ANTERIORMENTE.

A partir de este momento, el taller se desarrollará teniendo en cuenta un sistema Windows, aunque los procesos son muy semejantes. Pueden consultarse los comandos principales en un sistema Mac en la página

http://lucene.apache.org/solr/quickstart.html

Eliminación de palabras vacías

La tarea de análisis efectuada con los documentos de una colección consta de un proceso de tokenización (o de segmentación de la cadena de caracteres de cada documento en 'palabras' o 'tokens'), más una serie de procesos optativos de filtrado de los tokens, entre los que habitualmente se hallan los siguientes:

- Reducción de todos los caracteres a minúsculas
- Eliminación de palabras vacías
- Stemming

En este apartado veremos cómo modificar la configuración del programa Solr para que efectúe un proceso de eliminación de palabras vacías conforme a un listado recogido en el fichero:

$SOLR_INSTALL\example\techproducts\solr\techproducts\conf\lang\ stopwords_en.txt

Comprobaremos a continuación los resultados del proceso con un ejemplo.

La configuración del programa Solr se recoge principalmente en dos ficheros: 'solrconfig.xml' y 'managed-schema.xml', localizados habitualmente en un subdirectorio 'conf'.

Como hemos iniciado el programa con la colección/core de ejemplo 'techproducts', debemos abrir el archivo:

$SOLR_INSTALL\example\techproducts\solr\techproducts\conf\managed-schema.xml

Con un editor de textos (con Wordpad o Notepad++, por ejemplo).

Todos los procesos de que consta la tarea de análisis se recogen en el schema del programa dentro de un apartado <analyzer>. A su vez, consta de dos partes esenciales:

- Una parte/type denominada <analyzer type=”index”>, que incluye los procesos que se llevarán a cabo con los documentos de la colección en el momento de la indexación.
- Otra parte/type denominada <analyzer type=“query”>, que incluye los procesos que se realizarán al analizar las consultas introducidas por los usuarios.

A su vez, el apartado “analyzer” estará incluido dentro del epígrafe < fieldType name=”…” > correspondiente al tipo de documento empleado en nuestra colección (ya que se debe configurar el programa con la descripción de cada uno de los tipos de documentos que componen la colección manejada en cada momento). En nuestro caso, la colección/core ‘techproducts’ cuenta con un tipo de documento “text_general” válido para todos ellos (< fieldType name=”text_general” >).

En consecuencia, debemos modificar las siguientes líneas del archivo de configuración ‘managed-schema.xml’ (en negrita las líneas concretas que se han modificado):

```
<fieldType name=”text_general”
        class=”solr.TextField”
        positionIncrementGap=”100”>
        <analyzer type=”index”>
            ……
            <filter class=”solr.StopFilterFactory” ignoreCase=”true”
                words=”lang/stopwords_en.txt”
                />
            …….
        </analyzer>
```

```
<analyzer type="query">
    ……
    <filter class="solr.StopFilterFactory" ignoreCase="true"
        words="lang/stopwords_en.txt"
        />
……
……
```

Una vez guardados los cambios efectuados en el archivo 'managed-schema.xml', debemos reindexar la colección/core de ejemplo 'techproducts' conforme al nuevo proceso de análisis configurado. Para ello, seguimos los siguientes pasos:

1.- Eliminamos primeramente la indexación previa realizada con la colección/core de ejemplo 'techproducts'. Ello se puede conseguir yendo al epígrafe 'Core Admin' del menú izquierdo y haciendo clic en 'Unload'.

2.- Indexamos de nuevo la colección/core de ejemplo 'techproducts'. En un terminal (para la apertura de un terminal o línea de comando, ver el apartado 'Apertura de una consola, terminal o línea de comandos'), tecleamos (en negrita los comandos):

```
cd $SOLR_INSTALL\bin
solr start -e techproducts
```

Solr facilita la comprobación del funcionamiento de la cadena de análisis configurada de manera directa, sin necesidad de realizar consultas de prueba. Para ello dispone de un formulario 'Analysis', que surge en un menú contextual después de elegir la colección/core en el menú lateral izquierdo de la pantalla de administración.

Una vez que estemos en la pantalla de 'Analysis', seguimos los siguientes pasos:

1.- Introduce el siguiente texto de prueba en el área de texto 'Field Value (Index)':

#Good! :) Drinking a latte at Caffé Grecco in SF's historic North Beach… Sent from my i-pad [[ejemplo basado en Grainger & Potter (2014), p. 117]]

2.- Selecciona 'text_general' en el apartado 'Analyse Fieldname/FieldType'

3.- Haz clic en el botón 'Analyse Values' para ver el resultado.

Solr indicará en pantalla los procesos seguidos en el análisis del texto durante su indexación conforme al fieldType 'text_general'. Solr utiliza las siguientes abreviaturas:

ST: StandardTokenizer [Tokenización]

SF: StopFilter [Eliminación de palabras vacías]

LCF: LowerCaseFilter [Conversión de todos los caracteres a minúsculas]

En cada fila (las que comienzan con ST, SF o LCF, por ejemplo) se pueden observar los términos resultantes tras cada uno de dichos procesos. Si una determinada palabra no aparece a partir de una cierta fila (ST, SF o LCF, por ejemplo), significa que la palabra se ha eliminado como término de indexación tras el sometimiento a dicho proceso. A partir de ese momento ya no aparecerá en los sucesivos procesos que incluya el analizador del programa.

EJERCICIO 1

Tras introducir en el área de texto 'Field Value (Index)' el siguiente texto de prueba:

#Good! :) Drinking a latte at Caffé Grecco in SF's historic North Beach… Sent from my i-pad

responde a las siguientes preguntas:

1. ¿Qué procesos de análisis se efectúan y en qué orden?
2. ¿Elimina acentos este proceso de análisis?
3. ¿Cuál es el token que resulta tras el análisis de 'SF's'?
4. ¿Cuál es el token que resulta tras de análisis de 'in'?
5. ¿Cuál es el token que resulta de 'i-pad'?

SOLUCIÓN EJERCICIO 1

1. ¿Qué procesos de análisis se efectúan y en qué orden?
   ST (Tokenización), SF (Eliminación de palabras vacías) y LCF (Conversión a minúsculas)
2. ¿Elimina acentos este proceso de análisis?
   No
3. ¿Cuál es el token que resulta tras el análisis de 'SF's'?
   sf's
4. ¿Cuál es el token que resulta tras de análisis de 'in'?
   Ninguno, el token se elimina
5. ¿Cuál es el token que resulta de 'i-pad'?
   i pad [el token queda dividido en dos términos]

Este mismo formulario permite comprobar si ante una cierta consulta se recuperaría este documento de prueba sin tener que indexarlo. Para ello, seguimos los siguientes pasos:

1.- Introduce 'drinking a latte' [sin comillas] en el área de texto 'Field Value (Query)'

2.- Haz clic en el botón 'Analyse Values'

Solr destacará todos los términos del documento que concuerdan con la consulta introducida.

EJERCICIO 2

Tras introducir la consulta:

drinking a latte

en el área de texto 'Field Value (Query)', responde a la siguiente pregunta:

1. ¿Qué términos del documento de prueba concuerdan con dicha consulta?

Si se introduce ahora la consulta 'San Francisco drink cafe ipad' [sin comillas] y se hace clic en el botón 'Analyse Values',

2. ¿Qué términos del documento de prueba coindicen ahora con dicha consulta?
3. ¿Qué conclusiones deduce de cara a la indexación y recuperación de documentos en una colección?

SOLUCIÓN EJERCICIO 2

1. ¿Qué términos del documento de prueba concuerdan con dicha consulta?
   'drinking' y 'latte'
2. ¿Qué términos del documento de prueba coindicen ahora con dicha consulta?
   Ningún término
3. ¿Qué conclusiones deduce de cara a la indexación y recuperación de documentos en una colección?
   Únicamente los términos que coincidan plenamente con un término del fichero diccionario serán recuperados

Stemming o reducción morfológica

Para incluir el proceso de stemming dentro de la cadena de análisis de los textos de la colección, debemos indicar al programa que utilice alguno de los algoritmos de stemming que soporta Solr dentro de la tarea de análisis. Nosotros emplearemos aquí el algoritmo de Porter en inglés.

Como ya indicamos en el epígrafe anterior, dedicado a la eliminación de palabras vacías, debemos modificar el archivo 'managed-schema.xml' de configuración de la colección/core de ejemplo 'techproducts' con un editor de textos (Wordpad o Notepad++, por ejemplo). Este archivo de configuración se encuentra en la siguiente ruta:

$SOLR_INSTALL\example\techproducts\solr\techproducts\conf\managed-schema.xml

incluyendo las siguientes líneas (en negrita las líneas concretas que se han añadido):

```
<fieldType name="text_general"
        class="solr.TextField"
        positionIncrementGap="100">
        <analyzer type="index">
            ……….
            <filter class="solr.StopFilterFactory" ignoreCase="true"
                words="lang/stopwords_en.txt" />
            <filter class="solr.LowerCaseFilterFactory" />
            <filter class="solr.PorterStemFilterFactory" />
        </analyzer>
        <analyzer type="query">
            ……….
            <filter class="solr.StopFilterFactory" ignoreCase="true"
                words="lang/stopwords_en.txt" />
            <filter class="solr.LowerCaseFilterFactory" />
            <filter class="solr.PorterStemFilterFactory" />
</analyzer>
```

Una vez guardados los cambios efectuados en el archivo 'managed-schema.xml', debemos reindexar otra vez la colección/core de ejemplo 'techproducts' conforme al

nuevo proceso de análisis configurado. Para ello, seguimos los mismos pasos realizados anteriormente, al incluir la eliminación de palabras vacías:

1.- Eliminamos primeramente la indexación previa realizada con la colección/core de ejemplo 'techproducts'. Ello se puede conseguir yendo al epígrafe 'Core Admin' del menú izquierdo y haciendo clic en 'Unload'.

2.- Indexamos de nuevo la colección/core de ejemplo 'techproducts'. En un terminal (para la apertura de un terminal o línea de comandos, ver el apartado 'Apertura de una consola, terminal o línea de comandos'), tecleamos (los comandos figuran en negritas):

```
cd $SOLR_INSTALL\bin
solr start -e techproducts
```

Comprobamos ahora el funcionamiento mediante el formulario 'Analysis', tras la elección previa de la colección/core 'techproducts' en el menú lateral izquierdo de la pantalla de administración. Los pasos a seguir son:

1.- Introduce el mismo texto de prueba empleado en el EJERCICIO 1 (vid. p. 20) en el área de texto 'Field Value (Index)':

```
#Good! :)  Drinking a latte at Caffé Grecco in SF's historic North Beach… Sent from my i-pad
```

2.- Selecciona 'text_general' en el apartado 'Analyse Fieldname/FieldType'

3.- Haz clic en el botón 'Analyse Values' para ver el resultado.

EJERCICIO 3

Tras introducir el texto de prueba:

#Good! :) Drinking a latte at Caffé Grecco in SF's historic North Beach… Sent from my i-pad

en el área de texto 'Field Value (Index)', responde a las siguientes preguntas:

1. ¿Qué procesos de análisis se efectúan ahora y en qué orden?
2. ¿Elimina acentos este proceso de análisis?
3. ¿Cuál es el token que resulta tras el análisis de "SF's"?
4. ¿Cuál es el token que resulta tras de análisis de "in"?
5. ¿Cuál es el token que resulta de "drinking"?

SOLUCIÓN EJERCICIO 3

1. ¿Qué procesos de análisis se efectúan ahora y en qué orden?
   ST, SF, LCF, PSF
2. ¿Elimina acentos este proceso de análisis?
   No.
3. ¿Cuál es el token que resulta tras el análisis de "SF's"?
   sf'
4. ¿Cuál es el token que resulta tras de análisis de "in"?
   Ninguno; el token es eliminado
5. ¿Cuál es el token que resulta de "drinking"?
   drink

EJERCICIO 4

Tras introducir la consulta:

drinks a lattes

en el área de texto 'Field Value (Query)', responde a las siguientes preguntas:

1. ¿Qué términos del documento de prueba concuerdan con dicha consulta?
2. ¿Qué diferencias observa en la recuperación en relación a la cadena de análisis efectuada en el ejercicio 2?

SOLUCIÓN EJERCICIO 4

1. ¿Qué términos del documento de prueba concuerdan con dicha consulta?
   'drink' y 'latt'
2. ¿Qué diferencias observa en la recuperación en relación a la cadena de análisis efectuada en el ejercicio 2?
   El algoritmo de stemming en la cadena de análisis, tanto de los documentos como de las consultas, ha permitido recuperar el documento porque ha convertido los términos de manera que ahora sí coinciden plenamente los términos de la consulta y los términos disponibles en el fichero diccionario

Creación de una colección

Una de las tareas más comunes con un sistema de recuperación de información consiste en crear e incorporar nuestra propia colección al sistema. En este apartado crearemos una colección/core denominada 'prueba', localizada en la ruta:

$SOLR_INSTALL\example\prueba

Para ello efectuamos los siguientes pasos:

1.- Creamos el directorio 'prueba' en la ruta:

$SOLR_INSTALL\example\prueba

2.- Copiamos el subdirectorio 'conf' COMPLETO, disponible en:

$SOLR_INSTALL\example\techproducts\solr\techproducts\conf

en la ruta:

$SOLR_INSTALL\example\prueba\conf

3.- Creamos el subdirectorio 'data' en la ruta:

$SOLR_INSTALL\example\prueba\data

4.- Vamos a la interfaz de administración y en el menú izquierdo hacemos clic en 'Core Admin'.

5.- Hacemos clic en 'Add Core' y rellenamos el formulario como sigue:

Name: prueba instanceDir: $SOLR_INSTALL\example\prueba dataDir: $SOLR_INSTALL\example\prueba\data config: $SOLR_INSTALL\example\prueba\conf\solrconfig.xml schema: $SOLR_INSTALL\example\prueba\conf\managed-schema.xml

6.- Hacemos clic en ‘Add Core’

7.- En el menú lateral izquierdo, desplegar el menú de colecciones/cores disponibles (texto ‘Core Selector’ enmarcado en una fina línea) y elegimos la colección/core ‘prueba’. En el apartado ‘Overview’ nos confirma que todavía esta colección no tiene documentos indexados (‘Num Docs: 0’).

Indexación de documentos

Para facilitar la tarea de recuperación, partiremos de una colección de prueba que se ha obtenido a partir de los fondos sobre ciencias de la Biblioteca de la Universidad Complutense de Madrid. Dicha colección (disponible en el Campus Virtual de la asignatura) **DEBE COPIARSE en el directorio $SOLR_INSTALL\colciencias**, estando compuesta por 1000 asientos bibliográficos.

En caso de no tener acceso a esta colección, basta crear una carpeta con varios archivos de texto plano, cada uno con denominación distinta y la misma extensión 'txt'. Si dicha carpeta lleva por nombre 'colciencias' y se copia en el directorio $SOLR_INSTALL\colciencias, puede seguirse sin variación alguna lo dicho en este epígrafe.

Es necesario incorporar dichos documentos a nuestro sistema e indexarlos. Para ello, existe una herramienta en Solr, denominada SimplePostTool, que rastrea todo un directorio de archivos de manera recursiva, accediendo a todos los archivos con extensión HTML, PDF, TXT, etc., enviando su contenido en crudo a Solr para su análisis e indexación.

El comando para ello es el siguiente, **siempre que estemos situados en $SOLR_INSTALL**, esto es, el directorio donde esté instalado Solr:

```
java –classpath example\exampledocs\post.jar
-Dauto -Drecursive -Dc=prueba org.apache.solr.util.SimplePostTool colciencias\
```

Una vez terminada la operación de indexación, aparecerá en la pantalla la siguiente respuesta:

1001 files indexed

Ya tenemos indexados los 1000 documentos en la colección/core 'prueba'. Se puede comprobar en la interfaz de administración, en 'Overview' de la colección/core 'prueba', donde indica 'NumDocs: 1001'.

Recuperación de documentos por defecto

Solr, por defecto, efectúa el cálculo de la respuesta en relación a las consultas de los usuarios mediante una mezcla de modelos booleano y vectorial con normalización (la clase DefaultSimilarityFactory realiza tal proceso de recuperación). A raíz de la introducción de una consulta, Solr primeramente selecciona –conforme al modelo booleano- los documentos que verifican la consulta. A su vez, con el resultado obtenido conforme al modelo booleano, efectúa una ordenación de los documentos de ese resultado previo conforme a un modelo vectorial con normalización.

Como en el epígrafe anterior hemos indexado la colección/core 'prueba', ya podemos realizar búsquedas en Solr cargando la interfaz de búsqueda:

http://localhost:8983/solr/#/prueba/query

Esta pantalla de búsqueda puede obtenerse igualmente si en la interfaz de administración, una vez elegida la colección/core 'prueba', seleccionamos 'Query'.

- **Búsqueda de un término aislado**

  Basta introducir el término en el campo 'q'. Suele optarse también por seleccionar 'json' como el formato de la respuesta en el campo 'wt', por su mayor facilidad de visualización. El resto puede dejarse con los valores por defecto que figuran.

EJERCICIO 5

Realizar una búsqueda del término **congreso** en la colección:

1. ¿Cuántos documentos se obtienen en la respuesta?
2. ¿Qué documentos/fichas ocupan las cinco primeras posiciones?

SOLUCIÓN EJERCICIO 5

1. ¿Cuántos documentos se obtienen en la respuesta?

   62 documentos
2. ¿Qué documentos/fichas ocupan las cinco primeras posiciones?
   1. Ficha147
   2. Ficha89
   3. Ficha318
   4. Ficha319
   5. Ficha180

- **Búsqueda por frases**

  Se realiza rodeando la frase entre comillas dobles en el campo ‘q’.

EJERCICIO 6

Realizar una búsqueda con la frase ‘analisis multivariante’ (sin acento):

1. ¿Cuántos documentos se obtienen en la respuesta?
2. ¿Qué documentos/fichas ocupan las cinco primeras posiciones?

SOLUCIÓN EJERCICIO 6

1. ¿Cuántos documentos se obtienen en la respuesta?

   2 documentos
2. ¿Qué documentos/fichas ocupan las cinco primeras posiciones?
   1. Ficha238
   2. Ficha435

- **Búsqueda con AND**

  Se realiza anteponiendo a cada término el signo + (sin espacios en medio).

  También es posible utilizar la conectiva AND (en mayúsculas) entre los dos términos.

EJERCICIO 7

Realizar una búsqueda de modo que me devuelva los documentos en los que aparezcan simultáneamente los términos 'elsevier' y 'manrique':

1. ¿Cómo efectuaría la consulta?
2. ¿Cuántos registros se obtienen en la respuesta?
3. ¿Qué documentos/fichas ocupan las cinco primeras posiciones?

SOLUCIÓN EJERCICIO 7

4. ¿Cómo efectuaría la consulta?

   +elsevier +manrique
5. ¿Cuántos registros se obtienen en la respuesta?

   1 documento
6. ¿Qué documentos/fichas ocupan las cinco primeras posiciones?

   1. Ficha21

- **Búsqueda con OR**

  Se realiza simplemente introduciendo los términos con un espacio entre ellos (operador por defecto). También es posible utilizar la conectiva OR (en mayúsculas) entre los dos términos.

EJERCICIO 8

Realizar una búsqueda con los documentos en los que aparezcan al menos una de las siguientes palabras: 'jornadas', 'tesis' y 'programa'.

1. ¿Cómo efectuaría la consulta?
2. ¿Cuántos registros se obtienen en la respuesta?
3. ¿Qué documentos/fichas ocupan las cinco primeras posiciones?

SOLUCIÓN EJERCICIO 8

1. ¿Cómo efectuaría la consulta?

   jornadas tesis programa
2. ¿Cuántos registros se obtienen en la respuesta?

   102 documentos
3. ¿Qué documentos/fichas ocupan las cinco primeras posiciones?
   1. Ficha55
   2. Ficha113
   3. Ficha277
   4. Ficha3
   5. Ficha464

- **Búsqueda con NOT**

  Se realiza anteponiendo al término el signo – (sin espacios en medio).

EJERCICIO 9

Realizar una búsqueda con los documentos que no contengan el término 'universitario'.

1. ¿Cómo efectuaría la consulta?
2. ¿Cuántos documentos se obtienen en la respuesta?
3. ¿Qué documentos/fichas ocupan las cinco primeras posiciones?

Posteriormente, realizar una búsqueda con los documentos que contengan el término 'ciencias' y en los que no aparezca el término 'universitario'.

4. ¿Cómo efectuaría la consulta?
5. ¿Cuántos documentos se obtienen en la respuesta?
6. ¿Qué documentos/fichas ocupan las cinco primeras posiciones?

SOLUCIÓN EJERCICIO 9

1. ¿Cómo efectuaría la consulta?
   -universitario
2. ¿Cuántos documentos se obtienen en la respuesta?
   979 documentos
3. ¿Qué documentos/fichas ocupan las cinco primeras posiciones?
   1. Ficha1
   2. Ficha10
   3. Ficha100
   4. Ficha1000
   5. Ficha1001
4. ¿Cómo efectuaría la consulta?
   +ciencias -universitario
5. ¿Cuántos documentos se obtienen en la respuesta?
   977 documentos

6. ¿Qué documentos/fichas ocupan las cinco primeras posiciones?
    1. Ficha518
    2. Ficha892
    3. Ficha994
    4. Ficha1001
    5. Ficha188

Recuperación conforme a diversos modelos

Solr permite utilizar otros modelos de recuperación de información, entre los que destacan:

- BM25SimilarityFactory: clase que desarrolla el modelo probabilístico con normalización BM25.
- DFRSimilarityFactory: clase que desarrolla el modelo DFR (Divergence From Randomness). Se basa en la comparación de la distribución de las frecuencias de los términos en cada documento de la colección para determinar el peso de los términos en cada documento. Se valora fundamentalmente que la frecuencia sea distinta de la esperada en el conjunto de la colección.

Para cambiar el modelo de recuperación con el que el sistema calcula la respuesta ante las consultas, debe modificarse la configuración de la fase de recuperación del sistema. En concreto, debe cambiarse el algoritmo seguido por el SRI para hallar la respuesta a cada consulta del usuario. Ello implica de nuevo la edición del archivo 'managed-schema', específicamente un cambio en la sección correspondiente a 'similarity'.

A continuación comprobaremos los cambios en la respuesta de Solr desarrollando dos sistemas que indexan de igual forma la misma colección de prueba (colpdf), pero diferenciándose en el proceso seguido en la recuperación: mientras uno de ellos utiliza el modelo de recuperación por defecto (colección/core 'pruebadefecto'), otro utilizará el modelo BM25 (colección/core 'pruebabm25').

Crearemos en primer lugar una nueva colección/core 'pruebadefecto' que cumpla las siguientes condiciones:

1. Indexa la colección 'colpdf'.
2. Utiliza el modelo booleano+vectorial de recuperación (modelo por defecto).

Para crear la colección/core ‘pruebadefecto’, seguimos los pasos indicados en el epígrafe ‘Creación de una colección’.

Para realizar la comparación de modelos de recuperación utilizaremos una colección de prueba ‘colpdf’, compuesta por 20 artículos científicos en formato pdf. Dicha colección (disponible en el Campus Virtual de la asignatura) **DEBE COPIARSE en el directorio $SOLR_INSTALL\colpdf**.

En caso de no tener acceso a esta colección, basta crear una carpeta con varios archivos en formato pdf, cada uno con denominación distinta y la misma extensión ‘pdf’. Si dicha carpeta lleva por nombre ‘colpdf’ y se copia en el directorio $SOLR_INSTALL\colpdf, puede seguirse sin variación alguna lo dicho en este epígrafe.

Es necesario incorporar dichos documentos a nuestra colección/core ‘pruebadefecto’ e indexarlos. Para ello seguimos los pasos indicados en el epígrafe ‘Indexación de documentos’.

Dado que deseamos que la colección/core ‘pruebadefecto’ emplee el modelo de recuperación por defecto, no es necesario efectuar ningún cambio en el archivo de configuración.

Crearemos ahora otra colección/core ‘pruebabm25’ que cumpla las siguientes condiciones:

1. Indexa la misma colección ‘colpdf’.
2. Utiliza el modelo BM25 de recuperación.

Para crear la colección/core ‘pruebabm25’, seguimos los pasos indicados en el epígrafe ‘Creación de una colección’.

Para incorporar la colección ‘colpdf’ a la colección/core ‘pruebabm25’, seguiremos los pasos indicados en el epígrafe ‘Indexación de documentos’.

Por otra parte, para que el sistema emplee el modelo BM25 en el proceso de recuperación, modificamos el archivo:

$SOLR_INSTALL\example\pruebabm25\conf\managed-schema.xml

Añadiendo la siguiente entrada al final de dicho archivo:

```
<similarity   class=”solr.BM25SimilarityFactory”>
      <float   name=”k1”>1.6</float>
      <float   name=”b”>0.35</float>
</similarity>
```

Los valores más habituales para los parámetros k1 y b son los siguientes:

1. k1=1.2
2. b=0.75

Sin embargo, con el fin de observar mejor la diferencia de resultados según el modelo empleado, hemos elegido los valores k1=1.6 y b=0.35.

Una vez guardados los cambios efectuados en el archivo ‘managed-schema.xml’, debemos:

1. Eliminar la indexación previa realizada. Para ello, seleccionar ‘Core Admin’ en el menú izquierdo y haz clic en ‘Unload’.

2. Borrar el subdirectorio COMPLETO ‘data’ en la ruta:
   $SOLR_INSTALL\example\pruebabm25\data

3. Crear un nuevo subdirectorio ‘data’ COMPLETAMENTE VACÍO en:
   $SOLR_INSTALL\example\pruebabm25\data

4. Hacer clic en ‘Add Core’ y rellenar el formulario como sigue:

```
Name: pruebabm25
instanceDir: $SOLR_INSTALL\example\pruebabm25
dataDir: $SOLR_INSTALL\example\pruebabm25\data
config: $SOLR_INSTALL\example\pruebabm25\conf\solrconfig.xml
schema: $SOLR_INSTALL\example\pruebabm25\conf\managed-schema.xml
```

5. Reindexar la colección/core ‘pruebabm25’ conforme al nuevo archivo de configuración. Para ello, se puede consultar el epígrafe ‘Indexación de documentos’.

Por último, comparamos los resultados obtenidos con estos dos modelos de recuperación repitiendo las mismas búsquedas con la colección/core ‘pruebadefecto’ y ‘pruebabm25’. Para observar mejor la diferencia de resultados en una y otra colección, introduzca los siguientes parámetros al efectuar las consultas:

1. Campo ‘q’: la consulta que se formule en cada caso
2. Campo ‘wt’: json
3. Campo ‘fl’: *,score [todos los campos habituales más la puntuación obtenida en la relevancia del documento conforme al modelo empleado]

EJERCICIO 10

Realizar una búsqueda en las colecciones/core ‘pruebadefecto’ y ‘pruebabm25’ del término ‘obsolescence’:

1. ¿Cuántos documentos se obtienen en la respuesta en cada colección?
2. ¿Qué artículos ocupan las cinco primeras posiciones en cada caso, en qué orden y con qué puntuación/score?
3. ¿Observa algún cambio (orden o puntuación/score) de los cinco primeros documentos de la respuesta según el modelo de recuperación empleado?

SOLUCIÓN EJERCICIO 10

1. ¿Cuántos documentos se obtienen en la respuesta en cada colección?
   11 documentos
2. ¿Qué artículos ocupan las cinco primeras posiciones en cada caso, en qué orden y con qué puntuación/score?

pruebadefecto:	pruebabm25:
1. Art12 (1.3063076)	1. Art12 (1.5375448)
2. Art6 (1.3054152)	2. Art5 (1.525011)
3. Art8 (1.2973288)	3. Art6 (1.5227213)
4. Art5 (1.2970226)	4. Art8 (1.5202005)
5. Art20 (1.2915925)	5. Art20 (1.5108013)

3. ¿Observa algún cambio (orden o puntuación/score) de los cinco primeros documentos de la respuesta según el modelo de recuperación empleado?
   Existen cambios tanto en el orden de los documentos como en la puntuación

EJERCICIO 11

Realizar una búsqueda en las colecciones/core ‘pruebadefecto’ y ‘pruebabm25’ de los documentos donde aparezcan simultáneamente los términos ‘world’, ‘wide’ y ‘web’:

1. ¿Cuántos documentos se obtienen en la respuesta en cada colección?
2. ¿Qué artículos ocupan las cinco primeras posiciones en cada caso, en qué orden y con qué puntuación/score?
3. ¿Observa algún cambio en el orden o en la puntuación/score de los cinco primeros documentos de la respuesta según el modelo de recuperación empleado?

SOLUCIÓN EJERCICIO 11

1. ¿Cuántos documentos se obtienen en la respuesta en cada colección?
   2 documentos
2. ¿Qué artículos ocupan las cinco primeras posiciones en cada caso, en qué orden y con qué puntuación/score?

pruebadefecto:	pruebabm25:
1. Art4 (5.192871)	1. Art3 (5.9589663)
2. Art3 (5.146044)	2. Art4 (5.8930655)

3. ¿Observa algún cambio en el orden o en la puntuación/score de los cinco primeros documentos de la respuesta según el modelo de recuperación empleado?
   Existen cambios tanto en el orden de los documentos como en la puntuación

EJERCICIO 12

Realizar una búsqueda en las colecciones/core ‘pruebadefecto’ y ‘pruebabm25’ de los documentos que no contengan el término ‘age’:

1. ¿Cuántos documentos se obtienen en la respuesta en cada colección?
2. ¿Qué artículos ocupan las cinco primeras posiciones en cada caso, en qué orden y con qué puntuación/score?
3. ¿Observa algún cambio en el orden o en la puntuación/score de los cinco primeros documentos de la respuesta según el modelo de recuperación empleado?

SOLUCIÓN EJERCICIO 12

1. ¿Cuántos documentos se obtienen en la respuesta en cada colección?
   6 documentos
2. ¿Qué artículos ocupan las cinco primeras posiciones en cada caso, en qué orden y con qué puntuación/score?

pruebadefecto:	pruebabm25:
1. Art10 (1.0)	1. Art10 (1.0)
2. Art11 (1.0)	2. Art11 (1.0)
3. Art15 (1.0)	3. Art15 (1.0)
4. Art16 (1.0)	4. Art16 (1.0)
5. Art17 (1.0)	5. Art17 (1.0)

3. ¿Observa algún cambio en el orden o en la puntuación/score de los cinco primeros documentos de la respuesta según el modelo de recuperación empleado?
   No existe ninguna diferencia. Adviértase que el sistema no puede realmente ordenar los documentos, pues todos ellos verifican EN LA MISMA MEDIDA que NO INCLUYEN el término en cuestión.

EJERCICIO 13

Realizar una búsqueda en las colecciones/core ‘pruebadefecto’ y ‘pruebabm25’ de los documentos en los que aparezcan al menos una de las siguientes palabras: ‘music’ y ‘citation’:

1. ¿Cuántos documentos se obtienen en la respuesta en cada colección?
2. ¿Qué artículos ocupan las cinco primeras posiciones en cada caso, en qué orden y con qué puntuación/score?
3. ¿Observa algún cambio en el orden o en la puntuación/score de los cinco primeros documentos de la respuesta según el modelo de recuperación?

SOLUCIÓN EJERCICIO 13

1. ¿Cuántos documentos se obtienen en la respuesta en cada colección?
   12 documentos
2. ¿Qué artículos ocupan las cinco primeras posiciones en cada caso, en qué orden y con qué puntuación/score?

pruebadefecto:	pruebabm25:
1. Art5 (5.936718)	1. Art5 (6.9960866)
2. Art3 (4.310213)	2. Art3 (5.01267)
3. Art2 (1.3117106)	3. Art2 (1.5449018)
4. Art7 (1.3098216)	4. Art19 (1.5414988)
5. Art19 (1.3089157)	5. Art7 (1.5394377)

3. ¿Observa algún cambio en el orden o en la puntuación/score de los cinco primeros documentos de la respuesta según el modelo de recuperación empleado?
   Existen cambios tanto en el orden de los documentos como en la puntuación

EJERCICIO 14

Realizar una búsqueda en las colecciones/core 'pruebadefecto' y 'pruebabm25' de la frase 'random indexing':

1. ¿Cuántos documentos se obtienen en la respuesta en cada colección?
2. ¿Qué artículos ocupan las cinco primeras posiciones en cada caso, en qué orden y con qué puntuación/score?
3. ¿Observa algún cambio en el orden o en la puntuación/score de los cinco primeros documentos de la respuesta según el modelo de recuperación empleado?

SOLUCIÓN EJERCICIO 14

1. ¿Cuántos documentos se obtienen en la respuesta en cada colección?
   1 documento
2. ¿Qué artículos ocupan las cinco primeras posiciones en cada caso, en qué orden y con qué puntuación/score?
   pruebadefecto: pruebabm25:
   1. Art16 (2.593104) 1. Art16 (3.0147972)
3. ¿Observa algún cambio en el orden o en la puntuación/score de los cinco primeros documentos de la respuesta según el modelo de recuperación empleado?
   No existe diferencia en el número/orden de los documentos, pero sí en la puntuación obtenida, aunque en este caso carece de importancia al tratarse de un único documento

A continuación comprobaremos los cambios en la respuesta de Solr desarrollando un sistema que indexa de igual forma la misma colección de prueba (colpdf), pero diferenciándose en que utilizará el modelo de recuperación DFR (Divergence From Randomness).

Crearemos en primer lugar una nueva colección/core ‘pruebadfr’. Para ello, seguimos los pasos indicados en el epígrafe ‘Creación de una colección’.

Para incorporar la colección ‘colpdf’ a la colección/core ‘pruebadfr’, seguiremos los pasos indicados en el epígrafe ‘Indexación de documentos’.

Para que el sistema emplee el modelo DFR en el proceso de recuperación, modificamos el archivo:

$SOLR_INSTALL\example\pruebadfr\conf\managed-schema.xml

Añadiendo la siguiente entrada (es una de las opciones más frecuentes) al final de dicho archivo:

```
<similarity   class=”solr.DFRSimilarityFactory”>
        <str name=”basicModel” >P</str>
        <str name=”afterEffect”>L</str>
        <str name=”normalization”>H2</str>
        <float   name=”c”>7</float>                    [[También suele usarse c=3]]
</similarity>
```

Una vez guardados los cambios efectuados en el archivo ‘managed-schema.xml’, debemos:

1. Eliminar la indexación previa realizada. Para ello, seleccionar ‘Core Admin’ en el menú izquierdo, seleccionar luego la colección/core ‘pruebadfr’ y hacer clic en ‘Unload’.
2. Borrar el subdirectorio COMPLETO ‘data’ en la ruta:

   $SOLR_INSTALL\example\pruebadfr\data

3. Crear un nuevo subdirectorio ‘data’ COMPLETAMENTE VACÍO en:

   $SOLR_INSTALL\example\pruebadfr\data

4. Hacer clic en ‘Add Core’ y rellenar el formulario como sigue:

Name: pruebadfr instanceDir: $SOLR_INSTALL\example\pruebadfr dataDir: $SOLR_INSTALL\example\pruebadfr\data config: $SOLR_INSTALL\example\pruebadfr\conf\solrconfig.xml schema: $SOLR_INSTALL\example\pruebadfr\conf\managed-schema.xml

5. Reindexar la colección/core ‘pruebadfr’ conforme al nuevo archivo de configuración. Para ello, se puede consultar el epígrafe ‘Indexación de documentos’.

EJERCICIO 15

Realizar las mismas búsquedas de los ejercicios 10, 11, 12, 13 y 14 con los valores indicados anteriormente para los parámetros del modelo DFR.

SOLUCIÓN EJERCICIO 15

obsolescence (11 documentos)

1. Art12 (2.5696652)
2. Art6 (2.1388435)
3. Art5 (2.069067)
4. Art8 (1.9504864)
5. Art20 (1.7083547)

+world +wide +web (2 documentos)

1. Art3 (7.339361)
2. Art4 (7.3312244)

-age (6 documentos)

1. art10 (1.0)
2. art11 (1.0)
3. art15 (1.0)
4. art16 (1.0)
5. art17 (1.0)

music citation (12 documentos)

1. art5 (6.194088)
2. art2 (2.152527)
3. art19 (1.9449873)
4. art7 (1.8756512)
5. art8 (1.6718856)

“random indexing” (1 documento)

1. art16 (6.9870324)

Por último, compararemos los resultados obtenidos con este modelo DFR de recuperación, introduciendo otros de los valores más habitualmente empleados en los parámetros del modelo.

Para ello, emplearemos una nueva colección/core ‘pruebadfr2’, siguiendo los pasos indicados en los epígrafes ‘Creación de una colección’ e ‘Indexación de documentos’.

Una vez creada la colección/core ‘puebadfr2’, cambiaremos los parámetros correspondientes en el archivo:

$SOLR_INSTALL\example\pruebadfr2\conf\managed-schema.xml

Modificamos, pues, el final de dicho archivo con estos nuevos valores (es otra de las opciones más frecuentes):

```
<similarity   class=”solr.DFRSimilarityFactory”>
        <str name=”basicModel” >I(F)</str>
        <str name=”afterEffect”>B</str>
        <str name=”normalization”>H2</str>
</similarity>                                    [[esta opción no incluye el parámetro c]]
```

Una vez guardados los cambios efectuados en el archivo ‘managed-schema.xml’, debemos:

1. Eliminar la indexación previa realizada. Para ello, seleccionar ‘Core Admin’ en el menú izquierdo, seleccionar luego la colección/core ‘pruebadfr’ y hacer clic en ‘Unload’.
2. Borrar el subdirectorio COMPLETO ‘data’ en la ruta:

   $SOLR_INSTALL\example\pruebadfr2\data

3. Crear un nuevo subdirectorio ‘data’ COMPLETAMENTE VACÍO en:

   $SOLR_INSTALL\example\pruebadfr2\data

4. Hacer clic en ‘Add Core’ y rellenar el formulario como sigue:

   ```
   Name: pruebadfr2
   instanceDir: $SOLR_INSTALL\example\pruebadfr2
   dataDir: $SOLR_INSTALL\example\pruebadfr2\data
   config: $SOLR_INSTALL\example\pruebadfr2\conf\solrconfig.xml
   schema: $SOLR_INSTALL\example\pruebadfr2\conf\managed-schema.xml
   ```

5. Reindexar la colección/core ‘pruebadfr2’ conforme al nuevo archivo de configuración. Para ello, se puede consultar el epígrafe ‘Indexación de documentos’.

Para observar los efectos de los nuevos valores de los parámetros en el modelo DFR, podemos repetir las mismas búsquedas de los ejercicios 10, 11, 12, 13 y 14.

EJERCICIO 16

Realizar las mismas búsquedas de los ejercicios 10, 11, 12, 13 y 14 con los nuevos valores de los parámetros en el modelo DFR.

SOLUCIÓN EJERCICIO 16

obsolescence (11 documentos)

1. Art12 (2.4362729)
2. Art6 (2.4346235)
3. Art8 (2.4220448)
4. Art5 (2.4219515)
5. Art20 (2.4130495)

+world +wide +web (2 documentos)

1. Art4 (7.907995) 2. Art3(7.856372)

-age (6 documentos)

1. art10 (1.0)
2. art11 (1.0)
3. art15 (1.0)
4. art16 (1.0)
5. art17 (1.0)

music citation (12 documentos)

1. art5 (11.8688)
2. art3 (8.86081)
3. art2 (2.4717107)
4. art7 (2.4685993)
5. art19 (2.4673855)

“random indexing” (1 documento)

1. art16 (4.5451574)

INDRI

Instalación, apertura y cierre de Indri en Windows

En este taller emplearemos la última versión disponible del programa Indri en el momento de escribir estas líneas, en concreto la versión 5.12, que puede descargarse desde la página https://sourceforge.net/projects/lemur/ , haciendo clic sucesivamente en 'Files', 'lemur', 'indri 5.12', y finalmente en 'Indri-5.12-win64-install.exe'. En el directorio por defecto de descargas observaremos que se ha descargado el fichero 'Indri-5.12-win64-install.exe'.

Una vez allí, basta hacer doble clic en el archivo 'Indri-5.12-win64-install.exe'. Pulsar 'Aceptar' en la ventana de licencia de uso, y seleccionar posteriormente 'Base' en el tipo de instalación. A continuación, elegir el directorio de instalación por defecto. En nuestro caso, elegiremos el directorio 'Indri' que hemos creado previamente en el Escritorio/Desktop, por lo que en el 'Destination Folder' pondremos:

C:\Users\Juan\Desktop\Indri

Hacemos clic en el botón 'Install'. El programa Indri no necesita más tareas para su instalación.

En adelante denominaremos $INDRI_INSTALL a la ruta donde hayamos situado la carpeta 'Indri' con el programa; en nuestro caso, la ruta hasta llegar a la carpeta 'Indri'. Por tanto, de ahora en adelante:

En Windows, $INDRI_INSTALL es equivalente a

C:\Users\juan\Desktop\Indri

Antes de empezar a utilizar el programa, es IMPRESCINDIBLE que el usuario que utiliza el programa Indri tenga CONTROL TOTAL; ESTO ES, EL USUARIO QUE UTILIZARÁ EL PROGRAMA DEBE TENER TODOS LOS PERMISOS DE ESCRITURA, CREACIÓN, ELIMINACIÓN Y EJECUCIÓN, NO SOLO DE LECTURA, SOBRE EL DIRECTORIO DE INSTALACIÓN 'Indri' Y TODOS LOS SUBDIRECTORIOS Y ARCHIVOS BAJO ÉL.

La propiedad y el control total sobre el subdirectorio 'Indri' puede conseguirse de dos maneras esencialmente: a través de la interfaz gráfica y a través de comandos.

La propiedad y el control total de la carpeta 'Indri' a través de la interfaz gráfica se consigue haciendo clic con el botón derecho en el directorio/carpeta 'Indri'. Seleccionar 'Propiedades' y a continuación elegir el apartado 'Seguridad'. Dentro de él, hacer clic en 'Opciones avanzadas' y luego en 'Propietario'. Comprobar que el propietario que figura es el usuario que va a utilizar el programa; en nuestro caso, 'JUAN-HP\juan'. Si no lo es, pulsar en 'Editar' para modificarlo. A su vez, dentro de 'Opciones avanzadas', junto a 'Propietario' figura el apartado 'Permisos efectivos'. Al hacer clic en 'Seleccionar' el 'Nombre de grupo o de usuario', en el área de texto debemos teclear el mismo usuario propietario; en nuestro caso, 'JUAN-HP\juan'. Aparecerán en pantalla los permisos para dicho usuario. Si no están marcados, deberán marcarse todas las opciones que se muestren en pantalla. Finalmente, hacer clic sucesivamente en 'Aceptar' hasta cerrar todas las pantallas abiertas. Debe advertirse que este proceso debe repetirse con cada uno de los subdirectorios y archivos dentro de la carpeta 'Indri'.

La propiedad y el control total de la carpeta 'Indri' a través de comandos puede conseguirse ejecutando en un TERMINAL, CONSOLA o LÍNEA DE COMANDOS (para la apertura de un terminal o línea de comandos, ver el apartado 'Apertura de una consola, terminal o línea de comandos') los siguientes comandos que figuran en negrita (debe ponerse especial cuidado en introducir los espacios correctamente):

takeown /f C:\Users\juan\Desktop\Indri* /r

[[Este comando realiza el cambio de propietario]]

cd C:\Windows\System32\es-ES

[[Nos vamos al directorio donde se encuentre el comando 'icacls']]

icacls C:\Users\juan\Desktop\Indri /grant JUAN-HP\juan:F /t

[[Este comando otorga control total a la carpeta 'Indri' de manera recursiva]]

[[NOTA BENE: Si diese error introduciendo el comando 'icacls' de esta manera, pruebe a introducir el nombre del propietario por su abreviatura, tal como figura en el apartado 'Propietario'. En nuestro caso, se introduciría 'juan' en lugar de 'JUAN-HP\juan']]

Indexación de una colección

Para poder utilizar el motor de búsqueda Indri se debe cumplir una condición previa, la existencia de una colección de documentos sobre la que recuperar.

Para este taller elegiremos una colección de prueba 'WebAP', guardada en un único fichero, que ponen a nuestra disposición los desarrolladores del buscador Indri. Para ello, ir a la dirección:

https://ciir.cs.umass.edu/downloads/WebAP/index.html

Y hacer clic en el archivo 'WebAP tar gzip archive'. Una vez descargado en el directorio de nuestra preferencia (en nuestro caso, en el mismo Escritorio/Desktop), lo descomprimimos en ese mismo directorio hasta obtener una carpeta denominada 'WebAP', que consta de un archivo 'README' y de una carpeta 'gradedText' que consta, a su vez, de dos archivos con una colección de consultas (ficheros 'gov2.queriesAllRawFile' y 'gov2.query.json') y un único archivo con toda la colección de documentos (fichero 'grade.trectext_patched').

No debemos crear previamente el subdirectorio donde acoger el fichero inverso de la colección, pues el programa lo hará por nosotros al indexar la colección. En cualquier caso, sí debemos tener pensado el nombre y localización de dicho subdirectorio/carpeta.

La creación del fichero inverso/índice de la colección se puede llevar a cabo por dos procedimientos: por línea de comandos y por interfaz gráfica. Nosotros aquí lo haremos por línea de comandos, por ser el procedimiento más habitual.

Como hemos dicho anteriormente, decidimos primeramente el nombre y localización del subdirectorio donde se guardará el fichero inverso. En nuestro caso, optamos por una carpeta denominada 'salida1' bajo el directorio 'Indri'. En este subdirectorio el programa almacenará el índice correspondiente a la colección de documentos elegida previamente (WebAP). Pero **NO LO CREAMOS PREVIAMENTE**, pues el programa lo creará en el momento de la indexación.

En el caso de indexación por línea de comandos, es imprescindible crear un fichero XML que incluya los parámetros que deben utilizarse en dicho proceso, básicamente los dos indicados anteriormente: la colección que va a emplearse y dónde almacenar el índice/fichero inverso.

Para ello, creamos el siguiente archivo (que posteriormente denominaremos 'parameter1.txt') en un documento de texto **SIN FORMATO ALGUNO** (Con el bloc de notas, por ejemplo):

```
<parameters>
        <memory>512m</memory>
        <index>C:/Users/juan/Desktop/Indri/salida1/</index>

        <corpus>
<path>C:\Users\juan\Desktop\WebAP\gradedText\grade.trectext_patched</path>
                <class>trectext</class>
        </corpus>

        <field><name>docno</name></field>
</parameters>
```

[[NOTA BENE: Es importante NO INTRODUCIR NINGÚN ESPACIO dentro de las líneas de texto en este archivo. Por tanto, los caracteres deben ir seguidos desde el carácter inicial '<' hasta el carácter final '>' DENTRO DE CADA UNA DE LAS LÍNEAS]]

Con este archivo efectuamos una indexación por defecto que no incluye ningún proceso en el análisis (ni eliminación de palabras vacías ni stemming o reducción morfológica). Posteriormente veremos cómo incluir dichos procesos en la indexación.

Una vez creado el archivo, lo guardamos con el nombre 'parameter1.txt' en un directorio de nuestra elección. En nuestro caso, lo guardamos en

C:\Users\juan\Desktop\parameter1.txt

A continuación abrimos una consola o terminal (para la apertura de una consola, terminal o línea de comandos, ver el apartado 'Apertura de una consola, terminal o línea de comandos'), tecleando los siguientes comandos (que figuran en negrita):

cd $INDRI_INSTALL\bin

[[en nuestro caso: 'C:\Users\juan\Desktop\Indri\bin']]

IndriBuildIndex C:\Users\juan\Desktop\parameter1.txt

El sistema responderá con un mensaje semejante al siguiente:

0:00: Created repository C:/Users/juan/Desktop/Indri/salida1/

Adding docno to trectext as an indexed field

Adding docno to trectext as an included tag

0:00: Opened C:\Users\juan\Desktop\WebAP\gradedText\grade.trectext_patched

0:32: Documents parsed: …… Documents indexed: …….

0:32: Closed C:\Users\juan\Desktop\WebAP\gradedText\grade.trectext_patched

0:32: Closing index

0:34: Finished

EJERCICIO 1

Tras indexar la colección, responda a las siguientes preguntas:

1. ¿Cuántos documentos han sido indexados?
2. ¿Es compatible ese número con el hecho de que la colección conste de un único fichero?

SOLUCIÓN EJERCICIO 1

1. ¿Cuántos documentos han sido indexados?
   6399 documentos
2. ¿Es compatible ese número con el hecho de que la colección conste de un único fichero?
   Sí, es compatible, porque en ocasiones los documentos se formatean como líneas dentro de un único fichero que contiene toda la colección.

Eliminación de palabras vacías

La indexación de la colección 'WebAP' efectuada anteriormente incluía una tarea de análisis básica que constaba únicamente de un proceso de tokenización (o de segmentación de la cadena de caracteres de cada documento en 'palabras' o 'tokens'), sin someter los documentos a un proceso de eliminación de palabras vacías ni de reducción morfológica o stemming (entre otros posibles, como la reducción de todos los caracteres a minúsculas o la eliminación de acentos, por ejemplo).

En este apartado veremos cómo modificar la indexación de esta colección de manera que efectúe un proceso de eliminación de palabras vacías. El sistema Indri permite incorporar palabras vacías al análisis de los documentos de la colección mediante el parámetro 'stopper' en el fichero con los parámetros (fichero que hemos denominado anteriormente 'parameter1.txt', situado en nuestro caso en la carpeta 'C:\Users\juan\Desktop\parameter1.txt'). Con el fin de conservar las sucesivas versiones del fichero con los parámetros, crearemos un nuevo fichero que denominaremos 'parameter2.txt' que incorpore el parámetro 'stopper' a la versión inicial 'parameter1.txt'.

Debe advertirse que Indri no permite por defecto la reunión de todas las palabras vacías en un fichero ajeno e independiente del archivo de configuración. En consecuencia, cada palabra vacía que queramos considerar en el sistema debe ser introducida en una línea dentro del parámetro 'stopper' con el siguiente formato:

<word>palabra</word>

Como ejemplo, si quisiéramos añadir las palabras vacías: a, about, above, oil, industry e history, el archivo 'C:\Users\juan\Desktop\parameter2.txt' quedaría como sigue:

```
<parameters>
        <memory>512m</memory>
        <index>C:/Users/juan/Desktop/Indri/salida2/</index>

        <corpus>
<path>C:\Users\juan\Desktop\WebAP\gradedText\grade.trectext_patched</path>
```

```
            <class>trectext</class>
        </corpus>

        <field><name>docno</name></field>

        <stopper>
            <word>a</word>
            <word>about</word>
            <word>above</word>
            <word>oil</word>
            <word>industry</word>
            <word>history</word>
        </stopper>
</parameters>
```

Se ha modificado el nombre del índice que se creará (ahora se denomina 'salida2', en lugar de 'salida1') para no sobrescribir el índice anterior. Guardamos los cambios efectuados en el archivo y volvemos a realizar la indexación de la colección conforme a esta nueva configuración del análisis. Para ello debe seguirse el proceso explicado en el epígrafe anterior 'Indexación de una colección'.

Una vez indexada la colección, podemos comprobar el funcionamiento del sistema con esta configuración efectuando consultas al mismo. El procedimiento para introducir consultas en Indri se explica con detalle en el epígrafe 'Recuperación de documentos'. Esencialmente debemos tener en cuenta dos aspectos característicos en el sistema Indri:

1.- Debe crearse un fichero XML que incluya la configuración de la recuperación, esto es, los parámetros que deben utilizarse en la recuperación de documentos, semejante al fichero XML de configuración de la indexación.

2.- Las consultas no pueden introducirse de manera independiente, sino que deben incluirse dentro del archivo de configuración de la recuperación.

Creamos, pues, el siguiente archivo (que posteriormente denominaremos 'param_query2.txt') en un documento de texto **SIN FORMATO ALGUNO** (Con el bloc de notas, por ejemplo):

```
<parameters>
        <memory>512m</memory>
        <index>C:/Users/juan/Desktop/Indri/salida2/</index>
        <count>10</count>

        <query>
                <type>indri</type>
                <number>01</number>
                <text>oil industry history</text>
        </query>
</parameters>
```

Dentro de la etiqueta 'query', la etiqueta 'text' incluye la consulta que deseamos hacer al sistema (en el ejemplo, 'oil industry history'). Una vez creado el archivo, lo guardamos con el nombre 'param_query2.txt' en un directorio de nuestra elección. En nuestro caso, lo guardamos en

C:\Users\juan\Desktop\param_query2.txt

Junto a los archivos 'parameter1.txt' y 'parameter2.txt'.

A continuación abrimos una consola o terminal, tecleando los siguientes comandos (que figuran en negrita):

cd $INDRI_INSTALL\bin

[[en nuestro caso: 'C:\Users\juan\Desktop\Indri\bin']]

IndriRunQuery C:\Users\juan\Desktop\param_query2.txt

E**JERCICIO 2**

Ejecutar la consulta ‘oil industry history’ tras la creación del archivo de configuración de la recuperación ‘param_query2.txt’.

1. ¿Devuelve el sistema algún resultado?
2. ¿Cuál cree que es la causa de tal comportamiento?

SOLUCIÓN EJERCICIO 2

1. ¿Devuelve el sistema algún resultado?
   No, no devuelve ningún resultado en relación a ‘oil industry history’
2. ¿Cuál cree que es la causa de tal comportamiento?
   Ello es debido a que todos los términos de la consulta están incluidos como palabras vacías

Stemming o reducción morfológica

Indri incluye la posibilidad de añadir un algoritmo de stemming entre los procesos optativos de la cadena de análisis. Indri permite seleccionar directamente entre dos algoritmos de stemming muy conocidos: Porter y Krovetz (algoritmo menos radical en la reducción morfológica que el de Porter). Para incorporar un proceso de stemming dentro del análisis de los documentos de una colección, debemos añadir dicho proceso en la configuración de la indexación.

Procederemos, pues, a modificar la indexación de la colección 'WebAP' de manera que efectúe:

1.- La eliminación de las siguientes palabras vacías: a, about, above

2.- Un proceso de stemming conforme al algoritmo de stemming de Porter.

Para ello, crearemos el fichero 'parameter3.txt', situado en nuestro caso en la carpeta 'C:\Users\juan\Desktop\parameter3.txt', de la siguiente manera:

```
<parameters>
        <memory>512m</memory>
        <index>C:/Users/juan/Desktop/Indri/salida3/</index>

        <stemmer>
                <name>porter</name>
        </stemmer>

        <corpus>
                <path>C:\Users\juan\Desktop\WebAP\gradedText\grade.trectext_pat
        ched</path>
                <class>trectext</class>
        </corpus>

        <field><name>docno</name></field>
<stopper>
                <word>a</word>
                <word>about</word>
                <word>above</word>
</stopper>
</parameters>
```

Guardamos los cambios efectuados en el archivo 'parameter3.txt' y volvemos a realizar la indexación de la colección conforme a esta nueva configuración del análisis. Para ello debe seguirse el proceso explicado en el epígrafe 'Indexación de una colección'.

Una vez indexada la colección, podemos comprobar el funcionamiento del sistema con esta configuración efectuando consultas al mismo. Creamos, pues, el siguiente archivo en un documento de texto **SIN FORMATO ALGUNO** (Con el bloc de notas, por ejemplo):

```
<parameters>
        <memory>512m</memory>
        <index>C:/Users/juan/Desktop/Indri/salida3/</index>
        <count>10</count>

        <query>
                <type>indri</type>
                <number>02</number>
                <text>oil industry history</text>
        </query>
        <query>
                <type>indri</type>
                <number>03</number>
                <text>oil industrial histories</text>
        </query>
</parameters>
```

Una vez creado el archivo, lo guardamos con el nombre 'param_query3.txt' en un directorio de nuestra elección. En nuestro caso, lo guardamos junto a los demás en

C:\Users\juan\Desktop\param_query3.txt

A continuación abrimos una consola o terminal, tecleando los siguientes comandos (que figuran en negritas):

cd $INDRI_INSTALL\bin [[en nuestro caso: 'C:\Users\juan\Desktop\Indri\bin']]

IndriRunQuery C:\Users\juan\Desktop\param_query3.txt

EJERCICIO 3

Ejecutar las consultas ‘oil industry history’ y ‘oil industrial histories’ tras la creación del archivo de configuración de la recuperación ‘param_query3.txt’.

1. ¿Devuelve el sistema algún resultado en relación a ‘oil industry history’?
2. ¿La respuesta a las consultas: ‘oil industry history’ y ‘oil industrial histories’ es la misma o se observa alguna diferencia entre ellas, tanto en los documentos como en la puntuación –número al principio de cada línea- que obtiene cada documento?

SOLUCIÓN EJERCICIO 3

1. ¿Devuelve el sistema algún resultado en relación a ‘oil industry history’?
   Ahora sí devuelve 10 resultados en relación a la consulta ‘oil industry history’
2. ¿La respuesta a las consultas: ‘oil industry history’ y ‘oil industrial histories’ es la misma o se observa alguna diferencia entre ellas, tanto en los documentos como en la puntuación –número al principio de cada línea- que obtiene cada documento?
   La respuesta a las consultas ‘oil industry history’ y ‘oil industrial histories’ es la misma. En ambos casos el primer documento es GX056-35-8200518-798 (con puntuación -5.33342) y el décimo documento es GX232-43-0102505-701 (con puntuación -5.78349).

Recuperación de documentos

Indri efectúa, por defecto, el cálculo de la respuesta en relación a las consultas de los usuarios mediante una mezcla de un modelo de lenguaje estadístico y un modelo de redes de inferencia (ambos se consideran clases dentro del modelo probabilístico). A raíz de la introducción de una consulta, Indri tratará –a grandes líneas- de estimar la probabilidad de que cada documento de la colección hubiese generado los términos de la consulta. Ello le permite efectuar una ordenación de los documentos en relación a cada consulta.

Indri permite, además, modificar directamente el modelo de recuperación por defecto del sistema, de manera que no se aplique un modelo de lenguaje estadístico. Las posibilidades son las siguientes:

1.- TF-IDF. Es la base del modelo vectorial clásico. Para ello, debe incluirse la línea siguiente en el fichero de configuración de la recuperación:

<baseline>tfidf</baseline>

2.- OKAPI. Es la base de los modelos probabilísticos avanzados. Para ello, debe incluirse la línea siguiente en el fichero de configuración de la recuperación:

<baseline>okapi</baseline>

Una vez indexada la colección 'WebAP' conforme a una determinada configuración –como en el epígrafe 'Stemming o reducción morfológica'-, en principio estamos ya capacitados para realizar búsquedas en dicha colección mediante Indri. Solo es necesario indicarle al programa algunos parámetros esenciales:

1. Cuánta memoria emplear en el proceso de recuperación.
2. Dónde localizar el índice/fichero inverso de la colección.
3. Cuántos documentos debe mostrar en la respuesta a las consultas.
4. Cuál o cuáles son las consultas.
5. Parámetros de análisis de las consultas (palabras vacías, stemming….)

6. Otros parámetros (modelos de recuperación, valor de parámetros de los modelos…)

Todos estos parámetros deben reunirse en un documento XML semejante al desarrollado para la indexación de la colección. Nosotros incluiremos aquí solamente los parámetros principales para observar el funcionamiento del motor de búsqueda.

Creamos, pues, el siguiente archivo en un documento de texto **SIN FORMATO ALGUNO** (Con el bloc de notas, por ejemplo), donde cada línea corresponde a los 4 primeros factores enumerados anteriormente:

```
<parameters>
        <memory>512m</memory>
        <index>C:/Users/juan/Desktop/Indri/salida3/</index>
        <count>10</count>

        <query>
                <type>indri</type>
                <number>04</number>
                <text>oil industry history</text>
        </query>
</parameters>
```

Dentro de la etiqueta ‘query’, la etiqueta ‘text’ incluye la consulta que hacemos al sistema (en el ejemplo, ‘oil industry history’). Una vez creado el archivo, lo guardamos con el nombre ‘param_query4.txt’ en un directorio de nuestra elección. En nuestro caso, lo guardamos en ‘C:\Users\juan\Desktop\param_query4.txt’, junto con el resto de archivos con los parámetros de las indexaciones y recuperaciones realizadas anteriormente.

A continuación abrimos una consola o terminal, tecleando los siguientes comandos (que figuran en negrita):

cd $INDRI_INSTALL\bin [[en nuestro caso: ‘C:\Users\juan\Desktop\Indri\bin’]]

IndriRunQuery C:\Users\juan\Desktop\param_query4.txt

El sistema responderá con el siguiente mensaje:

-5.33342 GX056-35-8200518-798 0 31628

-5.3654 GX025-72-6112588-701 0 1472

……..

-5.78349 GX232-43-0102505-701 0 5842

En cada fila de la respuesta, y de manera ordenada de más a menos relevante, el sistema incluye un documento de la colección. A su vez, en la primera columna se indica la puntuación final obtenida por el documento; en la segunda columna se indica el nombre del documento; en la tercera columna el primer término del documento incluido en la respuesta; por último, en la cuarta columna, se señala el último término del documento incluido en la respuesta.

Búsqueda de un término aislado

Basta introducir el término en la etiqueta 'text', dejar el resto de los valores como figuran arriba, guardar los cambios efectuados en el archivo 'param_query_ejercicio_4.txt' y volver a correr el mismo comando apuntado inicialmente.

EJERCICIO 4

Realizar una búsqueda del término '**dogs'** (sin comillas) en la colección.

1. ¿Qué puntuación obtiene el primer documento de la respuesta?
2. ¿Cuál es el nombre del primer documento de la respuesta?

SOLUCIÓN EJERCICIO 4

1. ¿Qué puntuación obtiene el primer documento de la respuesta?
   -3.81584
2. ¿Cuál es el nombre del primer documento de la respuesta?
   GX038-06-13278660-794

- **Búsqueda por frases**

  Se realiza introduciendo en la etiqueta 'text' la siguiente cadena: #1(frase).

Ejemplo: <text>#1(white house)</text>. El resto de los valores se deja como figuran arriba. Guardar los cambios efectuados en el archivo 'param_query_ejercicio_5.txt' y volver a correr el mismo comando apuntado inicialmente.

EJERCICIO 5

Realizar una búsqueda con la frase ‘public libraries’ (sin comillas).

1. ¿Qué puntuación obtiene el primer documento de la respuesta?
2. ¿Cuál es el nombre del primer documento de la respuesta?

SOLUCIÓN EJERCICIO 5

1. ¿Qué puntuación obtiene el primer documento de la respuesta?
   -2.71782
2. ¿Cuál es el nombre del primer documento de la respuesta?
   GX239-81-4505280-747

- **Búsqueda con AND**

Se realiza introduciendo en la etiqueta ‘text’ la siguiente cadena: #band(t1 t2 …tn).

Ejemplo: <text>#band(white house)</text>. El resto de los valores se deja como figuran arriba. Guardar los cambios efectuados en el archivo ‘param_query_ejercicio_6.txt’ y volver a correr el mismo comando apuntado inicialmente.

EJERCICIO 6

Realizar una búsqueda de modo que me devuelva los documentos en los que aparezcan simultáneamente los términos 'oil', 'industry' e 'history' (sin comillas)

1. ¿Qué puntuación obtiene el primer documento de la respuesta?
2. ¿Cuál es el nombre del primer documento de la respuesta?

SOLUCIÓN EJERCICIO 6

1. ¿Qué puntuación obtiene el primer documento de la respuesta?
   -6.43004
2. ¿Cuál es el nombre del primer documento de la respuesta?
   GX068-83-6288039-701

- **Búsqueda con OR**

Se realiza introduciendo en la etiqueta 'text' los términos entre espacios (es el operador por defecto).

Ejemplo: <text>white house</text>. El resto de los valores se deja como figuran arriba. También es posible realizar este tipo de consulta introduciendo en la etiqueta 'text' la siguiente cadena: #combine(t1 t2 … tn).

Ejemplo: <text>#combine(white house)</text>. El resto de los valores se deja como figuran arriba. Guardar los cambios efectuados en el archivo 'param_query_ejercicio_7.txt' y volver a correr el mismo comando apuntado inicialmente.

EJERCICIO 7

Realizar una búsqueda con los documentos en los que aparezcan al menos una de las siguientes palabras: 'recycling' y 'projects' (sin comillas).

1. ¿Qué puntuación obtiene el primer documento de la respuesta?
2. ¿Cuál es el nombre del primer documento de la respuesta?

SOLUCIÓN EJERCICIO 7

1. ¿Qué puntuación obtiene el primer documento de la respuesta?
   -4.06777
2. ¿Cuál es el nombre del primer documento de la respuesta?
   GX232-78-0890114-734

- **Búsqueda de cercanía**

Para recuperar cualquier documento donde el primer término aparezca antes que el segundo término introducido, existiendo entre ellos un máximo de 'n' términos, basta introducir en la etiqueta <text> la siguiente cadena: #n(t1 t2).

Ejemplo: <text>#5(white house)</text>. El resto de los valores se deja como figuran arriba. Guardar los cambios efectuados en el archivo 'param_query_ejercicio_8.txt' y volver a correr el mismo comando apuntado inicialmente.

EJERCICIO 8

Realizar una búsqueda con los documentos que contengan los términos ‘artificial’ e ‘intelligence’, en ese orden y con 4 términos entre ellos como máximo.

1. ¿Qué puntuación obtiene el primer documento de la respuesta?
2. ¿Cuál es el nombre del primer documento de la respuesta?

SOLUCIÓN EJERCICIO 8

1. ¿Qué puntuación obtiene el primer documento de la respuesta?

   -4.46155
2. ¿Cuál es el nombre del primer documento de la respuesta?

   GX074-74-3524237-741

TERRIER

Prerrequisitos de Terrier en Windows

Terrier es un programa de código abierto de búsqueda y recuperación de información que exige para su funcionamiento la instalación previa de Java (JRE) versión 1.8 o superior en nuestro sistema. Por tanto, debemos asegurarnos de tener instalado Java en nuestro ordenador antes de instalar el programa Terrier. Para comprobar si Java (JRE) está instalado en Windows, hacemos clic en 'Inicio' (el icono de Windows en el margen inferior izquierdo de la pantalla), luego en 'Panel de control' y posteriormente en 'Programas':

Inicio > Panel de control > Programas

Si no apareciese el programa 'Java' en 'Inicio > Panel de control > Programas', podemos hacer otra comprobación previa en la siguiente ruta:

Inicio > Panel de control > Programas > Programas y características

Allí debe incluirse una entrada del tipo 'Java 8 Update 131 (64-bit)'. Esta entrada, por ejemplo, corrobora que en nuestro ordenador está instalada la versión '1.8.131' del programa Java. Si no disponemos en nuestro ordenador del programa Java (JRE), debemos instalarlo (ir al apéndice 'Instalación de Java (JRE) en Windows', más adelante en este mismo documento).

No solamente es necesario tener instalado el programa Java (JRE). De igual forma, debemos asegurarnos de que la versión instalada en nuestro sistema es la 1.8 o superior (como en este caso, la versión 1.8.131).

Si ya disponemos del programa, en 'Programas' debería aparecer 'Java' junto a su icono (una taza de café). Si es así, comprobaremos la versión de Java instalada. Para ello, hacemos clic en 'Java'. En la pantalla que aparece, hacemos clic en la pestaña 'General' y a continuación hacemos clic en el botón 'Acerca de'. En la pantalla resultante figurará un mensaje del tipo: 'Versión 8 Actualización 131'. Ello indica que tenemos

instalada la versión ‘1.8.131’. Debemos asegurarnos de que la versión instalada en nuestro sistema es la 1.8 o superior (como en este caso, la versión 1.8.131). Si no disponemos en nuestro ordenador de una versión 1.8 o superior, debemos instalarla (ir al apéndice ‘Instalación de Java (JRE) en Windows’, más adelante en este mismo documento).

También podemos comprobar qué versión de Java tenemos instalada en nuestro ordenador abriendo una CONSOLA, TERMINAL O LÍNEA DE COMANDOS. Para ello, basta ir a Inicio (el icono en el margen inferior izquierdo de la pantalla con la bandera de Windows), y en el buscador que aparece en la parte inferior de la pantalla con la frase ‘Buscar programas y archivos’ introducimos ‘cmd’ [sin las comillas simples; solamente las letras cmd]. Aparecerá una pantalla con fondo negro y una última línea semejante a:

C:\Users\Juan>_

Hasta el signo > el sistema nos está indicando en qué subdirectorio nos encontramos (en este caso, en el subdirectorio ‘Juan’, dentro del directorio ‘Users’, dentro de la unidad ‘C’ de nuestro ordenador). El prompt (_) está parpadeando en espera de que introduzcamos un comando (de donde la abreviatura ‘cmd’ introducida anteriormente). Hemos abierto, pues, una CONSOLA, TERMINAL O LÍNEA DE COMANDOS, que nos permite introducir directamente comandos al sistema para su ejecución. Si introducimos el siguiente comando:

java -version

el sistema nos devolverá un mensaje como el siguiente, siempre que tengamos instalado Java:

java version ‘1.8.0_131’

Java <TM> SE Runtime Environment <build 1.8.0_131-b01>

Java HotSpot <TM> 64-Bit Server VM <build 25.144-b01, mixed mode>

Si no está instalado el programa Java (JRE) o la versión instalada es inferior a la 1.8, procederemos a instalar Java (JRE) antes de proceder a instalar el programa Terrier, siguiendo para ello los pasos que se indican en el apartado 'Instalación de Java (JRE) en Windows'.

Si disponemos de una versión adecuada de Java (JRE) en nuestro sistema, continuaremos directamente con el apartado 'Instalación, apertura y cierre del programa Terrier en Windows'.

Instalación, apertura y cierre del programa Terrier en Windows

En este taller emplearemos la última versión disponible del programa Terrier, en concreto la versión 4.2, que puede descargarse de la página oficial del programa, http://terrier.org/ , haciendo clic en el menú 'Download' y seleccionando el archivo 'terrier-core-4.2-bin.zip', versión disponible para el sistema operativo Windows.

Una vez descargado el archivo, movemos el archivo recién descargado al directorio donde queramos instalar el programa; en nuestro caso, movemos el archivo al Escritorio/Desktop. Allí lo descomprimimos, resultando en nuestro caso el archivo 'C:\Users\juan\Desktop\terrier-core-4.2'.

En adelante denominaremos $TERRIER_INSTALL a la ruta donde hayamos situado la carpeta con el programa; en nuestro caso, la ruta hasta llegar a la carpeta 'terrier-core-4.2'. Por tanto, de ahora en adelante, en Windows,

$TERRIER_INSTALL es equivalente a

C:\Users\juan\Desktop\terrier-core-4.2

Ahora debemos crear una variable de entorno TERRIER_HOME para el usuario del programa (en nuestro caso, 'juan'), cuyo valor debe ser $TERRIER_INSTALL. Para ello, en Windows, ir sucesivamente a 'Panel de control', 'Sistema y seguridad', 'Sistema', 'Configuración avanzada del sistema', 'Variables de entorno', y allí crear una 'Nueva' variable de entorno de usuario, no del sistema. En nuestro caso, el valor impuesto a la variable de entorno TERRIER_HOME es

C:\Users\juan\Desktop\terrier-core-4.2

A continuación, tenemos que copiar el archivo

$TERRIER_INSTALL\etc\terrier.properties.sample

en el mismo subdirectorio ‘etc’ pero con la denominación ‘terrier.properties’ (esto es, eliminando la extensión ‘sample’). Para ello, basta teclear en una consola o terminal (para abrir una consola o terminal, puede consultarse el apéndice ‘Apertura de una consola, terminal o línea de comandos’ en este mismo documento) el siguiente comando (que figura en negritas):

cp C:\Users\juan\Desktop\terrier-core-4.2\etc\terrier.properties.sample C:\Users\juan\Desktop\terrier-core-4.2\etc\terrier.properties

Es conveniente comprobar que en el subdirectorio $TERRIER_INSTALL\etc tenemos un archivo ‘terrier.properties’, pues este archivo es esencial para el funcionamiento del programa Terrier.

A partir de este momento, el programa está ya instalado. Para comprobar que todo ha ido bien, en un TERMINAL o CONSOLA o LÍNEA DE COMANDOS (para abrir un terminal, puede consultarse el apéndice ‘Apertura de una consola, terminal o línea de comandos’ en este mismo documento) ejecutamos los siguientes comandos (que figuran en negritas):

cd $TERRIER_INSTALL [[En nuestro caso, C:\Users\juan\Desktop\terrier-core-4.2]]
bin\interactive_terrier.bat

Veremos el siguiente mensaje en pantalla:

> 00:00:00 [main] ERROR o.t.applications. InteractiveQuerying – Failed to load index. Perhaps index files are missing
>
> 00:00:00 [main] INFO o.t.applications.InteractiveQuerying – time to initialise index: 0.042
>
> 00:00:00 [main] ERROR o.t.applications.InteractiveQuerying – Problem loading Manager (org.terrier.querying.Manager)
>
> ……..
>
> Please enter your query:

Estos mensajes de error son debidos a que no disponemos de una colección con la que trabajar. Es necesario, pues, indexar una colección en el sistema.

Indexación de una colección

Partiremos de la colección 'colwt2g', una selección de unos 17000 documentos de la colección 'WT2G' empleada en las conferencias TREC. Dicha colección se encuentra comprimida en el Campus Virtual de la asignatura, junto al texto del taller del sistema Terrier. Si no se tiene acceso a la colección en el Campus Virtual, se puede acceder a ella en la dirección de la Universidad de Glasgow:

http://ir.dcs.gla.ac.uk/test_collections/access_to_data.html

Descargamos el directorio, lo descomprimimos y lo guardamos en un subdirectorio de nuestra elección. En nuestro caso, 'C:\Users\juan\Desktop\colwt2g'.

Procedemos a indicar a Terrier que debe utilizar la colección recién incorporada y preparamos el programa para su indexación. Para ello, en un terminal (para abrir una consola o terminal, puede consultarse el apéndice 'Apertura de una consola, terminal o línea de comandos' en este mismo documento) se deben ejecutar los siguientes comandos (que figuran en negritas):

cd $TERRIER_INSTALL [[En nuestro caso, C:\Users\juan\Desktop\terrier-core-4.2]]
bin\trec_setup.bat C:\Users\juan\Desktop\colwt2g

En ocasiones, al introducir el segundo comando, se recibe un mensaje de error del tipo 'The system cannot find the path specified', o su versión en español, 'El sistema no puede encontrar la ruta especificada'. En tal caso, consultar en apéndices el apartado 'Error 'The system cannot find the path…' en este mismo documento, donde se indican los pasos para solucionar el problema.

El archivo collection.spec debe contener exclusivamente un listado de todos los archivos que componen la colección que se indexará en su orden. Para comprobarlo, se puede abrir con WordPad el archivo '$TERRIER_INSTALL\etc\collection.spec'. En caso necesario, se pueden eliminar los archivos que no deseemos que sean indexados. Si se ha modificado el archivo, no olvidar guardar los cambios introducidos.

Antes de indexar la colección, debemos asegurarnos de borrar cualquier indexación anterior, eliminando el contenido del subdirectorio

$TERRIER_INSTALL\var\index

Pero manteniendo el subdirectorio 'index' (que debe estar, por tanto, completamente vacío).

Ahora estamos en condiciones de realizar una indexación de la colección (denominada de doble pasada en Terrier) con una configuración inicial que no incluye ni eliminación de palabras vacías ni stemming o reducción morfológica. Ello se consigue tecleando en un terminal (para abrir una consola o terminal, puede consultarse el apéndice 'Apertura de una consola, terminal o línea de comandos' en este mismo documento) los siguientes comandos (que figuran en negritas):

cd $TERRIER_INSTALL [[En nuestro caso, C:\Users\juan\Desktop\terrier-core-4.2]]
bin\trec_terrier.bat -i -Dtermpipelines=

Una vez indexada la colección, se pueden consultar las estadísticas de la colección introduciendo en un terminal (para abrir una consola o terminal, puede consultarse el apéndice 'Apertura de una consola, terminal o línea de comandos' en este mismo documento) los siguientes comandos (que figuran en negritas):

cd $TERRIER_INSTALL [[En nuestro caso, C:\Users\juan\Desktop\terrier-core-4.2]]
bin\trec_terrier.bat --printstats

EJERCICIO 1

Tras indexar la colección ‘colwt2g’, responde a las siguientes preguntas:

1. ¿Qué número de documentos ha sido indexado?
2. ¿Qué número de términos distintos –vocabulary- se ha encontrado?
3. ¿Qué número de tokens se ha localizado en la colección?

SOLUCIÓN EJERCICIO 1

1. ¿Qué número de documentos ha sido indexado?

   Nº of indexed documents: 17073
2. ¿Qué número de términos distintos –vocabulary- se ha encontrado?

   Size of vocabulary: 175920
3. ¿Qué número de tokens se ha localizado en la colección?

   Nº of tokens: 9851528

Eliminación de palabras vacías

La indexación de la colección 'colwt2g' efectuada anteriormente incluía una tarea de análisis básica que constaba únicamente de un proceso de tokenización (o de segmentación de la cadena de caracteres de cada documento en 'palabras' o 'tokens'), sin someter los documentos a un proceso de eliminación de palabras vacías ni de reducción morfológica o stemming (en otros posibles, como la reducción de todos los caracteres a minúsculas o la eliminación de acentos, por ejemplo).

En este apartado veremos cómo modificar la indexación de esta colección de manera que efectúe un proceso de eliminación de palabras vacías conforme a un listado recogido en el fichero '$TERRIER_INSTALL\share\stopword-list.txt', que es el listado por defecto que emplea Terrier.

Para ello debemos seguir los siguientes pasos:

1.- Eliminamos primeramente la indexación previa realizada con la colección 'colwt2g'. Ello se consigue eliminando el contenido del subdirectorio

$TERRIER_INSTALL\var\index

Pero manteniendo el subdirectorio 'index' (que debe estar, por tanto, completamente vacío).

2.- Indexamos de nuevo la colección 'colwt2g' imponiendo un proceso de eliminación de palabras vacías en el análisis de los documentos. En un terminal (para abrir una consola o terminal, puede consultarse el apéndice 'Apertura de una consola, terminal o línea de comandos' en este mismo documento) se deben ejecutar los siguientes comandos (que figuran en negritas):

cd $TERRIER_INSTALL [[En nuestro caso, C:\Users\juan\Desktop\terrier-core-4.2]]
bin\trec_terrier.bat -i -Dtermpipelines=Stopwords

Una vez indexada la colección, se puede observar la reducción que este proceso provoca en el número de términos –vocabulary- y en el número de tokens consultando las estadísticas de la colección en un terminal (para abrir una consola o terminal, puede consultarse el apéndice 'Apertura de una consola, terminal o línea de comandos' en este mismo documento) mediante los comandos (que figuran en negritas):

cd $TERRIER_INSTALL [[En nuestro caso, C:\Users\juan\Desktop\terrier-core-4.2]]
bin\trec_terrier.bat --printstats

EJERCICIO 2

Tras indexar la colección 'colwt2g' con un proceso de eliminación de palabras vacías, responde a las siguientes preguntas:

1. ¿Qué número de documentos ha sido indexado?
2. ¿Qué número de términos distintos –vocabulary- se ha encontrado?
3. ¿Qué número de tokens se ha localizado en la colección?

SOLUCIÓN EJERCICIO 2

1. ¿Qué número de documentos ha sido indexado?
   Nº of indexed documents: 17073
2. ¿Qué número de términos distintos –vocabulary- se ha encontrado?
   Size of vocabulary: 175402
3. ¿Qué número de tokens se ha localizado en la colección?
   Nº of tokens: 6108624

Antes de realizar consultas, debemos ser conscientes de que Terrier devuelve, en relación a cada consulta, los primeros 1000 resultados más relevantes. Para restringir

este número, de manera que se puedan observar los primeros resultados en el terminal, debemos modificar el archivo

$TERRIER_INSTALL\etc\terrier.properties

Para ello abrimos el archivo 'terrier.properties' con el programa Wordpad, y al final del archivo añadimos las siguientes líneas:

```
#retrieved set size
matching.retrieved_set_size=10
```

Una vez indexada la colección y limitado el número de documentos en la respuesta, podemos comprobar el funcionamiento del sistema con esta configuración efectuando consultas al mismo. Para ello, en un terminal (para abrir una consola o terminal, puede consultarse el apéndice 'Apertura de una consola, terminal o línea de comandos' en este mismo documento) teclear los siguientes comandos (que figuran en negritas):

cd $TERRIER_INSTALL [[En nuestro caso, C:\Users\juan\Desktop\terrier-core-4.2]]
bin\interactive_terrier.bat

El sistema responderá con un mensaje que incluye al final:

Please enter your query:

EJERCICIO 3

Tras indexar la colección ‘colwt2g’ con un proceso de eliminación de palabras vacías, responde a las siguientes preguntas:

1. ¿Qué respuesta da el sistema si realizamos la siguiente consulta: of?
2. ¿La respuesta a las consultas: ‘security of airports’ [sin comillas] y ‘airports security’ [sin comillas] es la misma o se observa alguna diferencia entre ellas, tanto en los documentos como en la puntuación –número al final de cada línea- que obtiene cada documento?

SOLUCIÓN EJERCICIO 3

1. ¿Qué respuesta da el sistema si realizamos la siguiente consulta: of?

   00:00:00 [main] WARN org.terrier.querying.Manager

   -Returning empty results as query 1 is empty
2. ¿La respuesta a las consultas: ‘security of airports’ [sin comillas] y ‘airports security’ [sin comillas] es la misma o se observa alguna diferencia entre ellas, tanto en los documentos como en la puntuación –número al final de cada línea- que obtiene cada documento?

   La respuesta es la misma, sin diferencias en el número de documentos y en la puntuación obtenida:

   0 WTX001-B03-118 7.75892

   ……..

   9 WTX001-B07-97 5.69755

Stemming o reducción morfológica

Terrier incluye la posibilidad de incluir el algoritmo de stemming de Porter entre los procesos optativos de la cadena de análisis. Para incorporar dicho proceso de stemming dentro del análisis de los documentos de una colección, debemos añadir dicho proceso en la configuración del programa.

Procederemos, pues, a modificar la indexación de la colección 'colwt2g' de manera que efectúe un proceso de eliminación de palabras vacías conforme a un listado recogido en el fichero '$TERRIER_INSTALL\share\stopword-list.txt' (que es el listado por defecto que emplea Terrier), y además un proceso de stemming conforme al algoritmo de stemming de Porter (de igual forma, el algoritmo de stemming por defecto en Terrier).

Para ello debemos seguir los siguientes pasos:

1.- Eliminamos primeramente la indexación previa realizada con la colección 'colwt2g'. Ello se consigue eliminando el contenido del subdirectorio

$TERRIER_INSTALL\var\index

Pero manteniendo el subdirectorio 'index' (que debe estar, por tanto, completamente vacío).

2.- Indexamos de nuevo la colección 'colwt2g' imponiendo un proceso de eliminación de palabras vacías y un proceso de stemming (algoritmo de Porter) en el análisis de los documentos. En un terminal (para abrir una consola o terminal, puede consultarse el apéndice 'Apertura de una consola, terminal o línea de comandos' en este mismo documento) tecleamos los siguientes comandos (que figuran en negritas):

cd $TERRIER_INSTALL [[En nuestro caso, C:\Users\juan\Desktop\terrier-core-4.2]]
bin\trec_terrier.bat -i -Dtermpipelines=Stopwords,PorterStemmer

Una vez indexada la colección, se puede observar la reducción que este proceso provoca en el número de términos –vocabulary- (aunque ya no en el número de tokens en relación a la indexación con palabras vacías) consultando las estadísticas de la colección en un terminal (para abrir una consola o terminal, puede consultarse el apéndice 'Apertura de una consola, terminal o línea de comandos' en este mismo documento) con los siguientes comandos (que figuran en negritas):

cd $TERRIER_INSTALL [[En nuestro caso, C:\Users\juan\Desktop\terrier-core-4.2]]
bin\trec_terrier.bat --printstats

EJERCICIO 4

Tras indexar la colección 'colwt2g' con procesos de eliminación de palabras vacías y de reducción morfológica o stemming, responde a las siguientes preguntas:

1. ¿Qué número de documentos ha sido indexados?
2. ¿Qué número de términos distintos –vocabulary- se ha encontrado?
3. ¿Qué número de tokens se ha localizado en la colección?

SOLUCIÓN EJERCICIO 4

1. ¿Qué número de documentos ha sido indexados?
   Nº of indexed documents: 17073
2. ¿Qué número de términos distintos –vocabulary- se ha encontrado?
   Size of vocabulary: 140456
3. ¿Qué número de tokens se ha localizado en la colección?
   Nº of tokens: 6108624

A continuación podemos comprobar el funcionamiento del sistema con esta configuración efectuando consultas al mismo. Para ello, tecleamos los siguientes comandos (que figuran en negritas) en un terminal (para abrir una consola o terminal, puede consultarse el apéndice 'Apertura de una consola, terminal o línea de comandos' en este mismo documento):

cd $TERRIER_INSTALL [[En nuestro caso, C:\Users\juan\Desktop\terrier-core-4.2]]
bin\interactive_terrier.bat

El sistema responderá con un mensaje que incluye al final:

Please enter your query:

EJERCICIO 5

Tras indexar la colección 'colwt2g' con un proceso de eliminación de palabras vacías y un proceso de stemming o reducción morfológica, responde a las siguientes preguntas:

1. ¿La respuesta a las consultas: 'decline birth rate' [sin comillas] y 'declining birth rates' [sin comillas] es la misma o se observa alguna diferencia entre ellas, tanto en los documentos como en la puntuación –número al final de cada línea- que obtiene cada documento?
2. Conforme a este resultado, ¿a qué clase o clases de morfemas afecta el algoritmo de stemming de Porter?

SOLUCIÓN EJERCICIO 5

1. ¿La respuesta a las consultas: 'decline birth rate' [sin comillas] y 'declining birth rates' [sin comillas] es la misma o se observa alguna diferencia entre ellas, tanto en los documentos como en la puntuación –número al final de cada línea- que obtiene cada documento?
   La respuesta es la misma, sin diferencias en los documentos mostrados ni en la puntuación obtenida por cada uno de ellos:
   0 WTX001-B17-95 10.4438
   …….
   9 WTX001-B19-153 6.9800
2. Conforme a este resultado, ¿a qué clase o clases de morfemas afecta el algoritmo de stemming de Porter?
   El algoritmo de Porter afecta a los morfemas de número (singular/plural) y de conjugación verbal (presente/gerundio/….)

Recuperación de documentos

El programa Terrier emplea, por defecto, un modelo de recuperación denominado DFR (Divergence From Randomness), perteneciente a la familia de los modelos probabilísticos.

Es posible en Terrier emplear muy diversos modelos de recuperación de información. Si deseamos modificar el modelo de recuperación por defecto, debemos modificar el archivo

$TERRIER_INSTALL\etc\terrier.properties

Para ello, abrimos el archivo con el programa Wordpad y añadimos al final del archivo las siguientes líneas:

```
#retrieval model
trec.model=LemurTF_IDF
```

Por defecto, Terrier devuelve, en relación a cada consulta, los primeros 1000 documentos de la colección. Para restringir este número, de manera que se puedan consultar los primeros resultados en el terminal, debemos añadir las siguientes líneas al final del archivo '$TERRIER_INSTALL\etc\terrier.properties':

```
#retrieved set size
matching.retrieved_set_size=10
```

En este mismo archivo '$TERRIER_INSTALL\etc\terrier.properties', hacia el final del mismo, se describen los procesos llevados a cabo en el análisis de los documentos de la colección con unas líneas del tipo:

```
#the processing stages a term goes through
termpipelines=……..  [[los procesos separados por comas]]
```

No olvide guardar los cambios introducidos en el archivo 'terrier.properties'.

EJERCICIO 6

Observando el archivo '$TERRIER_INSTALL\etc\terrier.properties', responde a las siguientes preguntas:

1. ¿De cuántos procesos consta el análisis de los documentos por defecto?
2. ¿Se incluye eliminación de palabras vacías?
3. ¿Se incluye stemming?

SOLUCIÓN EJERCICIO 6

1. ¿De cuántos procesos consta el análisis de los documentos por defecto?
   Stopwords + PorterStemmer
2. ¿Se incluye eliminación de palabras vacías?
   Sí
3. ¿Se incluye stemming?
   Sí

Ya podemos efectuar consultas con esta colección. Para ello, basta navegar hasta el directorio $TERRIER_INSTALL en un terminal y teclear el comando de recuperación interactiva (para abrir una consola o terminal, puede consultarse el apéndice 'Apertura de una consola, terminal o línea de comandos' en este mismo documento; los comandos figuran en negritas):

cd $TERRIER_INSTALL [[En nuestro caso, C:\Users\juan\Desktop\terrier-core-4.2]]
bin\interactive_terrier.bat

El Sistema responderá con un mensaje que incluye al final:

Please enter your query:

- **Búsqueda de un término aislado**

  Basta introducir el término a continuación del mensaje 'Please enter your query:'

EJERCICIO 7

Realizar una búsqueda del término 'law' [sin comillas] en la colección. ¿Qué puntuación obtiene el primer documento relevante en la respuesta?

SOLUCIÓN EJERCICIO 7

Realizar una búsqueda del término 'law' [sin comillas] en la colección. ¿Qué puntuación obtiene el primer documento relevante en la respuesta?

El primer documento mostrado por el sistema es: WTX001-B34-90

La puntuación obtenida por el primer documento es: 5.46651

- **Búsqueda con AND**

  Se realiza anteponiendo a cada término el signo '+' [sin comillas] a continuación del mensaje 'Please enter your query:'

EJERCICIO 8

Realizar una búsqueda de modo que me devuelva los documentos en los que aparezcan simultáneamente los términos ‘tropical’ [sin comillas] y ‘storms’ [sin comillas]. ¿Qué puntuación obtiene el primer documento relevante en la respuesta?

SOLUCIÓN EJERCICIO 8

Realizar una búsqueda de modo que me devuelva los documentos en los que aparezcan simultáneamente los términos ‘tropical’ [sin comillas] y ‘storms’ [sin comillas]. ¿Qué puntuación obtiene el primer documento relevante en la respuesta?

El primer documento mostrado por el sistema es: WTX001-B38-205

La puntuación obtenida por el primer documento es: 13.991

- **Búsqueda con OR**

  Se realiza simplemente introduciendo los términos con un espacio entre ellos (operador por defecto) a continuación del mensaje ‘Please enter your query:’

EJERCICIO 9

Realizar una búsqueda de modo que me devuelva los documentos en los que aparezcan al menos una de las siguientes palabras: ‘steel’ [sin comillas], ‘gold’ [sin comillas]. ¿Qué puntuación obtiene el primer documento relevante en la respuesta?

SOLUCIÓN EJERCICIO 9

Realizar una búsqueda de modo que me devuelva los documentos en los que aparezcan al menos una de las siguientes palabras: ‘steel’ [sin comillas], ‘gold’ [sin comillas]. ¿Qué puntuación obtiene el primer documento relevante en la respuesta?

El primer documento mostrado por el sistema es: WTX001-B34-450

La puntuación obtenida por el primer documento es: 7.992356

- **Búsqueda con NOT**

  Se realiza anteponiendo al término el signo – (sin espacios en medio) a continuación del mensaje ‘Please enter your query:’

EJERCICIO 10

Realizar una búsqueda de modo que me devuelva los documentos en los que NO aparezca el término ‘rates’ [sin comillas], pero en los que aparezcan simultáneamente los términos ‘declining’ [sin comillas] y ‘birth’ [sin comillas]. ¿Qué puntuación obtiene el primer documento relevante en la respuesta?

SOLUCIÓN EJERCICIO 10

Documentos en los que NO aparezca el término ‘rates’ [sin comillas] y aparezcan simultáneamente los términos ‘declining’ y ‘birth’ [sin comillas]. ¿Qué puntuación obtiene el primer documento relevante en la respuesta?

El primer documento mostrado por el sistema es: WTX001-B17-95

La puntuación obtenida por el primer documento es: 10.44381

Bibliografía esencial

BIBLIOGRAFÍA ESENCIAL SOLR

- GRAINGER, T.; POTTER, T. (2014). Solr in action. Manning Publications Co.
- APACHE SOLR REFERENCE GUIDE [página web]. Disponible en: https://lucene.apache.org/solr/guide/7_1
- SOLR TUTORIAL [página web]. Disponible en: https://lucene.apache.org/solr/guide/solr-tutorial.html

BIBLIOGRAFÍA ESENCIAL INDRI

- THE LEMUR PROJECT [sitio web]. Disponible en: https://www.lemurproject.org
- INDRI [página web]. Disponible en: https://www.lemurproject.org/indri/
- INDRI DOCUMENTATION [página web]. Disponible en: https://lemur.sourceforge.io/indri/index.html

BIBLIOGRAFÍA ESENCIAL TERRIER

- TERRIER IR PLATFORM [sitio web]. Disponible en: http://www.terrier.org
- DOCUMENTATION FOR TERRIER [página web]. Disponible en: http://terrier.org/docs/v4.2/
- QUICKSTART GUIDE: USING TERRIER FOR EXPERIMENTS [página web]. Disponible en: http://terrier.org/docs/v4.2/quickstart_experiments.html

Apéndices

Apertura de una consola, terminal o línea de comandos

Para abrir una CONSOLA, TERMINAL o LÍNEA DE COMANDOS, basta ir a Inicio (el icono en el margen inferior izquierdo de la pantalla con la bandera de Windows) y en el buscador que aparece en la parte inferior de la pantalla con la frase 'Buscar programas y archivos' introducimos 'cmd' [sin las comillas simples; solamente las letras cmd]. Aparecerá una pantalla con fondo negro y una última línea semejante a:

C:\Users\Juan>_

Hasta el signo > el sistema nos está indicando en qué subdirectorio nos encontramos (en este caso, en el subdirectorio 'Juan', dentro del directorio 'Users', dentro de la unidad 'C' de nuestro ordenador). El prompt (_) está parpadeando en espera de que introduzcamos un comando (de donde la abreviatura 'cmd' introducida anteriormente).

De esta manera hemos abierto una CONSOLA, TERMINAL o LÍNEA DE COMANDOS que nos permite introducir directamente comandos al sistema para su ejecución.

Instalación de Java (JRE) en Windows

En primer lugar, descargamos una versión superior a la 1.8 del programa Java (JRE) desde la página web de Oracle

http://www.oracle.com/technetwork/java/javase/downloads/index.html

En el margen derecho de la primera tabla que figura en esta tabla se observarán las opciones: JDK, Server JRE y JRE (con un botón 'Download' debajo de cada uno de ellos). En nuestro caso, puede escogerse la opción 'Server JRE' si se dispone de un ordenador de 64 bits, o la opción 'JRE' tanto si se dispone de un ordenador de 32 bits o de 64 bits. En nuestro caso, escogemos la opción 'Server JRE' (haciendo clic en el botón Download correspondiente) y luego la versión adecuada a un ordenador con sistema operativo Windows, en nuestro caso 'server-jre-8u144-windows-x64.tar.gz' (tras aceptar las condiciones de uso).

Una vez descargado el archivo comprimido 'server-jre-8u144-windows-x64.tar.gz' en la carpeta 'Descargas', lo descomprimimos en la carpeta donde queremos instalarlo. En nuestro caso, dada la utilidad del programa Java (JRE) para muchas aplicaciones, lo guardaremos en el subdirectorio

C:\"Program Files <86>"\Java\

Para ello, seleccionamos el archivo 'server-jre-8u144-windows-x64.tar.gz' en 'Descargas', con el botón derecho seleccionamos la opción 'Cortar' y a continuación navegamos hasta el subdirectorio donde queramos instalar el programa, en nuestro caso:

C:\Archivos de programa (x86)\Java

Dentro de la carpeta Java (si no existe, creamos dicha carpeta), pegamos el archivo 'server-jre-8u144-windows-x64.tar.gz'. En ocasiones podemos encontrar un mensaje del tipo: "Necesitará proporcionar permisos de administrador para mover a esta carpeta". En tal caso, basta con hacer clic en el botón 'Continuar'. Allí descomprimiremos ese archivo con un programa compresor-descompresor (WinRAR en nuestro caso). Para ello, con el botón derecho desplegamos el menú contextual

correspondiente al archivo que deseamos descomprimir y elegimos la opción 'Extraer aquí'. Tras unos segundos se mostrará en pantalla la carpeta con el nombre 'jdk1.8.0_144' o similar ('jre1.8.0_144', por ejemplo). El programa Java (JRE) está ya instalado en la ruta:

C:\"Program Files <86>"\Java\jdk1.8.0_144

Es importante advertir que debemos configurar adecuadamente la variable de entorno JAVA_HOME para que el Sistema emplee el programa Java (JRE) cuando lo necesite (al ejecutar el programa Solr, por ejemplo). Por ello, la instalación debe completarse con la configuración de la variable de entorno JAVA_HOME (se puede consultar este aspecto en el apéndice 'Configuración de la variable JAVA_HOME' en este mismo documento).

La instalación del programa Java no se habrá completado hasta configurar adecuadamente la variable JAVA_HOME.

Configuración de la variable JAVA_HOME

Para que el sistema emplee el programa Java (JRE) cuando lo precise, es necesario indicar al sistema en qué carpeta se ha instalado dicho programa. Este proceso se denomina de configuración de la variable de entorno 'JAVA_HOME'. En este apartado abordamos el proceso de configuración de dicha variable de entorno en Windows, lo que requiere los siguientes pasos:

Paso 1.- Seguir la siguiente ruta:

Inicio > Panel de Control > Sistema y Seguridad > Sistema > Configuración avanzada del sistema > Variables de entorno

Paso 2.- Allí, dentro del apartado 'Variables de usuario para ….', debemos seleccionar la línea correspondiente a la variable de entorno 'JAVA_HOME'. Se mostrará la línea en azul.

Paso 3.- Apuntamos la ruta completa en la que figura el archivo con el programa Java (JRE) en la definición de la variable de entorno 'JAVA_HOME'. En nuestro caso, por ejemplo:

C:\"Program Files <86>"\Java\jdk1.8.0_144

Paso 4.- Comprobamos que efectivamente en dicha ruta se halla la carpeta del programa Java (en nuestro caso, 'jdk1.8.0_144').

Paso 5.- Si en dicha ruta no figura el directorio correspondiente a la versión de Java empleada actualmente por el sistema (la variación puede deberse simplemente a la versión; en nuestro caso, p. ej., podría no figurar 'jdk1.8.0_144', sino una versión anterior como 'jre1.8.0_91'), debemos actualizar el valor de la variable de entorno 'JAVA_HOME'. Para ello ir a:

Equipo > Disco local (C:)

y navegar por los directorios 'Archivos de programa' y 'Archivos de programa (x86)' hasta localizar una carpeta 'Java' que contenga a su vez otra carpeta del tipo 'jdk1.8.0_144'. Si existiese tal carpeta en ambos directorios, escoger la ruta de la versión

más reciente. Repetir los pasos 1, 2 y 3 para apuntar la ruta completa de la carpeta elegida con el programa Java (JRE) en la variable de entorno JAVA_HOME. El resultado de este proceso es que la variable de entorno JAVA_HOME se ha definido con la ruta completa donde se encuentra una versión 1.8 o superior del programa Java (JRE).

Paso 6.- Una vez seleccionada la versión más reciente y apuntada la ruta donde se encuentra dicha versión en la variable de entorno JAVA_HOME, navegaremos hasta la carpeta ('jdk1.8.0_144' en nuestro caso) y hacemos doble clic en ella para ver su contenido. Debemos observar una carpeta 'bin' y un archivo 'README' entre otras carpetas y archivos. Si no es así, debemos instalar de nuevo el programa Java, escogiendo una versión más reciente.

Paso 7.- Reiniciar el sistema. Para ello, ir a 'Inicio' (icono con la bandera de Windows en el margen inferior izquierdo de la pantalla), y seleccionar 'Reiniciar' junto al botón 'Apagar'). Con ello hemos terminado la correcta instalación del programa Java (JRE).

Instalación de WinRAR en Windows

Paso 1. Ir a la dirección 'https://www.winrar.es'

Paso 2. Hacemos clic en la sección 'Descargas' dentro del menú superior de la página. Nos indicará cuál es nuestro sistema operativo y el archivo recomendado para la descarga, compatible con dicho sistema operativo.

Paso 3. Hacemos clic en el archivo recomendado en el paso anterior. Comenzará la descarga en nuestro ordenador local.

Paso 4. Una vez descargado, hacemos doble clic en dicho archivo, pudiendo aceptarse todas las opciones por defecto que nos indique durante el proceso de instalación.

Paso 5. Una vez instalado, surgirá una ventana indicándonos si deseamos utilizar WinRAR en ese instante.

Paso 6. Puede consultarse el apartado 'Cómo se descomprime un archivo con WinRAR' en la página 'https://www.winrar.es/soporte'.

Error ‘set de JAVA_HOME environment var.’

En ocasiones, al ejecutar un programa que requiere Java (JRE), se recibe el siguiente mensaje de error:

“Please set the JAVA_HOME environment variable to the path where you installed Java 1.8+”

Ello es debido habitualmente a dos posibles razones: una primera, que en la ruta especificada en la variable de entorno JAVA_HOME no figura la carpeta con el programa Java (porque está en otra ruta o porque la versión de la variable de entorno no coincide con la que se halla en la ruta especificada). En primer lugar, debemos comprobar que este aspecto se cumple satisfactoriamente. Esta comprobación puede realizarse de nuevo repitiendo los pasos 4 y 5 del apéndice ‘Configuración de la variable de entorno JAVA_HOME’.

La segunda razón que puede motivar la presencia de este error consiste en que el sistema no reconoce la ruta del programa Java (JRE) incluida en la variable de entorno JAVA_HOME, que para nosotros es aparentemente correcta. Ello suele deberse a que en la ruta figura el subdirectorio ‘Archivos de programa (x86)’, que el sistema escribe internamente de otra manera.

Para evitar problemas con la ruta donde se encuentra el programa Java (JRE), o si creemos posible que esta sea la causa del error, seguimos los siguientes pasos:

Paso 1.- Mediante el navegador nos trasladamos hasta la carpeta con el programa Java (JRE), en nuestro caso ‘jdk1.8.0_144’:

C:\Archivos de programa (x86)\Java\jdk1.8.0_144

Estando dentro de la carpeta jdk1.8.0_144, vamos al área de texto superior de la venta (donde se encuentra la ruta en la que nos hallamos) y hacemos clic en la parte derecha del área de texto (donde no haya ningún carácter ni el icono con la flecha hacia abajo).

Paso 2.- Al hacer clic, veremos que se marca en azul una ruta completa del tipo:

C:\Program Files (x86)\Java\jdk1.8.0_144

Estando seleccionada esta ruta, copiarla en el portapapeles (Ctrl+C).

Paso 3.- Ir de nuevo a:

> Panel de control > Sistema y Seguridad > Sistema > Configuración avanzada del sistema > Variables de entorno

Paso 4.- Allí, dentro del apartado 'Variables de usuario para…..', seleccionamos la línea correspondiente a la variable de entorno JAVA_HOME. Se marcará la línea en azul.

Paso 5.- Hacemos clic en el botón 'Editar'.

Paso 6.- Eliminamos la ruta inicial que figure dando simplemente a la tecla 'Supr' del teclado, comprobando que el 'valor de la variable' está totalmente vacío (si se observa algún carácter como punto, coma, punto y coma, etc., lo eliminamos).

Paso 7.- Pegamos la ruta copiada previamente en el portapapeles mediante 'Ctrl+V'

Paso 8.- Guardamos la nueva ruta pulsando en el botón 'Aceptar'.

Paso 9.- En la ventana 'Variables de entorno', volvemos a pulsar en 'Aceptar'.

Paso 10.- En la ventana 'Propiedades del sistema' volvemos a pulsar en 'Aceptar'.

Paso 11.- Reiniciamos el sistema.

Error ‘destination directory cannot be created’

En ocasiones, principalmente cuando se ejecuta por vez primera el programa Solr mediante los comandos:

```
cd  $SOLR_INSTALL\bin

solr  start  -e  techproducts
```

Podemos obtener el siguiente mensaje de error:

“ERROR: Destination
‘C:\ruta_hasta_el_directorio_de_instalacion_de_Solr\example\techproducts\solr’
directory cannot be created”

Ello suele ser debido a que el usuario que utiliza el programa debe tener TODOS LOS PERMISOS (de escritura, creación, eliminación y ejecución), NO SOLO EL PERMISO DE LECTURA, sobre el directorio de instalación de Solr y sobre todos los subdirectorios y archivos bajo la carpeta donde se haya instalado el programa.

La propiedad y el control total sobre el subdirectorio de instalación de Solr ($SOLR_INSTALL) puede conseguirse de dos maneras esencialmente: a través de la interfaz gráfica y a través de comandos.

La propiedad y el control total de la carpeta $SOLR_INSTALL a través de la interfaz gráfica se consigue haciendo clic con el botón derecho en el directorio/carpeta $SOLR_INSTALL. Seleccionar ‘Propiedades’ y a continuación elegir el apartado ‘Seguridad’. Dentro de él, hacer clic en ‘Opciones avanzadas’ y luego en ‘Propietario’. Comprobar que el propietario que figura es el usuario que va a utilizar el programa; en nuestro caso, ‘JUAN-HP\juan’. Si no lo es, pulsar en ‘Editar’ para modificarlo. A su vez, dentro de ‘Opciones avanzadas’, junto a ‘Propietario’ figura el apartado ‘Permisos efectivos’. Al hacer clic en ‘Seleccionar’ el ‘Nombre de grupo o de usuario’, en el área de texto debe figurar el mismo usuario propietario; en nuestro caso, ‘JUAN-HP\juan’. Aparecerán en pantalla los permisos para dicho usuario. Si no están marcados, deberán marcarse todas las opciones que se muestren en pantalla. Finalmente, hacer clic sucesivamente en ‘Aceptar’ hasta cerrar todas las pantallas abiertas. Debe advertirse

que este proceso debe repetirse con cada uno de los subdirectorios y archivos dentro de la carpeta $SOLR_INSTALL.

La propiedad y el control total de la carpeta $SOLR_INSTALL a través de comandos puede conseguirse ejecutando en un TERMINAL, CONSOLA o LÍNEA DE COMANDOS (ver el apéndice 'Apertura de una consola, terminal o línea de comandos' en este mismo documento) los siguientes comandos (que figuran en negritas):

takeown /f $SOLR_INSTALL* /r

[[Este comando realiza el cambio de propietario]]

cd C:\Windows\System32\es-ES

[[Nos vamos al directorio donde se encuentre el comando "icacls"]]

icacls $SOLR_INSTALL /grant JUAN-HP\juan:F /t

[[Este comando otorga control total a la carpeta $SOLR_INSTALL de manera recursiva]]

Error ‘the system cannot find the path’

En ocasiones, al ejecutar el comando:

bin\trec_setup.bat C:\Users\juan\Desktop\colwt2g

Se recibe el siguiente mensaje de error:

“The system cannot find the path specified”

O su correspondiente versión en español:

“El sistema no puede encontrar la ruta especificada”

Ello suele deberse a que el sistema no localiza la colección ‘colwt2g’ en la ruta introducida en el comando: ‘C:\Users\juan\Desktop\colwt2g’.

En consecuencia, debemos primeramente comprobar que la ruta donde se ha descomprimido la colección ‘colwt2g’ figura correctamente en el comando introducido. En concreto:

- No debemos olvidar que la ruta donde está localizada la **colección descomprimida** ‘colwt2g’ debe comenzar por ‘C:\’
- Debemos comprobar que existen los sucesivos directorios de la ruta hasta llegar a la colección (respetar mayúsculas, por ejemplo).

Para comprobar este último aspecto, puede navegarse paso a paso de manera que confirmemos que los sucesivos directorios realmente existen y cómo se denominan. Siguiendo con el ejemplo, introduciríamos sucesivamente los siguientes comandos en un terminal o consola:

```
cd C:\Users
cd juan
cd Desktop
cd colwt2g
```

También puede suceder que el error se deba al primer comando ejecutado justo antes de este, esto es:

cd $TERRIER_INSTALL

En ese caso, ello se debe habitualmente a que el sistema no tiene constancia de que el directorio $TERRIER_INSTALL es un directorio de trabajo donde se deben ejecutar programas.

Para comprobar este aspecto, cambiamos primeramente de directorio ejecutando el siguiente comando en un terminal (para abrir un terminal, puede consultarse el apéndice 'Apertura de una consola, terminal o línea de comandos' en este mismo documento):

cd $TERRIER_INSTALL\bin

Estando en este nuevo directorio, ejecutar el siguiente comando en el terminal:

trec_setup.bat C:\Users\juan\Desktop\colwt2g

Si el comando no se ejecuta tampoco, y nos devuelve algún tipo de error, debemos entonces revisar las variables de entorno del sistema. Debe tenerse en cuenta, como se dice al comienzo del taller (Véase el apartado 'Instalación, apertura y cierre del programa Terrier en Windows'), que la variable de entorno denominada 'TERRIER_HOME' debe figurar entre las variables del usuario (no entre las variables del sistema), y que su valor debe ser la ruta hasta el directorio de instalación de Terrier.

En nuestro caso, por ejemplo, TERRIER_HOME debe poseer el valor:

C:\Users\juan\Desktop\terrier-core-4.2

Por último, no debemos olvidar nunca dos aspectos a la hora de definir una variable de entorno:

- Una vez introducido el valor de la variable, debemos hacer clic en 'Aceptar' varias veces, en las sucesivas pantallas que vayan surgiendo.
- Conviene reiniciar el sistema para asegurarnos de que se adoptan los últimos valores recientemente definidos.