Práctica 1 BMI

Memoria

20/02/2020

Adrián Rubio Pintado
Jorge Muñoz Aguado

Qué versión(es) del modelo vectorial se ha(n) implementado en el ejercicio 2.

Se ha implementado el modelo *tf-idf* para el cálculo de las puntuaciones, junto con el coseno.

Cómo se ha conseguido colocar un documento en la primera posición de ránking, para cada buscador implementado en el ejercicio 2.

Se ha hecho uso de una lista con contenido como `Entry<Integer, Double>`, lo cual permite guardar el identificador del documento(docID) y la puntuación del ranking. A la hora de querer devolver el Top de los elementos con mayor puntuación ordenamos la lista en función de la puntuación.

Como la ordenación se hace de menor a mayor, damos la vuelta a la lista, aunque también se podría recorrer de manera inversa.

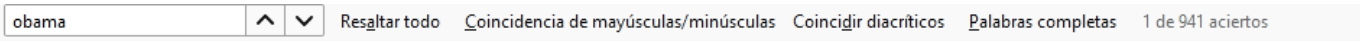
Añadir a mano un documento a la colección docs1k.zip de manera que aparezca el primero para la consulta "obama family tree" para este buscador. Documentar en la memoria cómo se ha conseguido y por qué resulta así.

El documento que aparece en primer resultado según este buscador es:

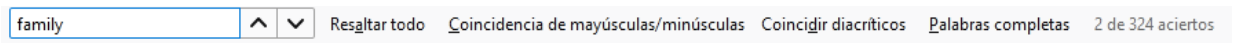
clueweb09-enwp02-06-15081.html

Dentro de su código fuente contamos cuantas veces aparecen los 3 términos de la consulta:

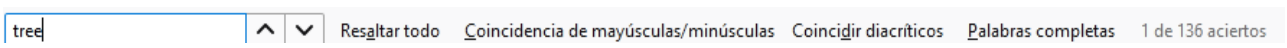
Obama:941 veces



Family: 324 veces



Tree: 136



Para posicionar nuestro documento (“DOC_NUEVO_OBAMA.html”) en el primer resultado, lo que hemos hecho es replicar cada termino tantas veces como aparecía en el documento anterior, que se posicionaba en el Top 1 del ranking, y le hemos añadido una aparición más de uno de los terminos, en este caso el término elegido fue “obama”. Es decir, que hemos incrementado en 1 la frecuencia de aparición de uno de los términos. A continuación, el resultado:

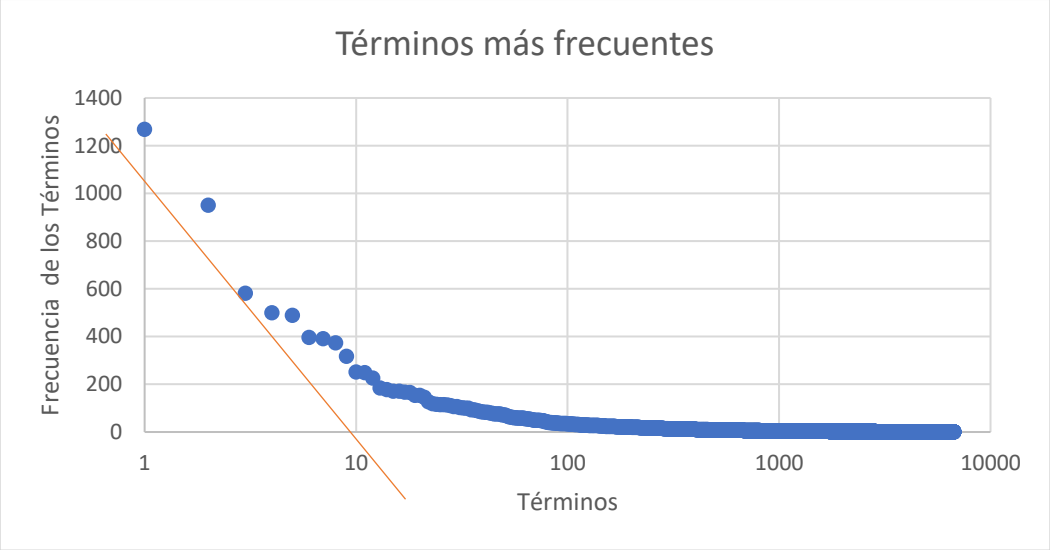
```
Checking search results
  LuceneEngine: top 5 for query 'obama family tree'
        4.708639621734619       file:collections/docs1k/DOC_NUEVO_OBAMA.html
        4.606169700622559       file:collections/docs1k/clueweb09-en0010-79-2218.html
        4.5331010818481445      file:collections/docs1k/clueweb09-en0001-02-21241.html
        4.4675140380859375      file:collections/docs1k/clueweb09-en0010-57-32937.html
        4.358207702636719       file:collections/docs1k/clueweb09-enwp02-06-15081.html

  VSMDotProductEngine: top 5 for query 'obama family tree'
        44.72024359282921       file:collections/docs1k/DOC_NUEVO_OBAMA.html
        32.58913886974855       file:collections/docs1k/clueweb09-enwp02-06-15081.html
        32.58840628987505       file:collections/docs1k/clueweb09-enwp03-07-2998.html
        32.58840628987505       file:collections/docs1k/clueweb09-enwp03-00-6901.html
        32.58840628987505       file:collections/docs1k/clueweb09-enwp01-06-3485.html

  VSMCosineEngine: top 5 for query 'obama family tree'
        0.0026949432036876765   file:collections/docs1k/DOC_NUEVO_OBAMA.html
        0.001963895347053642    file:collections/docs1k/clueweb09-enwp02-06-15081.html
        0.0019638512001306243   file:collections/docs1k/clueweb09-enwp03-07-2998.html
        0.0019638512001306243   file:collections/docs1k/clueweb09-enwp03-00-6901.html
        0.0019638512001306243   file:collections/docs1k/clueweb09-enwp01-06-3485.html
```

Ejercicio 3. Descripción

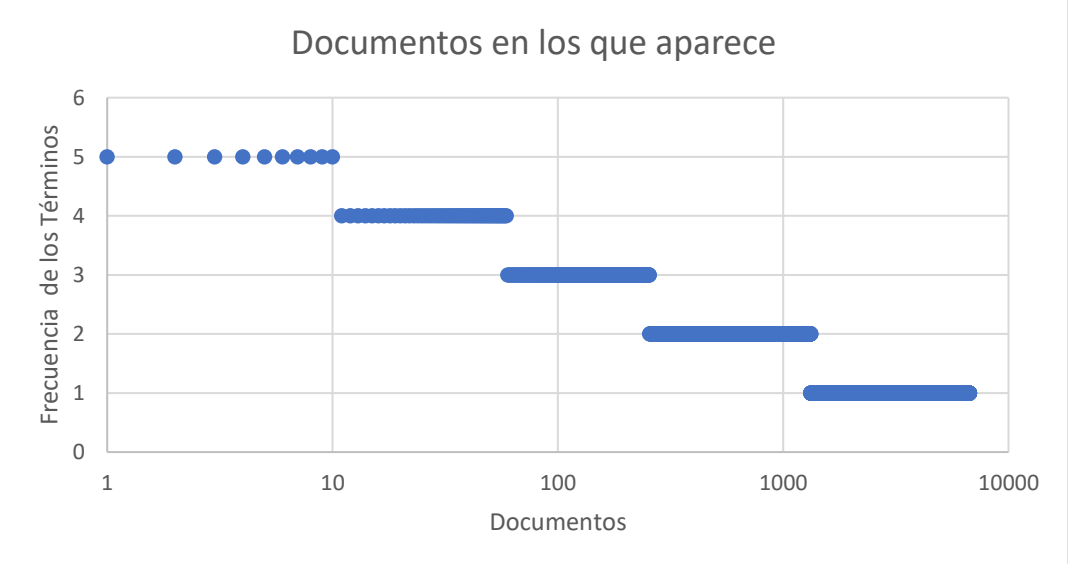
3.1 Figuras de las frecuencias.

Para la realización de esta parte hemos creado la clase `TermStats` en el paquete `es.uam.eps.bmi.search.test`. En esta clase *main* hacemos uso de partes de código ya implementadas por los profesores en la clase `TestEngine`, pero modificadas. Los resultados se han obtenido con el documento `urls.txt`.

A vista de las gráficas obtenidas en base logarítmica en base 10 observamos una clara tendencia casi lineal de los primeros términos usados. Si observamos toda la gráfica observamos una tendencia curva.



En el caso de la gráfica correspondiente número de documentos en los que aparecen los términos también observamos este efecto. La tendencia es más clara en esta gráfica. Algunos términos aparecen en todos los documentos (5) y cuanto en menos documentos aparecen los términos más palabras hay.



3.2 Descripción de las diferentes opciones probadas, y los cambios observados en los resultados.

Se ha implementado el uso de stopwords en inglés, dentro del analyzer. Configurando la siguiente variable dentro de LuceneBuilder, a false, habilitaremos el analyzer con uso de stopwords.

```
boolean standarAnalyzer = false;
```

De este modo eliminamos las palabras más comunes en inglés, que no aportan significado en la búsqueda, como las conjunciones o los artículos, por ejemplo: "the".

Como por características propias del lenguaje estos términos son los más comunes, si los eliminamos del ranking, obtendremos un buscador con mayor eficiencia, dado lo que esperamos de él.

A continuacion podemos ver un ejemplo de los "Most frequent terms" de las urls dadas en el ejercicio con y sin stopwords habilitadas.

Sin stopwords:

```
Testing indices and search on collections/urls.txt

  Most frequent terms:
        the     1268
        of      950
        a       580
        and     501
        entropy 489
```

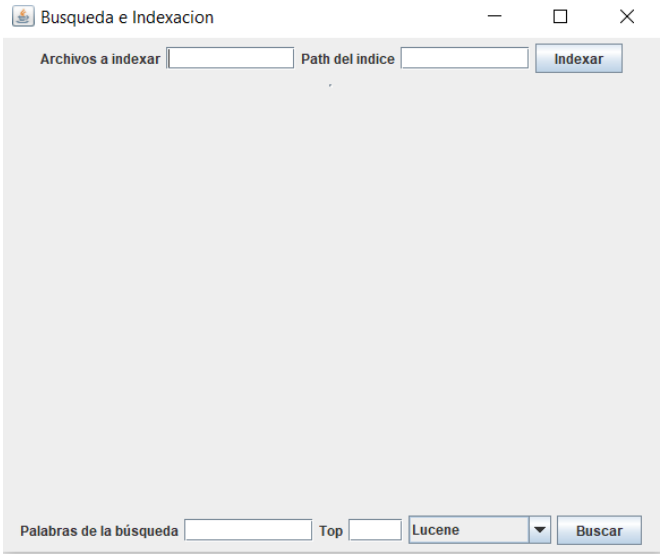
Con stopwords:

```
Testing indices and search on collections/urls.txt

  Most frequent terms:
        entropy 489
        information     249
        0       211
        x       183
        y       180
```

Observamos que hemos obtenido en el top, otra palabra que aporta información semántica: "**information**". Dado que no hemos incluido en la lista de stopwords letras,ni números o stopwords de otros idiomas, se nos "han colado" los terminos "0","x" e "y".

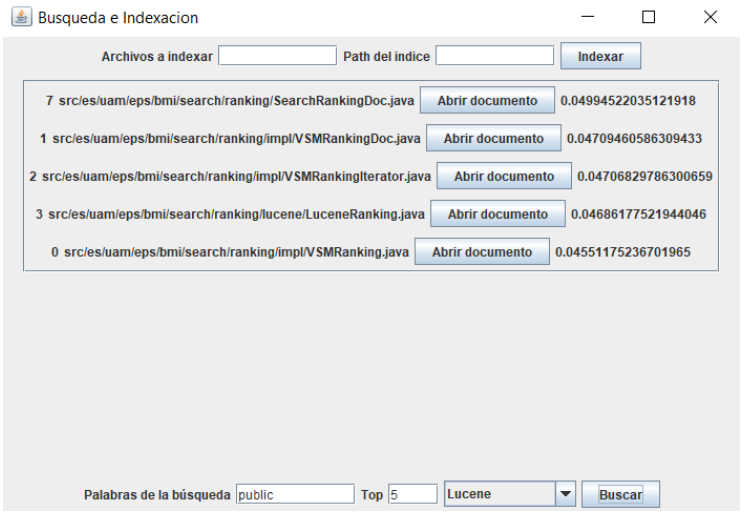
3.3 Explicación y captura de pantalla de la interfaz de usuario, e indicaciones para ejecutarla.

La interfaz de usuario se encuentra en el archivo `UserInterface.java` en el paquete `es.uam.eps.bmi.search.ui`. Al ejecutar nos aparecerá la siguiente ventana.



En la parte superior de la misma nos encontraremos el panel de indexación. En él podemos especificar la path de los archivos que deseamos indexar, `Archivos a indexar`, y la dirección donde se guardará el índice `Path del índice`. Al pulsar el botón `Indexar` se aplicarán nuestros cambios. Por defecto se indexa la dirección `src/es/uam/eps/bmi/search/ranking` y se guarda en `index/src`.

En la parte inferior nos encontramos el panel de búsqueda. Para buscar elementos debemos especificar las palabras de la búsqueda en la casilla de `Palabras de la búsqueda` y el número de elementos que deseamos que se muestren `Top`. También podremos seleccionar el motor de búsqueda que deseamos utilizar, entre los que se encuentran las siguientes opciones `Lucene`, `Producto escalar` y `Coseno`. Una vez especificada nuestra búsqueda podemos pulsar el botón `Buscar` y obtendremos el siguiente resultado.



Cada fila de la tabla se corresponde a un documento, donde la primera columna se refiere al identificar del documento en el índice, el segundo la dirección relativa del documento y la última a la puntuación obtenida por el fichero en la búsqueda. También se permite abrir documentos. En este caso al pulsar el botón se abrirá el explorador de ficheros y mostrará el archivo, el cual nosotros podremos abrir a continuación desde el programa deseado.