Optativa: Análisis y Recuperación de Información – Entrega Final

**Alumnos:**

**Mauricio Morales**

**Néstor Daniel Altamirano**

Profesora:

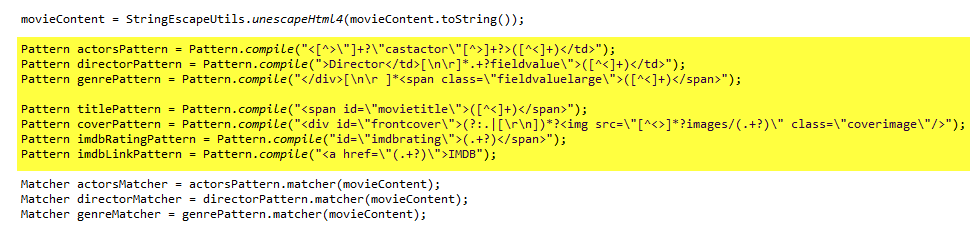
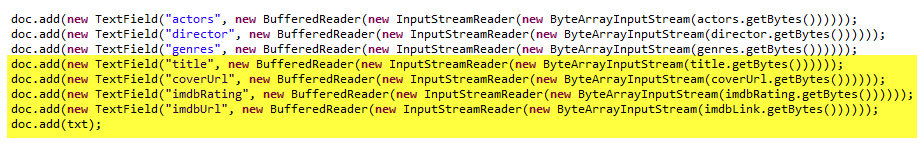
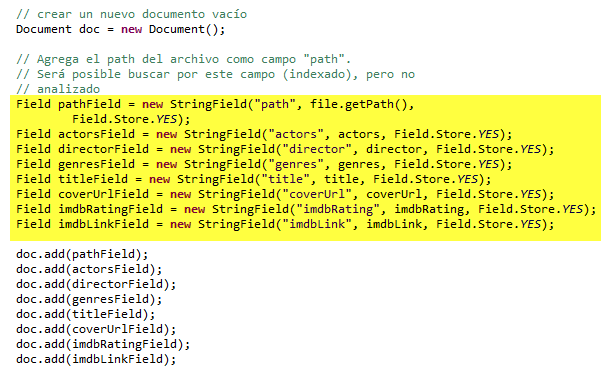
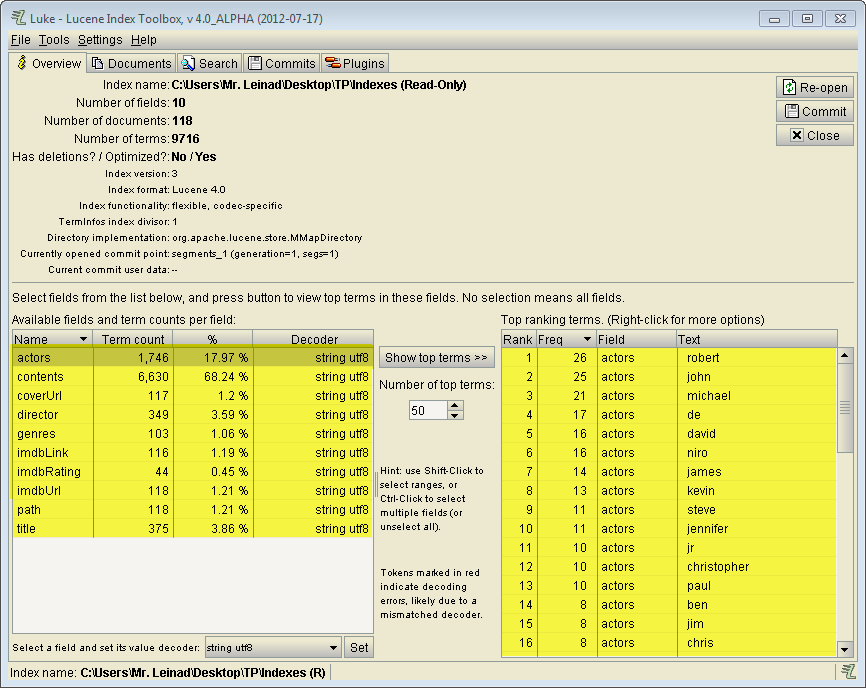
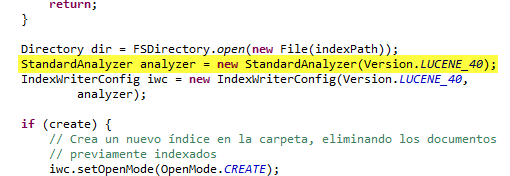
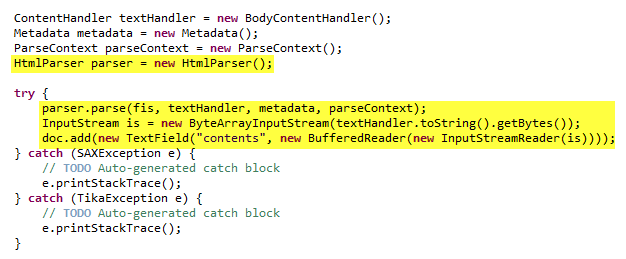
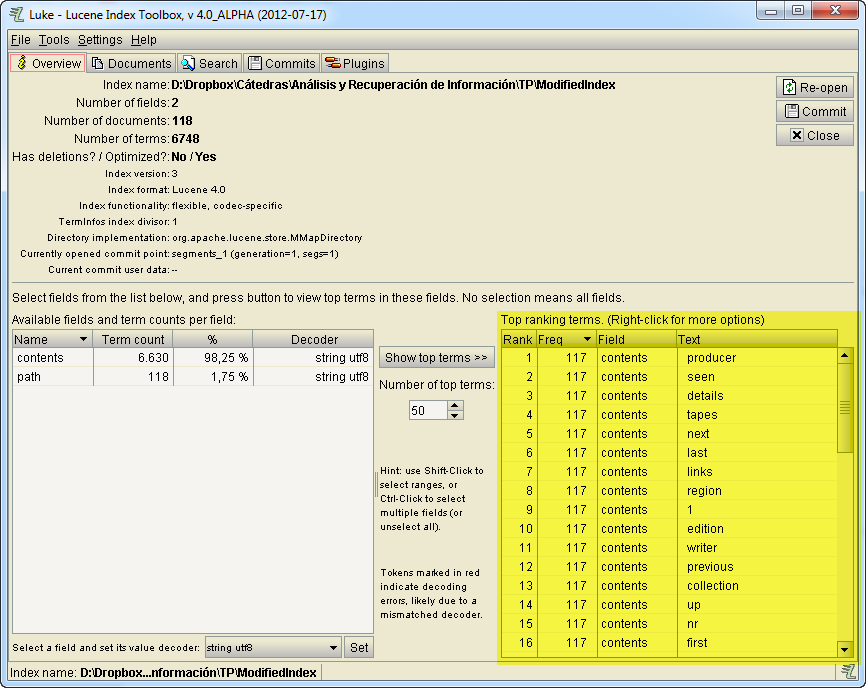
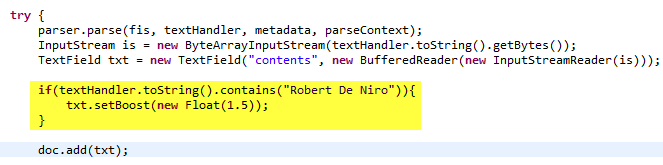
**Daniela Godoy**

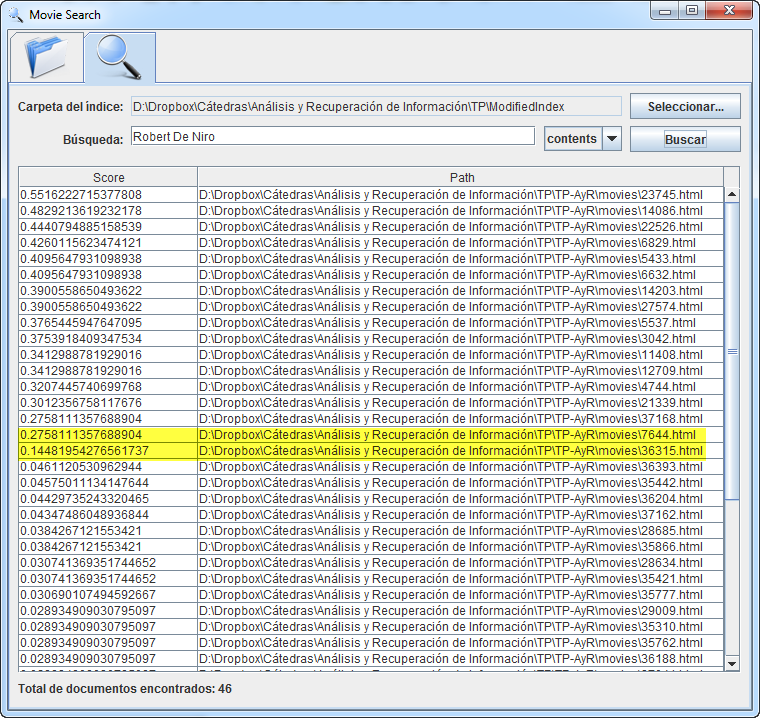
Cursada 2013

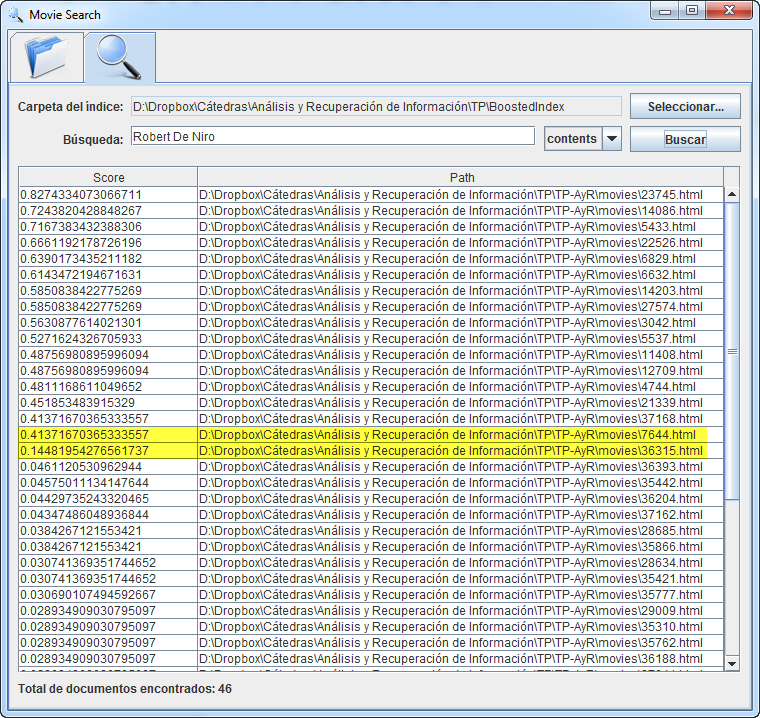
# Incisos

1. Parsear los documentos HTML proporcionados por la cátedra para que los documentos Lucene contengan los siguientes campos:
   * Género
   * Título
   * Imagen de tapa
   * Reparto (actores)
   * Director
   * Argumento
   * IMDB rating
   * IMDB link
2. Implementar los cambios sugeridos en el trabajo práctico de cursada relativos a los puntos 4, 5 y 6.
3. Mejorar la presentación de los resultados de las búsquedas mostrando el título y la imagen de tapa de las películas.

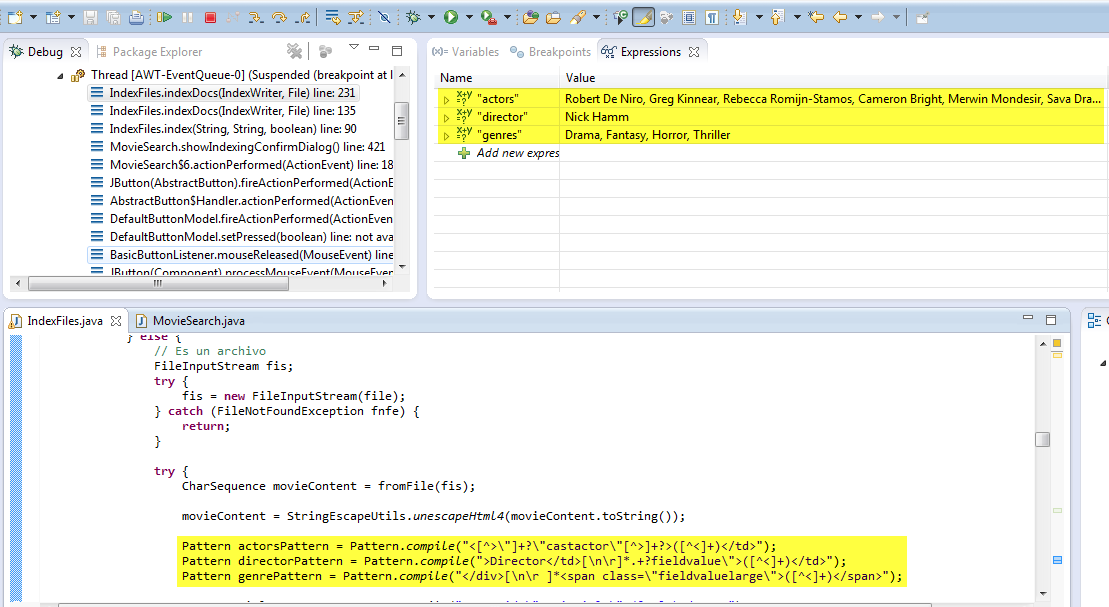
# Resolución

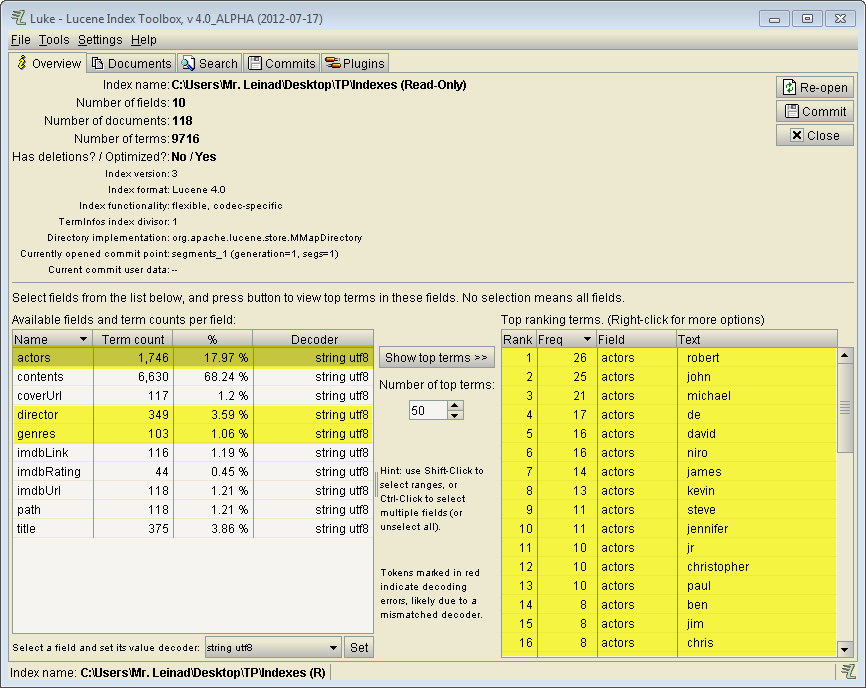
1. Utilizando expresiones regulares, extraemos los campos Género, Título, Imagen de tapa, Reparto (actores), Director, Argumento, IMDB rating, e IMDB link.  
     
     
     
     
     
   También almacenamos los campos para que puedan ser recuperables y no solamente indexables. Esto es útil para el campo “title”, el cual se utilizará en el último inciso para mostrar el título de la película al realizar una búsqueda.  
     
     
     
   En Luke, podemos analizar el índice y ver los campos almacenados y tokenizados.  
     
   
2. A fin de excluir StopWords del índice, utilizamos un StandardAnalyzer en lugar de un SimpleAnalyzer, el cual incluye un conjunto de StopWords a filtrar por defecto.  
     
   Luego, para excluir los tags HTML, utilizamos la librería Tika. Creamos un parser de tipo HtmlParser, y utilizamos el texto extraído para agregarlo al documento, sin incluir los tags HTML.  
     
     
     
   Y el resultado son las palabras importantes dentro del texto, sin incluir tags ni StopWords.  
     
     
     
     
   A fin de mejorar el posicionamiento de las películas en las que participe Robert De Niro, podemos aumentar el *Boosting* de aquellos documentos que contengan su nombre en el texto.  
     
     
     
   De esta manera, podemos observar la diferencia entre el score de una película en la que participa Robert De Niro (archivo **7644.html**) versus el score de una película en la que no participa (archivo **36315.html**), antes y después de realizar el cambio en el código.  
     
   **Índice antes de aumentar el *Boosting***

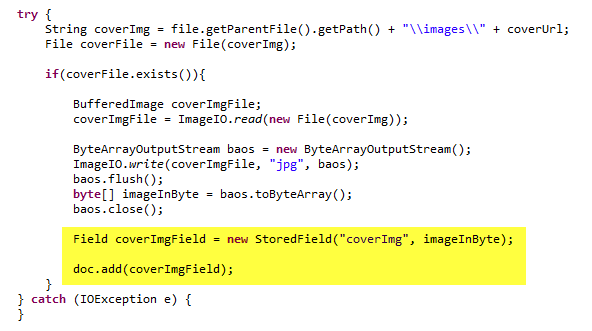
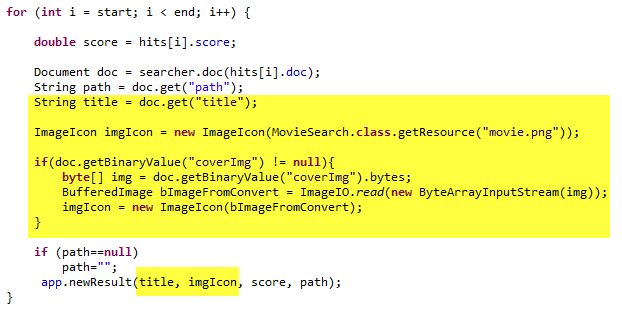
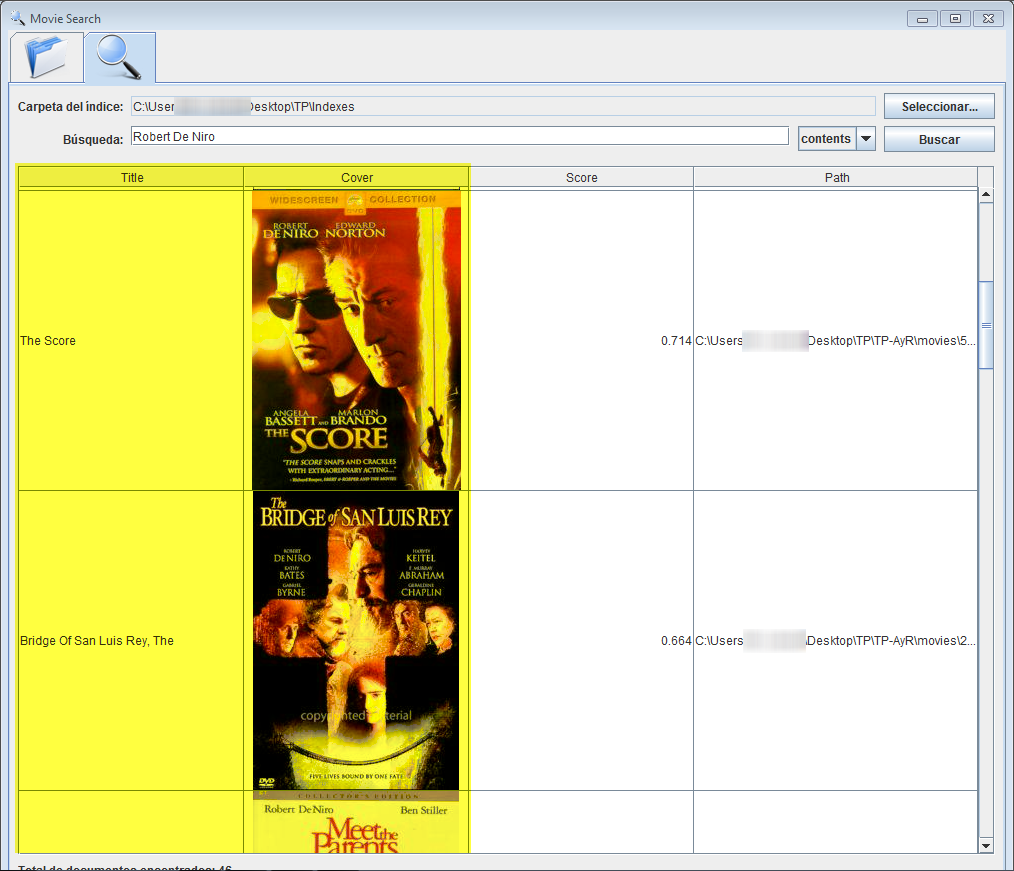


**Índice luego de aumentar el *Boosting***  


Para realizar una búsqueda avanzada por género, actores y director, lo que haríamos es parsear previamente la información al procesar cada página, extrayendo esos tres datos por separado, y almacenarlos en campos separados del contenido principal.  
  
Para lograr esto, podemos utilizar expresiones regulares que levanten la información del texto de cada página, y luego de separarla, crear campos *Genero*, *Actores*, y *Director*.





1. Almacenamos el título y la imagen provista para cada película por la cátedra en el índice Lucene, y luego la recuperamos y mostramos en los resultados para las búsquedas.  
     
     
     
     
     
     
     
   Para aquellas películas que no tienen una imagen disponible, utilizamos el ícono “movie.png” por defecto.  
     
   