

PRÁCTICA 4: DESARROLLO DE UN SISTEMA DE RECOMENDACIÓN BASADO EN FILTRADO COLABORATIVO

Manuel Blanco Rienda

GESTIÓN DE INFORMACIÓN EN LA WEB (2016-2017)

Memoria de la práctica 4 de GIW por Manuel Blanco Rienda

¿Qué se ha hecho?

La práctica cuatro de la asignatura Gestión de Información Web consta de dos partes diferenciadas, pero que dependen la una de la otra: la búsqueda de un vecindario de usuarios en base a las calificaciones que se especifican para una serie de películas, y la posterior recomendación en base a las películas que le podrían gustar, de entre las que han sido calificadas como buenas por los miembros de su vecindario. El sistema de recomendación que se ha desarrollado en definitiva, es por filtrado colaborativo, centrado en usuarios.

¿Cómo se ha hecho?

El proyecto se ha desarrollado utilizando los conceptos teóricos que se han proporcionado en la asignatura, aplicados en un marco práctico utilizando Netbeans como IDE de desarrollo. Tal y como se ha dicho antes, la práctica es de carácter evolutivo, en el sentido de que para llevar a cabo la funcionalidad final de la misma (recomendar películas al usuario), es necesario primero encontrar un vecindario que haya proporcionado unas calificaciones acordes a las que da el usuario al que queremos recomendar. Por ello, paso a describir cómo es la funcionalidad de la práctica, capa por capa.

El contenido en código de la práctica se ha estructurado en diferentes partes: dos vistas gráficas que hacen las veces de interfaz (Main y VistaResultados), dos clases de representación de tipo (Valoracion y Pelicula) y una clase que contiene la lógica principal del proceso de recomendación (Sugeridor).

Interfaz gráfica

La componen dos ventanas diferentes: “Main” y “VistaResultados”:

- Main: Es la ventana principal del sistema, a la que se accede al principio y en la que se muestran las veinte películas aleatorias que el usuario debe puntuar para poder obtener las recomendaciones que se le hacen. La puntuación se hace de forma muy gráfica, eligiendo de 1 estrella (no le ha gustado nada) a 5 estrellas (le ha gustado mucho) como baremo para calificar cada una de los elementos seleccionables de la lista, que representan una película.

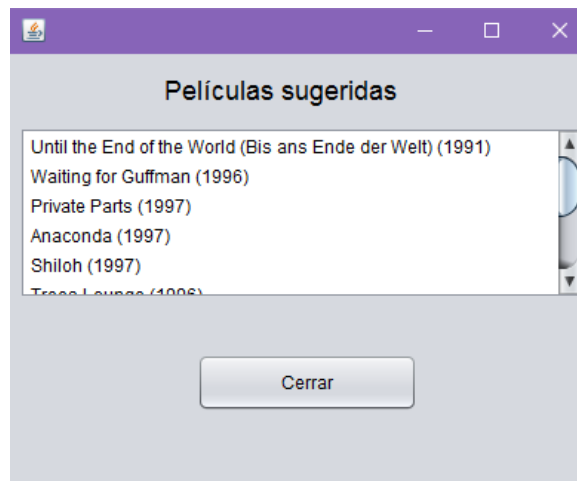
Una vez que el usuario las ha puntuado todas, pulsando sobre el botón “Ver películas sugeridas”, se procede a desencadenar el cálculo del vecindario del usuario y las recomendaciones que se le hacen en base a él. Una vez son obtenidas éstas, se le pasa el control a la ventana de resultados (“VistaResultados”).

Hay que tener en cuenta que pueden cambiarse las notas dadas a los ítems de la lista de películas cuando se quiera, obteniendo nuevas recomendaciones de forma dinámica.



A screenshot of a software window titled "Lista de Películas a valorar". The window has a purple title bar with standard Windows controls. The main area is light gray. At the top, the title "Lista de Películas a valorar" is centered. Below it is a list box containing five items: "797-Timecop (1994)", "124-Lone Star (1996)", "641-Paths of Glory (1957)", "71-Lion King, The (1994)", and "1509-Getting Even with Dad (1994)". To the right of the list box is a vertical scrollbar. Below the list box, there is a label "Nota (Sobre 5)" followed by a dropdown menu currently showing "No valor...". At the bottom of the window is a button labeled "Ver películas sugeridas".

- VistaResultados: Se encarga de mostrar únicamente la lista de películas sugeridas al usuario, en base a las notas que ha asignado a las películas que se le han presentado. Pueden ser seleccionadas para centrar el foco en ellas. Se provee de un botón para cerrar la vista de resultados y así poder repetir la entrada de datos al sistema y obtener recomendaciones.



A screenshot of a software window titled "Películas sugeridas". The window has a purple title bar with standard Windows controls. The main area is light gray. At the top, the title "Películas sugeridas" is centered. Below it is a list box containing five items: "Until the End of the World (Bis ans Ende der Welt) (1991)", "Waiting for Guffman (1996)", "Private Parts (1997)", "Anaconda (1997)", and "Shiloh (1997)". To the right of the list box is a vertical scrollbar. Below the list box is a button labeled "Cerrar".

Sugeridor

El sugeridor es la clase que contiene toda la lógica del sistema, en cuanto a búsqueda de un vecindario y obtención de unas películas que sean coherentes con respecto a las calificaciones dadas por el usuario, en base a ese vecindario. Para cada una de estas tareas se ha desarrollado un método que recoge el proceder para cada una:

- **calculaVecinos:** Es el método encargado de encontrar el vecindario más cercano posible para el usuario. Para ello, se utiliza una función de similitud del usuario principal con cada uno de los usuarios que han puntuado alguna de las películas que ha visto. Aquellos usuarios que tienen un valor de similitud con el principal más elevado, entran a formar parte del vecindario de este usuario principal, hasta un máximo de diez integrantes dentro de esa comunidad.

La función de similitud utilizada es la que corresponde con el algoritmo de Pearson y tiene la siguiente sintaxis:

$$sim(u, v) = \frac{\sum_{i \in S_{uv}} (r(u, i) - \bar{r}(u))(r(v, i) - \bar{r}(v))}{\sqrt{\sum_{i \in S_u} (r(u, i) - \bar{r}(u))^2} \sqrt{\sum_{i \in S_v} (r(v, i) - \bar{r}(v))^2}}$$

Donde:

- $r(u, i)$ es la calificación asignada por el usuario "u" (el principal) a la película "i".
 - $r_{media}(u)$ es la calificación media que suele asignar el usuario u a las películas que ve.
 - $r(v, i)$ es la calificación asignada por el usuario "v" (uno de los que están en la lista de calificaciones de películas) a la película "i".
- **calculaPelículas:** Una vez se tiene calculado el vecindario del usuario principal, se procede a obtener la lista de películas que se predicen que podrían obtener una muy buena calificación por parte del usuario. Para ello, al igual que en el anterior método, se sigue el filtrado colaborativo centrado en usuarios, concretamente con la función que tiene en cuenta la compensación de diferencias de interpretación y escala:

$$\hat{r}(u, i) = \bar{r}(u) + C \sum_{v \in N_u^*} sim(u, v)(r(v, i) - \bar{r}(v))$$

Donde:

- $r^{\wedge}(u,i)$ es la nota predicha que daría el usuario “u” al ítem “i”.
- $sim(u,v)$ es la similitud existente entre los usuarios “u” y “v”.
- “C” es una constante calculada en base a las similitudes del usuario principal con cada uno de sus vecinos. Tiene la siguiente sintaxis:

$$C = \frac{1}{\sum_{v \in N_u^+} |sim(u, v)|}$$

Una vez se tiene el conjunto de notas predichas, se procede a recomendar al usuario aquellas películas cuya nota predicha sea igual o superior a cuatro estrellas. Es decir, se le recomiendan aquellas películas que se espera que le vayan a gustar mucho.

Además de todo esto, la clase sugeridor posee varios métodos que en conjunto logran la obtención de la información de los ficheros de texto plano que contienen los datos de los usuarios, sus calificaciones y las películas, y su representación en objetos, ayudándose de las clases “Valoracion” y “Película”.

Clases de representación

Las clases de representación de información son aquellas que han servido para almacenar los datos de las instancias con las que ha sido necesario trabajar. Hasta el momento podemos diferenciar dos tipos:

- Valoracion: Es el tipo de dato que aglomera la información (almacenada en el fichero de texto plano “u.data”) de una valoración concreta dada por un usuario sobre una película, representada con un valor numérico del uno al cinco.
- Pelicula: Cada instancia de esta clase aglomera los datos de cada película existente en el fichero de texto plano “u.item”: id y título (aunque hay más atributos, solo estos dos son necesarios para mi sistema de recomendación).

Manual de usuario

La puesta en marcha y funcionamiento del sistema de recomendación es muy simple. Hay que pasarle obligatoriamente dos parámetros: la ruta del fichero de texto plano que tiene las valoraciones y la ruta del fichero de texto plano que contiene la información de las películas (en ese orden).

Si lo hacemos vía línea de comandos con el contenedor .jar que se incluye en esta entrega, debemos utilizar un comando con la siguiente sintaxis: "java -Xmx1024m -jar ruta_de_localizacion\recomendador.jar ruta_fichero_valoraciones ruta_fichero_peliculas". El fichero de valoraciones se llama "u.data" y el de las películas "u.item", y ambos están incluidos en la carpeta "data" dentro de los ficheros fuente del proyecto de la práctica.

Si no se especifican correctamente los dos argumentos descritos, el programa no funcionará o lo hará mal. En caso de no especificárselos, el programa avisará de ello al usuario. En caso de que se haya hecho correctamente, accedemos a la pantalla principal del sistema de la que se incluye captura de pantalla más arriba. Una vez ahí, se debe seleccionar una puntuación para cada uno de los veinte elementos que componen la lista de películas inicial, de forma que ninguna de ellas tenga la etiqueta "No valorada" en la nota. Hasta que esto anterior no se cumpla, el sistema no deja proceder con las recomendaciones.

Una vez esto se ha cumplido, se da paso a la ventana de resultados (de la que también se proporciona captura de pantalla más arriba), en la que podemos hacer "scroll" sobre una lista de películas con la misma forma que la que hay en la pantalla principal, pero sabiendo que éstas son recomendadas en base a las notas dadas a las primeras.

Elementos innovadores

Como elemento innovador que se ha desarrollado en la práctica, puede destacarse la interfaz gráfica del sistema, que no era requerida pero que sin embargo facilita sobremanera el proceso de entrada y obtención de datos de las películas.

Nota: No se ha utilizado más bibliografía que los apuntes proporcionados por el profesor de la asignatura.