



ugr

Universidad
de Granada

GESTIÓN DE INFORMACIÓN EN LA WEB
MÁSTER PROFESIONAL EN INGENIERÍA INFORMÁTICA

Desarrollo de un Sistema de Recomendación basado en Filtrado Colaborativo

Autor

Ernesto Serrano Collado

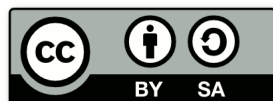
Profesor

Juan Manuel Fernández Luna



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍAS INFORMÁTICA Y DE
TELECOMUNICACIÓN

Granada, 17 de marzo de 2018



Desarrollo de un Sistema de Recomendación basado en Filtrado Colaborativo

Ernesto Serrano Collado

Resumen

Palabras clave: *software libre, recuperación información, filtrado colaborativo, recomendación*

Los objetivos de esta práctica son:

1. Entender el proceso de recomendación basado en filtrado colaborativo.
2. Ser capaz de plasmarlo en un programa de ordenador.

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

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

Fdo: Ernesto Serrano Collado

Granada, a 17 de marzo de 2018

Índice general

1. Introducción	1
2. Implementación	3
3. Manual	7
4. Bibliografía	9

Capítulo 1

Introducción

En esta práctica se construirá, partiendo de cero y en el lenguaje preferido, un sistema de recomendación de películas basado en filtrado colaborativo (usuario - usuario). Para tal fin, se puede utilizar una herramienta que muestre al usuario 20 películas al azar y las evalúe asignándole un valor de 1 estrella (*), indicando que no le gusta nada, hasta 5 estrellas (es una de sus películas favoritas).

El sistema de acción para el vecindario, es decir, el grupo de usuarios que más se parecen a él en cuanto a las películas y las evaluaciones dadas. Finalmente, la aplicación predecirá la valoración para el usuario activo de

todas las películas que haya visto a los vecinos y que no haya visto el usuario activo, y las estrellas predichas con cuatro o cinco estrellas.

Capítulo 2

Implementación

Se ha desarrollado un recomendador básico utilizando el lenguaje de programación `Python` y la base de datos `Movie Lens`.

En cuanto al desarrollo del algoritmos k-NN se han desarrollado siguiendo los ejemplos de la correlación de Pearson del temario de la asignatura puesto que indica que es la mas óptima.

$$sim(u, v) = \frac{\sum(r(u, i) - \bar{r}(u))(r(v, i) - \bar{r}(v))}{\sqrt{\sum(r(u, i) - \bar{r}(u))^2} \sqrt{\sum(r(v, i) - \bar{r}(v))^2}}$$

Donde $r(u, i)$ es la valoración del usuario u a la película i y $\bar{r}(u)$ la valoración media del usuario u .

Para *predecir* la valoración que le daríamos al resto de las películas hemos usado el siguiente algoritmo:

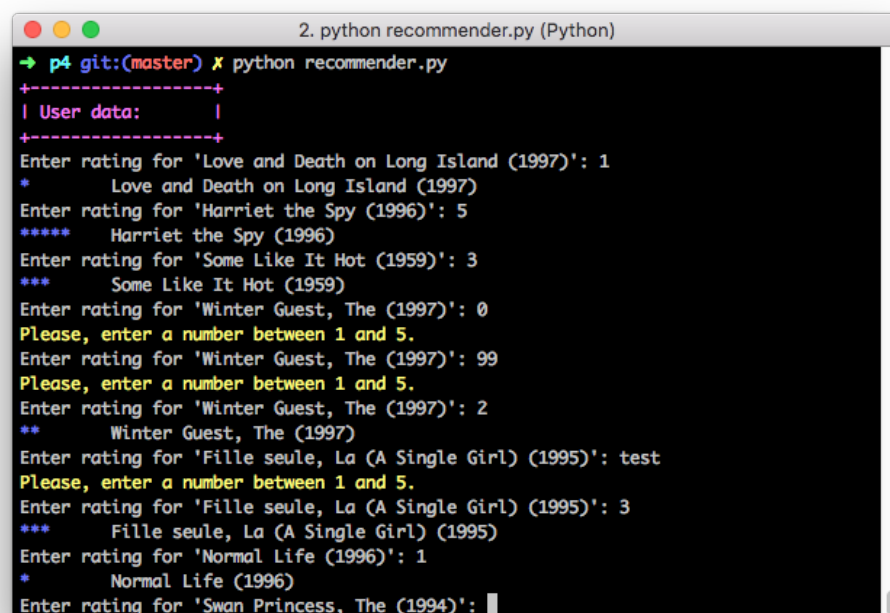
$$\hat{r}(u, i) = \bar{r}(u) + \frac{sim(u, v)(r(v, i) - \bar{r}(v))}{\sum |sim(u, v)|}$$

Donde $r(v, i)$ es la valoración del usuario v a la película i , $\bar{r}(u)$ la valoración media del usuario u y $sim(u, v)$ la similitud entre los usuarios u y v calculada anteriormente.

Por último se han seleccionado las n películas con mayor valoración siempre y cuando dicha valoración sea superior a 4 estrellas.

Como curiosidad hay que notar que el algoritmo recomienda algunas películas por encima de las 5 estrellas en base a la afinidad con otros vecinos. Se ha dejado así porque evidentemente una valoración por encima de 5 significa una mejor recomendación aunque en el código tenemos la posibilidad de limitar las valoraciones a 5 estrellas.

Para mejorar la interacción del programa se le han agregado colores a la salida en pantalla, además se permiten dos modos, el modo interactivo



```
2. python recommender.py (Python)
→ p4 git:(master) X python recommender.py
+-----+
| User data: |
+-----+
Enter rating for 'Love and Death on Long Island (1997)': 1
*      Love and Death on Long Island (1997)
Enter rating for 'Harriet the Spy (1996)': 5
***** Harriet the Spy (1996)
Enter rating for 'Some Like It Hot (1959)': 3
***      Some Like It Hot (1959)
Enter rating for 'Winter Guest, The (1997)': 0
Please, enter a number between 1 and 5.
Enter rating for 'Winter Guest, The (1997)': 99
Please, enter a number between 1 and 5.
Enter rating for 'Winter Guest, The (1997)': 2
**      Winter Guest, The (1997)
Enter rating for 'Fille seule, La (A Single Girl) (1995)': test
Please, enter a number between 1 and 5.
Enter rating for 'Fille seule, La (A Single Girl) (1995)': 3
***      Fille seule, La (A Single Girl) (1995)
Enter rating for 'Normal Life (1996)': 1
*      Normal Life (1996)
Enter rating for 'Swan Princess, The (1994)': 1
```

Figura 2.1: Ejecución interactiva

que iría preguntado la valoración que le queremos asignar a cada una de las películas seleccionada aleatoriamente así como el modo automático que asignaría las valoraciones de las películas del usuario de forma aleatoria.

Se ha hecho especial detalle en controlar las posibles excepciones del usuario a la hora de introducir las valoraciones, también se ha controlado la codificación de los archivos para no representar caracteres extraños.

Aunque se podría haber optimizado la ejecución de este programa usando la librería Numpy no ha sido necesario puesto que el volumen de datos es relativamente pequeño y se ha optado por usar los propios diccionarios de Python.

```
2. ernesto@bender: ~/Dropbox/Universidad/Master/Asignaturas/GIW/ugr_gestion_informa...
→ p4 git:(master) X python recommender.py auto
+-----+
| User data: |
+-----+
**      Swan Princess, The (1994)
****    Addams Family Values (1993)
***     Umbrellas of Cherbourg, The (Parapluies de Cherbourg, Les) (1964)
*       Strange Days (1995)
*****  Homage (1995)
*       Small Faces (1995)
**      Terminal Velocity (1994)
*       Maya Lin: A Strong Clear Vision (1994)
*       Empire Strikes Back, The (1980)
****    Game, The (1997)
***     Exotica (1994)
*       Unforgettable (1996)
*****  Bringing Up Baby (1938)
**      Hackers (1995)
***     Postino, Il (1994)
***     Braveheart (1995)
*****  Blue Angel, The (Blaue Engel, Der) (1930)
**      Star Trek: Generations (1994)
*****  Phat Beach (1996)
***     Midnight Dancers (Sibak) (1994)
+-----+
| Recommendations: |
+-----+
6.000000: How to Make an American Quilt (1995)
6.000000: Houseguest (1994)
6.000000: Friday (1995)
5.090478: Jaws 2 (1978)
5.011824: Little Princess, A (1995)
5.000000: Body Snatcher, The (1945)
5.000000: Striking Distance (1993)
5.000000: Winter Guest, The (1997)
5.000000: Safe Passage (1994)
5.000000: Ref, The (1994)
4.718339: Shadowlands (1993)
4.659684: Man Without a Face, The (1993)
4.659684: Bullets Over Broadway (1994)
4.659684: Forget Paris (1995)
4.511912: Shine (1996)
4.442915: Philadelphia (1993)
4.365548: Godfather, The (1972)
4.355785: Pinocchio (1940)
4.282835: Glory (1989)
4.267714: Shawshank Redemption, The (1994)
→ p4 git:(master) X
```

Figura 2.2: Ejecución automática

Capítulo 3

Manual

Para lanzar el programa de forma interactiva solo hay que lanzar el programa `recommender.py` sin ningún parámetro:

```
1 ./recommender.py
```

Si en cambio queremos asignar las valoraciones de forma aleatoria podemos llamar al programa con el parametro `auto`:

```
1 ./recommender.py auto
```


Capítulo 4

Bibliografía

- Documentación de Python: <http://docs.python.org>
- Transparencias de la asignatura: http://masteres.ugr.es/ing-informatica/pages/info_academica/guias/curso_actual/2semestre/optativas_ti2/gestion-de-informacion-en-web/
- StackOverflow: <https://stackoverflow.com>
- Creative Commons Share Alike 4.0: <https://creativecommons.org/licenses/by-sa/4.0/>

