Técnicas Avanzadas de Análisis de Datos: Sistemas Recomendadores

Daute Rodríguez Rodríguez

14 de marzo de 2020

1. Introducción y objetivo

Los **sistemas recomendadores** son herramientas software que permiten realizar sugerencias sobre ítems a usuarios. Ítem es el término que se utiliza para hacer referencia al elemento que es capaz de recomendar el sistema recomendador, desde películas y canciones hasta noticias y productos alimenticios.

El objetivo de esta práctica consiste en aplicar los conocimientos teóricos adquiridos sobre sistemas recomendadores. Para ello, se construirán sistemas recomendadores con el propósito de analizar, estudiar y comparar sus comportamientos.

El código desarrollado queda recogido en dos *Jupyter Notebooks*:

- *DataPreparation.ipynb*: Encargado de leer, preparar y reducir los datos utilizados para la obtención de los sistemas recomendadores.
- *RecommenderSystems.ipynb*: Este *Jupyter Notebook* contiene el código con el que se construyen y comparan los sistemas recomendadores.

2. Conjunto de datos y herramientas utilizadas

Se ha utilizado la librería *Surprise* [5] de *Python*. *Surprise* es un *toolkit* de *SciPy* que permite construir y analizar sistemas recomendadores a partir de datos de valoraciones de usuarios sobre ítems. Además, incorpora numerosos algoritmos para la obtención de sistemas recomendadores colaborativos.

El conjunto de datos utilizado para esta práctica es el denominado *Goodreads Datasets* [2, 6, 7], recoge las valoraciones que los usuarios de https://www.goodreads.com/ han realizado sobre los libros que han leído. Los datos se obtuvieron de manera anonimizada a partir de la información pública de las *shelves* (grupo de libros) de los usuarios de la plataforma a finales de 2017. El objetivo por tanto, consiste en obtener sistemas recomendadores capaces de sugerir a usuarios nuevos libros para su lectura.

Los datos disponibles aglutinan un total de 228.648.342 valoraciones, 829.529 autores, 876.145 usuarios y 2.360.650 libros. Dada la ingente cantidad de observaciones, resultó necesario reducir la dimensionalidad del conjunto de datos. En concreto, se crearon 3 conjuntos de datos al reducir los datos iniciales de una manera específica (con el *notebook DataPreparation.ipynb*), cada uno queda identificado por una cadena con la estructura *-X-Y-Z* dónde X indica la cantidad mínima de valoraciones permitidas por usuario, Y

la cantidad máxima de valoraciones permitidas por usuario y Z la cantidad mínima de valoraciones permitidas por libro. Los 3 conjuntos de datos utilizados son los siguientes:

- *-5-8-10*: 75.444 valoraciones, 13.031 usuarios, 1.101 libros y 903 autores.
- *-10-15-10*: 160.798 valoraciones, 13.819 usuarios, 1.955 libros y 1.596 autores.
- *-20-25-10*: 200.981 valoraciones, 9.378 usuarios, 1.892 libros y 1407 autores.

Una valoración queda definida como una tripleta (u, i, r), indicando que el usuario u le asignó una valoración r al libro i. Las valoraciones toman un valor comprendido en el rango $[0, 5]$. Para cada uno de los libros se dispone de la siguiente información:

- Título
- Formato
- Tamaño en páginas
- Año de publicación
- Géneros

Cabe destacar que la información sobre los géneros de cada libro se extrajo de manera difusa a partir de las clasificaciones más comunes que hacían los usuarios al almacenar los libros en *shelves*. Existen datos de un total de 17 géneros.

3. Sistemas recomendadores construidos

En esta sección del informe se presentan los distintos sistemas recomendadores construidos a partir de los datos de las valoraciones de los usuarios de *Goodreads*.

3.1. Sistema recomendador base

Con el propósito de poder comparar la bondad de los sistemas construidos se optó por implementar un recomendador base que devolviera como estimación de la valoración la media del total de valoraciones para cada par usuario-ítem (ecuación 1). En teoría, cualquier otro sistema recomendador debería mostrar un comportamiento superior al que muestre el recomendador base.

$$\hat{r}_{ui} = \mu \tag{1}$$

3.2. Sistemas recomendadores colaborativos

A continuación se presentan los sistemas recomendadores colaborativos construidos con la librería *Surprise*. Cabe mencionar que los valores de los hiperparámetros utilizados en la construcción de los sistemas fueron los valores por defecto.

3.2.1. *KNNWithMeans*

A modo de primera aproximación, se decidió construir un sistema recomendador ítem-ítem básico haciendo uso de la similitud coseno como medida de similitud entre ítems, teniendo en cuenta la media de valoraciones de cada ítem y un tamaño de vecindad igual a 40. La predicción ($\hat{r}_{ui}$) que realiza el usuario u sobre el ítem i queda definida por la ecuación 2:

$$\hat{r}_{ui} = \mu_i + \frac{\sum\limits_{j \in N_u^k(i)} sim(i,j) \cdot (r_{uj} - \mu_j)}{\sum\limits_{j \in N_u^k(i)} sim(i,j)} \tag{2}$$

3.2.2. *SVD*

Para el siguiente sistema recomendador se hizo uso del algoritmo de factorización de matrices *SVD*. La predicción queda definida por la ecuación 3:

$$\hat{r}_{ui} = \mu + b_u + b_i + q_i^T p_u \tag{3}$$

Dónde μ indica la media del total de valoraciones, b_u y b_i representan el bias o tendencia del usuario y el ítem respectivamente y el producto de los factores q_i^T y p_u simboliza la interacción usuario-ítem. Para más detalles, consulte el artículo *Matrix factorization techniques for recommender systems* [3].

3.2.3. *SlopeOne*

El tercer sistema recomendador colaborativo construido se basa el el algoritmo *SlopeOne* [4]. La predicción $\hat{r}_{ui}$ se computa siguiendo las ecuaciones 4 y 5, dónde $R_i(u)$ queda conformado por el conjunto de ítems relevantes (ítems valorados por el usuario y por otros usuarios que han valorado el ítem i) y $dev(i,j)$ se define como la diferencia media entre las valoraciones de los ítems i y j.

$$\hat{r}_{ui} = \mu + \frac{1}{|R_i(u)|} \sum_{j \in R_i(u)} dev(i,j) \tag{4}$$

$$dev(i,j) = \frac{1}{|U_{ij}|} \sum_{u \in U_{ij}} r_{ui} - r_{uj} \tag{5}$$

3.2.4. *CoClustering*

El último de los sistemas recomendadores colaborativos construidos se basa en un algoritmo de *co-clustering* [1]. En este algoritmo, tanto los usuarios como los ítems son asignados a *clusters* específicos (C_u, C_i) y a *co-clusters* (C_{ui}). La predicción dado un usuario e ítem especificos queda definida por la ecuación 6:

$$\hat{r}_{ui} = \overline{C_{ui}} + (\mu_u - \overline{C_u}) + (\mu_i - \overline{C_i}) \tag{6}$$

Dónde $\overline{C_{ui}}$ se define como la valoración media del *cluster* C_{ui} y $\overline{C_u}, \overline{C_i}$ son las valoraciones medias de los *clusters* del usuario u y el ítem i respectivamente.

3.3. Sistema recomendador basado en contenido

Gracias a la naturaleza de los datos de los que se dispone sobre los libros resultó posible obtener un sistema recomendador basado en contenido simple. Se optó por incorporar el conocimiento sobre los **géneros y autores** de los libros al proceso de recomendación, teniendo en cuenta las **preferencias del usuario**. Antes de continuar, resulta necesario introducir la notación que se utilizará para describir cómo realiza las predicciones el sistema:

- p_u^g: Preferencias de género de los libros para el usuario u. Consiste en un vector en el que cada posición representa un género. El valor de cada posición está comprendido en el intervalo $[0, 5]$ y representa la valoración media que ha dado el usuario u a libros de dicho género.
- p_u^a: Preferencias de autores para el usuario u. Se trata de una estructura que mapea un identificador de autor a un valor flotante, que simboliza la media de valoraciones realizadas por el usuario u a libros del autor en cuestión.
- f_i^g: Géneros del libro i. Consiste en un vector en el que cada posición representa un género, en las posiciones correspondientes a los géneros del libro habrá un 1 por valor, y en las demás posiciones un 0.
- f_i^a: Vector con los autores del libro i.

La ecuación 7 muestra el cálculo que se ha de realizar para obtener la predicción de la valoración que dará un determinado usuario a un libro:

$$\hat{r}_{ui} = w_g \cdot (p_u^g \cdot f_i^g) + w_a \cdot (\frac{1}{|f_i^a|} \sum_{a \in f_i^a} p_u^a) \tag{7}$$

El valor predicho lo componen dos sumandos: afinidad de géneros y afinidad de autores. Cada sumando está multiplicado por un peso, la suma de ambos pesos ha de ser igual a 1 ($w_a, w_g \in [0, 1] / w_a + w_g = 1$). La afinidad de géneros se define como el producto vectorial entre el vector de preferencias de géneros del usuario (p_u^g) y el vector de géneros del libro (f_i^g). La afinidad de autores se define como la media de valoraciones medias que ha realizado el usuario u a libros de los autores de i. Además de lo establecido, se han de tener varias consideraciones en cuenta:

- En caso de que un usuario no haya realizado valoraciones de libros de algún género, el valor correspondiente del vector de preferencias de géneros del usuario queda establecido como la media de valoraciones del usuario.
- En caso de que un usuario no haya realizado valoraciones de libros de un cierto autor, el valor de p_u^a se establece a la media de valoraciones del usuario.
- Si se va a realizar una predicción y el sistema no conoce al usuario, se devuelve la valoración media del total de valoraciones conocidas.
- A pesar de que la afinidad de autores también se puede definir como un producto vectorial, no se creyó conveniente por el gran tamaño que tendrían los supuestos vectores de preferencias de usuario y el vector de autores de cada libro.

A partir de lo anterior, es posible intuir que el entrenamiento del sistema consiste en extraer los vectores de géneros de los libros y las preferencias de cada usuario. Una de las ventajas que aporta esta aproximación reside en el hecho de poder mantener el sistema en constante actualización de manera simple cuando se produzcan nuevas valoraciones (consistiría en actualizar las valoraciones medias de los usuarios). Por otro lado, el sistema realizará recomendaciones de libros cuyos géneros o autores ya gustan a los usuarios por lo que las recomendaciones no dejan cabida a la exploración de nuevas posibilidades.

3.4. Sistema recomendador híbrido paralelizado ponderado

Como última aproximación se decidió construir un sistema recomendador híbrido paralelizado ponderado que utilizara un sistema colaborativo y el basado en contenido. La ecuación 8 muestra cómo el sistema híbrido realiza las predicciones:

$$\hat{r}_{ui} = w_{col} \cdot \hat{r}_{ui}^{col} + w_{con} \cdot \hat{r}_{ui}^{con} \tag{8}$$

Dónde cada una de las predicciones que realizan los sistemas recomendadores internos ($\hat{r}_{ui}^{col}, \hat{r}_{ui}^{con}$) es multiplicada por un peso, y la suma de ambos ha de ser igual a 1 ($w_{col}, w_{con} \in [0,1] / w_{col} + w_{con} = 1$).

4. Estudio experimental y resultados

A lo largo de esta sección se presentan los experimentos realizados y resultados obtenidos con los sistemas recomendadores construidos. Como ya se ha comentado con anterioridad, se han usado 3 conjuntos de datos de diferentes dimensiones (*-5-8-10*, *-10-15-10* y *-20-25-10*) con el objetivo de comprobar si se produce alguna variación del comportamiento de los sistemas recomendadores implementados.

Antes de comenzar conviene comentar que la métrica que se utilizará para medir la calidad de los recomendadores es la *raíz del error cuadrático medio (RMSE)*. En concreto, se aplicará validación cruzada de 5 pliegues por lo que se realizará una media del valor *RMSE*.

4.1. Comparativa de sistemas recomendadores colaborativos

En el cuadro 1 se presentan los valores medios obtenidos de la métrica *RMSE* para cada sistema recomendador colaborativo. De dichos valores, es posible extraer dos conclusiones:

1. La densidad de la matriz de valoraciones interviene directamente en la calidad del sistema recomendador. Es posible apreciar cómo los valores disminuyen de manera significativa de izquierda a derecha, es decir, comenzando en el conjunto de datos más pequeño y terminando en el de mayor tamaño.

2. El sistema recomendador que mejor comportamiento parece tener es el construido con el algoritmo de factorización de matrices *SVD*. Se trata del recomendador que menor *RMSE* obtuvo para los 3 conjuntos de datos estudiados. Por este motivo, será el que se utilice en el sistema recomendador híbrido paralelizado ponderado, junto al basado en contenido.

Sistema Recomendador	**Conjunto de datos**		
	-5-8-10	*-10-15-10*	*-20-25-10*
Base	1.5059	1.3646	1.2720
KNNWithMeans	1.4589	1.2716	1.1746
SVD	1.3883	1.2467	1.1694
SlopeOne	1.5042	1.2993	1.1843
CoClustering	1.4725	1.3045	1.2073

Cuadro 1: Valores de la media de la métrica *RMSE* de cada sistema recomendador colaborativo para cada conjunto de datos

4.2. Optimización de los parámetros w_a y w_g

Los valores a explorar de los parámetros w_a y w_g del sistema recomendador basado en contenido quedan determinados por la ecuación 9:

$$w_a = i \cdot 0,1 \quad w_g = 1 - w_a \quad / \quad i = 0, \ldots, 10 \tag{9}$$

En el cuadro 2 pueden encontrarse los valores medios de la métrica *RMSE* para los distintos pares de valores de w_a y w_g. Tal y como puede apreciarse, para los conjuntos de datos de mayor tamaño el valor mínimo se alcanza cuando el sistema recomendador queda configurado con los valores 0.4 y 0.6 para w_a y w_g respectivamente. Para el conjunto de datos de menor tamaño, los valores óptimos para los pesos son similares (0.5 y 0.5). Aplicando la regla de la mayoría, se optó por usar los valores óptimos para los conjuntos de datos de mayor tamaño, es decir, 0.4 para w_a y 0.6 para w_g.

Wa	**Wg**	**Conjunto de datos**		
		-5-8-10	*-10-15-10*	*-20-25-10*
0.0	1.0	1.4757	1.2901	1.2117
0.1	0.9	1.4450	1.2728	1.1976
0.2	0.8	1.4210	1.2605	1.1875
0.3	0.7	1.4041	1.2533	1.1815
0.4	0.6	1.3945	1.2514	1.1795
0.5	0.5	1.3924	1.2547	1.1817
0.6	0.4	1.3978	1.2633	1.1880
0.7	0.3	1.4107	1.2768	1.1983
0.8	0.2	1.4307	1.2954	1.2126
0.9	0.1	1.4578	1.3187	1.2308
1.0	0.0	1.4913	1.3464	1.2525

Cuadro 2: Valores de la media de la métrica *RMSE* del sistema recomendador basado en contenido con distintos valores de los parámetros w_a y w_g para cada conjunto de datos

4.3. Optimización de los parámetros w_{col} y w_{con}

Tal y como se ha comentado con anterioridad, el sistema recomendador híbrido queda conformado por el mejor sistema recomendador colaborativo de entre los estudiados (obtenido con el algoritmo *SVD*) y por el sistema recomendador basado en contenido. A cada

sistema interno se le ha de asignar un peso de manera que la predicción del híbrido quede determinada por la suma ponderada de las predicciones de los sistemas internos.

Los valores a explorar de los parámetros w_{col} y w_{con} del sistema recomendador híbrido paralelizado ponderado quedan determinados por la ecuación 10:

$$w_{col} = i \cdot 0,1 \quad w_{con} = 1 - w_{col} \quad / \quad i = 0, \ldots, 10 \tag{10}$$

El cuadro 3 muestra cómo para los conjuntos de datos de menor tamaño, la configuración óptima de los parámetros consiste en asignarles el mismo peso a ambas predicciones. Para el conjunto de datos *-20-25-10*, el valor mínimo de la métrica *RMSE* se obtiene al asignar un peso de 0.6 a la predicción del sistema *SVD* y 0.4 al basado en contenido. Al igual que con la optimización de los parámetros w_a y w_g del sistema recomendador basado en contenido, y aplicando la regla de la mayoría se seleccionarán los pesos 0.5 y 0.5 para los parámetros w_{col} y w_{con}.

W_{col}	W_{con}	**Conjunto de datos**		
		-5-8-10	*-10-15-10*	*-20-25-10*
0.0	1.0	1.39452	1.25144	1.17950
0.1	0.9	1.37984	1.24228	1.17054
0.2	0.8	1.36814	1.23506	1.16332
0.3	0.7	1.35958	1.22970	1.15788
0.4	0.6	1.35414	1.22634	1.15418
0.5	0.5	1.35194	1.22490	1.15232
0.6	0.4	1.35292	1.22544	1.15226
0.7	0.3	1.35712	1.22794	1.15396
0.8	0.2	1.36444	1.23234	1.15744
0.9	0.1	1.37488	1.23862	1.16260
1.0	0.0	1.38830	1.24672	1.16938

Cuadro 3: Valores de la media de la métrica *RMSE* del sistema recomendador híbrido paralelizado ponderado con distintos valores de los parámetros w_{col} y w_{con} para cada conjunto de datos

4.4. Comparativa de los sistemas recomendadores obtenidos

A continuación se presentan los valores medios de la métrica *RMSE* obtenidos por cada sistema recomendador para los distintos conjuntos de datos utilizados (cuadro 4 y figuras 1, 2 y 3). Para la obtención de estos valores se configuró el sistema híbrido y el basado en contenido con los parámetros óptimos. Tal y como puede apreciarse, el mejor comportamiento para los 3 conjuntos de datos estudiados lo demuestra el sistema recomendador híbrido paralelizado ponderado.

Cabe destacar que el recomendador *SVD* obtuvo el segundo lugar para los 3 conjuntos de datos. El recomendador basado en contenido obtuvo el tercer lugar para los conjuntos de datos *-5-8-10* y *-10-15-10*. Para el conjunto de datos *-20-25-10*, la tercera posición la ocupa el recomendador *KNNWithMeans* y el cuarto lugar el recomendador basado en contenido. Tal y cómo cabría esperar, el recomendador base ocupa la última posición para los 3 conjuntos de datos.

Sistema Recomendador	**Conjunto de datos**		
	-5-8-10	*-10-15-10*	*-20-25-10*
Base	1.5059	1.3646	1.2720
KNNWithMeans	1.4589	1.2716	1.1746
SVD	1.3883	1.2467	1.1694
SlopeOne	1.5042	1.2993	1.1843
CoClustering	1.4725	1.3045	1.2073
Basado en contenido	1.3924	1.2547	1.1817
Híbrido	1.3519	1.2249	1.1523

Cuadro 4: Valores de la media de la métrica *RMSE* de cada sistema recomendador obtenido para cada conjunto de datos

Por último, se debe recordar que los hiperparámetros de los algoritmos de construcción de sistemas recomendadores colaborativos (*SVD*, *KNNWithMeans*, *CoClustering* y *SlopeOne*) se establecieron a los valores por defecto. Lo más probable es que si dichos parámetros se ajustaran, sus comportamientos mejorasen notablemente.

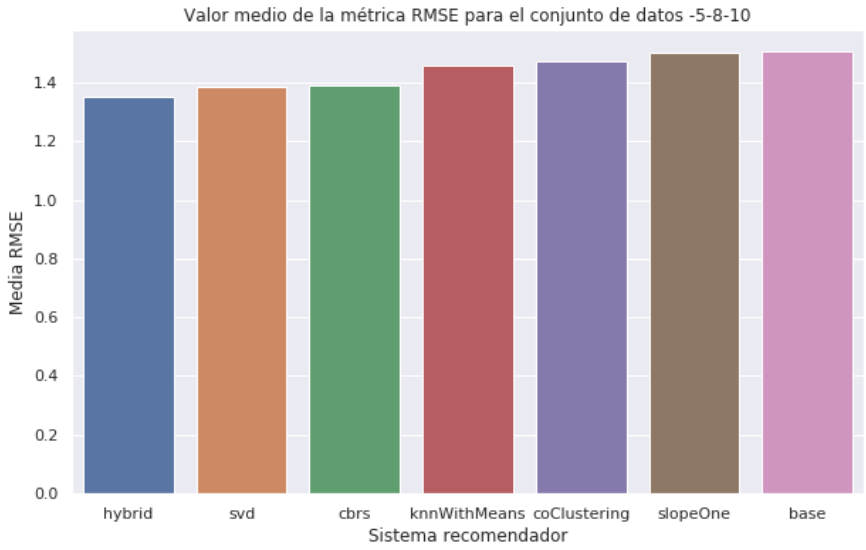

Figura 1: Comparación del valor *RMSE* medio de los sistemas recomendadores para el conjunto de datos -5-8-10

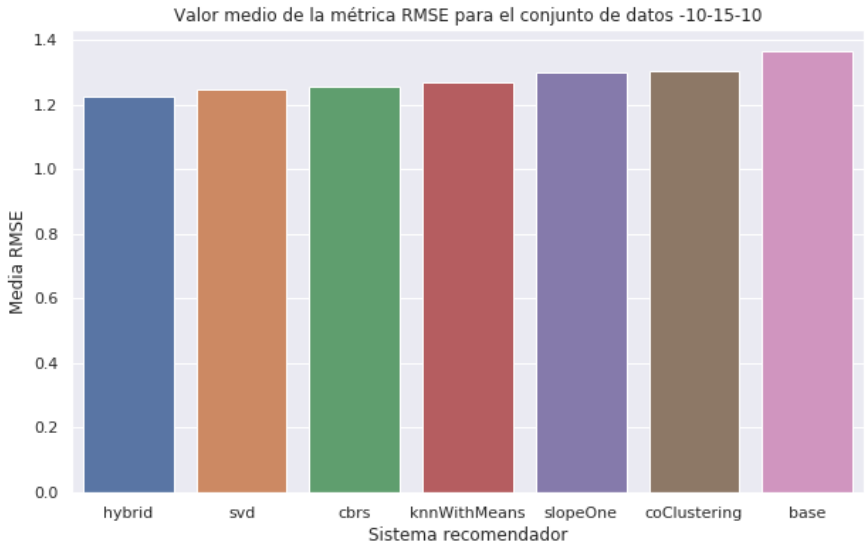

Figura 2: Comparación del valor *RMSE* medio de los sistemas recomendadores para el conjunto de datos -10-15-10

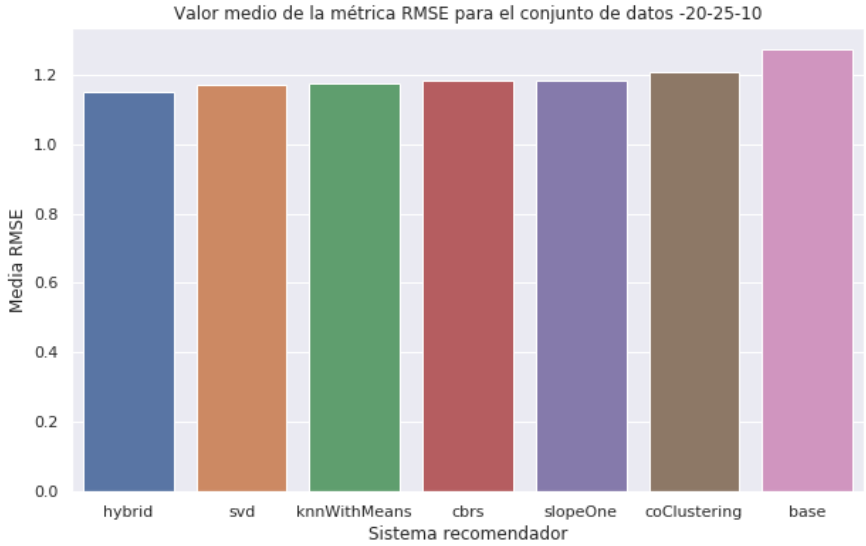

Figura 3: Comparación del valor *RMSE* medio de los sistemas recomendadores para el conjunto de datos -20-25-10

5. Conclusiones y posibles líneas futuras

En este apartado del informe se presentan las conclusiones alcanzadas y una serie de posibles líneas futuras. Tal y cómo se ha podido observar en el estudio experimental, el mejor sistema recomendador de entre los estudiados ha resultado ser el híbrido paralelizado ponderado que incorpora el basado en contenido y el *SVD*. Mediante el uso de la librería *Surprise* ha sido posible llevar a la práctica de manera sencilla algunos de los conceptos teóricos adquiridos durante la asignatura. A pesar de la simpleza del sistema

recomendador basado en contenido construido, se ha podido apreciar que demuestra un comportamiento notable frente a los demás evaluados, aunque también hay que resaltar que para la construcción de los demás sistemas recomendadores se utilizaron los valores por defecto para los hiperparámetros.

Como posibles líneas futuras del trabajo realizado para esta práctica se proponen las siguientes cuestiones:

1. Realizar un ajuste de los sistemas recomendadores colaborativos implementados optimizando sus hiperparámetros. Tal y como se ha comentado con anterioridad, ésto podría suponer una notable mejora en su comportamiento.

2. La librería *Surprise* incluye algunos algoritmos más de construcción de sistemas recomendadores colaborativos que deberían ser probados.

3. El sistema recomendador basado en contenido implementado es bastante sencillo y solo tiene en cuenta autores y géneros. Resultaría interesante añadir algunos aspectos cómo el tipo de escritura o al público al que está orientado cada libro (niños, jóvenes, personas mayores ...)

4. Dado que el sistema recomendador basado en contenido recomienda según los gustos conocidos de los usuarios, estaría bien obtener un recomendador que diera recomendaciones de ítems distintos a los ya valorados por el usuario. Combinando ambos en un recomendador híbrido resultaría posible obtener un buen compromiso entre la exploración de nuevos gustos y los gustos ya consolidados.

5. Por último, el incorporar información sobre el usuario al que se le va a realizar la recomendación podría ayudar a realizar recomendaciones más personalizadas y menos genéricas.

Referencias

[1] Thomas George and Srujana Merugu. A scalable collaborative filtering framework based on co-clustering. In *Fifth IEEE International Conference on Data Mining (ICDM'05)*, pages 4–pp. IEEE, 2005.

[2] Goodreads datasets. `https://sites.google.com/eng.ucsd.edu/ucsdbookgraph/home`. Accessed: 13-03-2020.

[3] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[4] Daniel Lemire and Anna Maclachlan. Slope one predictors for online rating-based collaborative filtering. In *Proceedings of the 2005 SIAM International Conference on Data Mining*, pages 471–475. SIAM, 2005.

[5] Surprise - a python scikit for recommender systems. `http://surpriselib.com/`. Accessed: 13-03-2020.

[6] Mengting Wan and Julian McAuley. Item recommendation on monotonic behavior chains. In *Proceedings of the 12th ACM Conference on Recommender Systems*, pages 86–94, 2018.

[7] Mengting Wan, Rishabh Misra, Ndapa Nakashole, and Julian McAuley. Fine-grained spoiler detection from large-scale review corpora. *arXiv preprint arXiv:1905.13416*, 2019.