

Recomendaciones de libros

Recuperación de Información y Recomendaciones en la Web



Docente

Libertad Tansini

Grupo 8

Giuliano Severi 4.671.833-6

Federico Fioritto Peraza 4.646.654-7

Florencia Mercant Morixe 4.629.959-4

Mateo Nicolás Nogueira Fandiño 4.830.584-6

Índice

1. Introducción	3
2. Problema	3
3. Descripción de la solución	3
3.1 Proceso de creación	4
3.2 Filtrado colaborativo	5
4. Funcionalidades	5
5. Evaluación y resultados	8
6. Conclusiones	8
7. Trabajos a futuro	8
8. Referencias	10

1. Introducción

Actualmente existen en la web muchas herramientas para obtener reviews y puntuaciones de distintos productos comerciales, como lo son películas, música o videojuegos. Sin embargo, cuando se trata de libros no podemos encontrar tan fácilmente herramientas similares que sirvan como apoyo a la hora de decidir cuál será el siguiente libro a leer.

Resultó entonces interesante para el grupo abordar el problema de recomendaciones de libros, enfocado además sobre lo que es el mercado local. Adicionalmente apreciamos que es una solución poco o analizada en cursos anteriores de la materia.

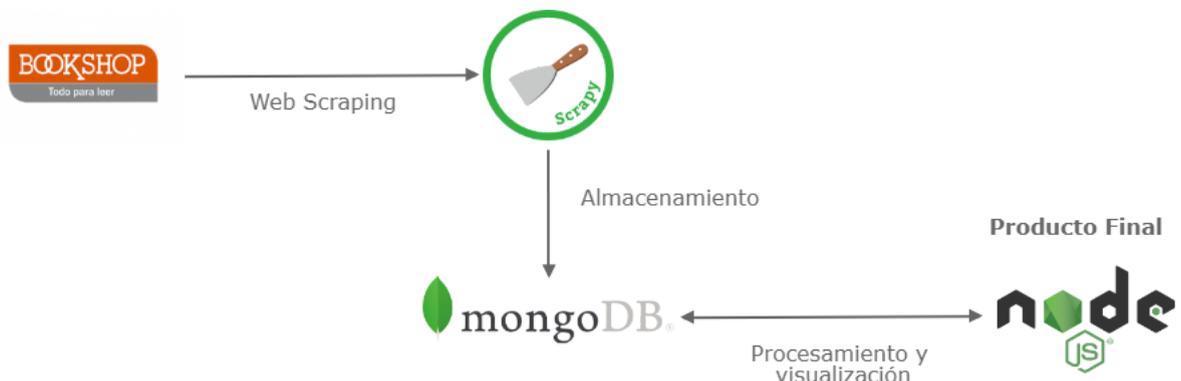
2. Problema

El problema a resolver entonces es facilitar la decisión de los usuarios sobre cuál libro escoger para leer. Esto se hará por medio de recomendaciones utilizando como insumo información que los usuarios brindan a la aplicación y datos obtenidos de páginas web de librerías uruguayas.

3. Descripción de la solución

Como se ha mencionado, se presenta una aplicación que ayuda a los usuarios a encontrar libros que podrían llegar a gustarles en base a otros usuarios con gustos parecidos. Para esto se utilizan técnicas de filtrado colaborativo. Además la aplicación cuenta con una base de datos con libros, usuarios y puntuaciones de los usuarios hacía algunos libros. Luego utilizando el filtrado colaborativo, nuevos libros son recomendados a los usuarios. Además para mitigar el problema que existe cuando un usuario acaba de registrarse en la aplicación (entonces no se tiene información suficiente como para realizarle recomendaciones) se recomiendan los libros más populares con los que cuenta la aplicación.

3.1 Proceso de creación



Los datos referentes a libros se obtuvieron por medio de la técnica web scraping ¹ sobre el sitio web de Bookshop Uruguay ² utilizando la librería Scrapy ³ para python. Específicamente para cada libro se obtuvo: Título, autor, resumen, identificador (en Bookshop), precio y url de la imagen. Se obtuvieron 730 libros de esta manera.

Los usuarios del sistema fueron generados con un script que carga 500 usuarios. Para lograr con cierta aleatoriedad de los *likes* de estos usuarios sobre los libros, cada usuario tiene un 30% de probabilidad de que le guste un libro dado.

El almacenamiento se realizó en una base de datos no relacional MongoDB ⁴, ya que el modelo de la aplicación cuenta con pocas entidades y pocas relaciones, además de que la cantidad de registros de cada entidad es grande. Se buscó crear una aplicación sencilla, que escale cuando se cuentan con cientos de miles de registros y que cumpla el objetivo propuesto. Se utilizarán 2 bases de datos alojadas en mlab ⁵ con el fin de proveer una rápida configuración inicial, y para poder probar la aplicación con varios usuarios. Una de las bases solo contiene los libros obtenidos de Bookshop y se utilizará para mostrar los casos básicos de filtrado. La otra se cargó con unos 500 usuarios además de los libros, y se implementó un algoritmo para que estos usuarios creados le dieran *like* a los libros cargados. De esta forma se obtuvo un total de *likes* repartidos entre los libros de la aplicación. Luego se corre el algoritmo de filtrado colaborativo para darle a cada uno de estos usuarios recomendaciones.

La aplicación final fue construida con NodeJS ⁶ y el framework express, y utilizando una biblioteca de filtrado colaborativo de dicho lenguaje ⁷.

3.2 Filtrado colaborativo

El filtrado colaborativo consiste en hacer predicciones sobre los intereses o comportamientos futuros de un sujeto. Como se mencionó anteriormente, nosotros

utilizamos el filtrado para recomendar libros que a una persona le podrían gustar. Si tenemos que las personas A y B leyeron un mismo libro X y además B leyó Y podemos predecir que a A le va a gustar Y. Para realizar esta tarea utilizamos una implementación open source de la empresa Asymmetrik llamada Akin ⁸.

Esta implementación considera como más importante la actividad más nueva de un usuario, es decir, es más importante lo que indicó que le gustaba más recientemente.

Una vez que se tiene los libros que a un usuario le gustan, se tiene un vector del largo de la cantidad de libros en el sistema y la entrada i representa si al usuario le gusta o no el libro i para cada usuario. Con estos vectores se crea una matriz M donde cada fila representa un usuario, cada columna un libro y la entrada i,j si al usuario i le gusta el libro j .

Entonces se puede calcular la similitud coseno entre un usuario y todos los demás. Luego se multiplica cada fila por la similitud coseno del usuario al que se le desea recomendar con el usuario dueño de esa fila y se suman las columnas. Se genera así un puntaje para cada libro donde un puntaje mayor indica que es más probable que al usuario le guste.

Un problema que tiene este sistema de recomendaciones es que antes de poder recomendar algo a un usuario nuevo es necesario tener datos previos, de libros que otros usuarios marcaron que les gustaban y algunos libros que el usuario nuevo marque que le gusten de forma que la similitud coseno sea distinta de 0 para algún usuario.

4. Funcionalidades

Sign Up

Permite a un usuario registrarse en la aplicación, obteniendo de esta manera todos los servicios que la misma brinda.

Log in / Log Out

Permiten a un usuario registrado loguearse y desloguearse en la aplicación.

Listado de libros

Cualquier visitante de la página puede acceder en la pestaña *Libros* a un listado del total de libros que contiene el sitio. Cada elemento de la lista permite acceder a información detallada del mismo.

Libros

- [100 FACTS EXPLORERS \(KELLY, MILES\)](#)
- [100 PREGUNTAS SOBRE EDUCACION ESCOLAR \(CESPEDES, AMANDA\)](#)
- [1001 ARMAS DE FUEGO \(OTROS\)](#)
- [1001 DINOSAURIOS \(OTROS\)](#)
- [1001 MARAVILLAS DEL MUNDO \(OTROS\)](#)
- [1001 MOTOS \(OTROS\)](#)
- [1001 MOVIES YOU MUST SEE BEFORE YOU DIE \(Steven Jay Schneider\)](#)
- [1001 TRENES \(OTROS\)](#)
- [1101 RECETAS VEGETARIANAS \(\(VARIOS\)\)](#)
- [111 SECRETOS FENG SHUI \(PÉRSICO, LUCRECIA\)](#)
- [200 COCTELES FABULOSOS \(OTROS\)](#)
- [30 MITOS DE LA HISTORIA RECIENTE \(LEICHT, FEDERICO\)](#)
- [50 HISTORIAS DE MONSTRUOS \(OTROS\)](#)
- [500 PREGUNTAS Y RESPUESTAS SOBRE LOS DINOSAURIOS \(\(OTROS\)\)](#)
- [75 CONSEJOS PARA SER POPULAR \(FRISA, MARIA\)](#)
- [A DAY IN LONDON \(FICHTE, ANDRÉ\)](#)
- [A TO Z PIZZARRA MAGNATAB \(OTROS\)](#)
- [A TRAVES DE LAS CANCIONES \(FRENKEL, DIEGO\)](#)
- [ABACOLOR MAXI X12 \(Miniland\)](#)
- [ABREMENTE 9 - 10 AÑOS \(OTROS\)](#)
- [ACTIVITY TIMER- RELOJ EDUCATIVO \(Miniland\)](#)
- [ADIVINANZAS PARA ANTES DE DORMIR DE ANIMALES \(\(OTROS\)\)](#)
- [ADOBE PHOTOSHOP CS5 \(MACRO\)](#)
- [ADOLESCENCIA \(Alcuri, Alvaro\)](#)
- [AGENDA 2018 PAULO COELHO CARTONE AZUL \(COELHO, PAULO\)](#)

• • •

Ver más populares

Un usuario logueado puede ver en la pestaña *Recomendaciones* un listado con los libros más populares del sitio. Además de oficiar como una recomendación más en todo momento, el principal objetivo de esta funcionalidad es brindar a nuevos usuarios de la aplicación recomendaciones cuando no se tiene información de los mismos.

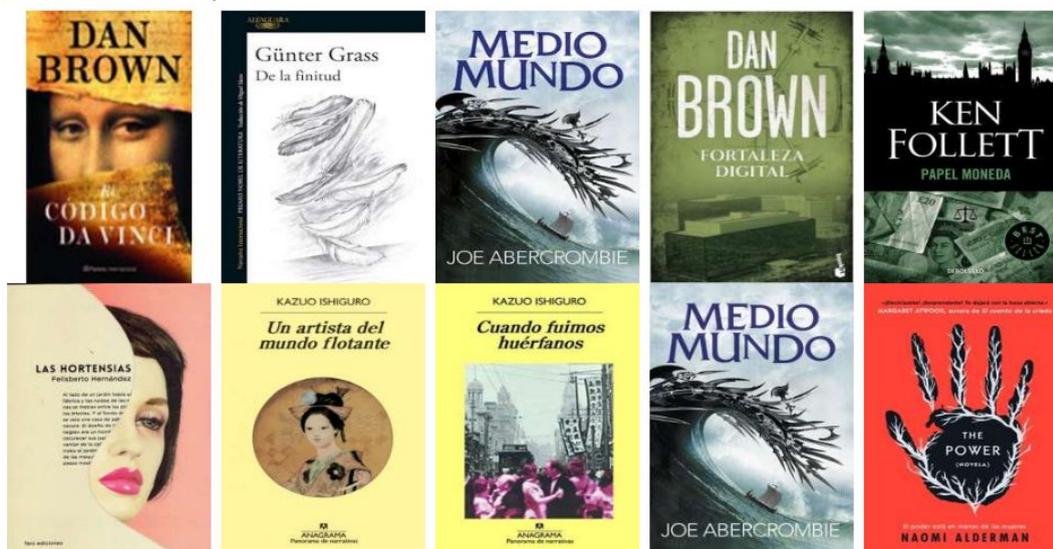
Más populares



Ver mis recomendaciones

Al igual que los libros más populares, esta funcionalidad se encuentra en la pestaña *Recomendaciones*. Se trata de un listado de los libros recomendados para cada usuario puntualmente, obtenidos a partir del filtrado colaborativo.

Recomendaciones para tí



Ver libro

Para cada libro del sitio se puede consultar por información detallada del mismo. Específicamente se proporciona título, autor, breve descripción, precio y una imagen con la tapa del producto.

[Inicio](#) / [Libros](#) / [Recomendaciones](#) / [Log out](#)

ATLAS ILUSTRADO DE FERRARI

OTROS

La historia de los Ferrari para carretera significa el triunfo de la belleza y la perfección técnica. Desde el elegante biplaza 166 MM Barchetta de los primeros tiempos hasta el moderno 458 Italia, que incorpora las últimas tecnologías derivadas de la Fórmula 1, cada modelo Ferrari es una obra maestra de brillante técnica y diseño innovador.

\$930

[Ya no me gusta](#)



Me gusta / Ya no me gusta

Todo usuario logueado puede indicar si un libro le gusta o no. Esta funcionalidad se encuentra al visitar el perfil de un libro en particular. La decisión no es permanente, el usuario tiene la posibilidad de cambiar de opinión en todo momento.

5. Evaluación y resultados

En términos generales nos encontramos conforme con el resultado para lo que es alcance del proyecto dentro del curso. Las herramientas para web scraping y filtrado colaborativo fueron muy intuitivas y no presentaron ningún problema a la hora de ser utilizadas.

Como detalle se puede mencionar que un principio se tuvo la idea de complementar la información de los libros obtenida de la web de la librería con información devuelta por la API de google books⁹ pero nos encontramos con una falla en dicha API: Las consultas por ISBN no funcionan correctamente, investigando un poco sobre el tema se corroboró el problema el cual parece existir desde hace tiempo. Por esta razón fue descartada la utilización de la API.

6. Conclusiones

Se logró llegar a una aplicación que muestra en una implementación reducida pero que permite ver claramente el problema del filtrado colaborativo. Se le ofrecen libros a los usuarios los cuales indican si les gustan o no y en base a eso reciben recomendaciones.

Durante la realización de este trabajo nos encontramos con diversos problemas. Principalmente obtener los libros y cómo realizar el filtrado colaborativo, dado que el equipo ya tenía experiencia con los lenguajes de programación utilizados.

Aprendimos entonces a utilizar la herramienta scrapy la cual nos permitió obtener de forma sencilla una gran cantidad de libros en formato JSON fácilmente persistibles en una base de datos MongoDB. Para el filtrado colaborativo encontramos una librería ya implementada como se dijo anteriormente así que fue solamente usar la API que proveía.

Finalmente comprendimos la importancia de los datos previos para este tipo de algoritmos dado que es necesaria una pequeña cantidad de datos para poder funcionar. Esto supone un problema a nuevos sistemas que recién comienzan.

7. Trabajos a futuro

Como se dijo en las conclusiones, la implementación fue reducida por lo que podría realizarse una versión con más funcionalidades.

En particular, podrían actualizarse los libros dinámicamente en vez de ser cargados una sola vez como ahora. Adicionalmente, se podría escalar la aplicación de forma natural realizando web scraping sobre más sitios, esto requeriría un reajuste en el módulo encargado del scraping ya que no todas las páginas siguen la misma estructura.

Además, dado que no presentó dificultad obtener el precio de los libros, un aspecto interesante a agregar sería realizar una búsqueda del sitio con el mejor costo y a su vez proporcionar un mecanismo para que el usuario ya pueda realizar la compra.

Por otro lado, se podría mantener por cada usuario una lista de los libros *leídos*, *a leer* y *actualmente leyendo*.

También sería factible utilizar un sistema de puntuación en una escala del 1 al 5 o 1 al 10 por ejemplo, en lugar de 0 y 1, otorgando mayor flexibilidad a las opiniones de los usuarios. Inclusive al momento de puntuar cada libro se podría permitir al usuario escribir una reseña expresando su opinión de dicho libro.

Otra funcionalidad que podría agregarse es la de poder filtrar libros por título, autor, género, año y/o nacionalidad dado que actualmente aparecen todos listados alfabéticamente.

Una idea que se podría implementar cuando la aplicación esté más avanzada sería el poder tener *amigos lectores* en donde cada usuario podría ver los libros y opiniones de sus *amigos lectores* y así poder obtener ideas sobre futuros títulos a leer. También se podría implementar una forma de recomendarse libros entre *amigos lectores* de forma voluntaria por ejemplo mediante un chat. Otra idea interesante sería la de poder formar comunidades tales como *grupos de lectura* en donde un grupo de usuarios intercambian opiniones sobre un mismo libro.

Finalmente nos parece que sería interesante explorar algoritmos de filtrado colaborativos distintos al utilizado e implementarlos realizando comparaciones entre los distintos algoritmos intentando concluir cual parece funcionar mejor.

Se puede apreciar que hay muchos aspectos a desarrollar que enriquecerían fuertemente a la aplicación. Además cabe destacar que no existe por el momento una aplicación sobre recomendaciones de libros con tales características en el Uruguay e inclusive en el mundo hay muy pocas o ninguna aplicación de este estilo en particular para lectores de habla hispana.

8. Referencias

- [1] https://es.wikipedia.org/wiki/Web_scraping
- [2] <https://www.bookshop.com.uy/>
- [3] <https://scrapy.org/>
- [4] <https://www.mongodb.com/es>
- [5] <https://mlab.com/>
- [6] <https://nodejs.org/es/>
- [7] <https://github.com/Asymmetrik/akin>
- [8] <https://www.asymmetrik.com/recommendation-engine-analytics/>
- [9] <https://developers.google.com/books/>