Informe Final
Cuarto Proyecto

Sebastián Romero Castrillón

Febrero 2021

Acámica

Data Science

Objetivo

Desarrollar diferentes tipos de sistemas de recomendación para el Dataset Steam y elegir el que presente un mejor desempeño.

Carga del Dataset y Preprocesamiento de los datos

Las librerías utilizadas para el preprocesamiento y análisis exploratorio son Numpy, Pandas, Matplotlib, Neaborn, Nltk, Re y Scikit-learn.

El Dataset Steam consta de dos archivos, un Dataset Reviews que contiene la reseña que cada usuario ha hecho sobre un juego en específico y algunos datos sobre la interacción entre dicho usuario y el juego en cuestión, y el Dataset Games que tiene toda la información sobre el contenido y características de los juegos. Inicialmente el Dataset Reviews consta de 7.793.069 entradas y 12 características y el Dataset Games consta de 32.135 entradas y 16 características.

Inicialmente se realizó la depuración de ambos datasets, eliminando las características poco informativas. En el caso del Dataset Reviews, se eliminaron las columnas page_order, date, early_acces, page, compensation y found_funny, además de la columna user_id ya que si bien es necesaria para codificar los usuarios, esta tiene una gran cantidad de valores faltantes. Posterior a todo el preprocesamiento se generó una columna user_id que tomará valores consecutivos para facilitar el desarrollo de los sistemas de recomendación. Por su parte en el Dataset Games las columnas url, release_date, discount_price, reviews_url, early_access, sentiment y metascore son eliminadas por ser poco relevantes. Además las columnas app_name y publisher fueron eliminadas porque cuentan con la misma información que las columnas title y publisher.

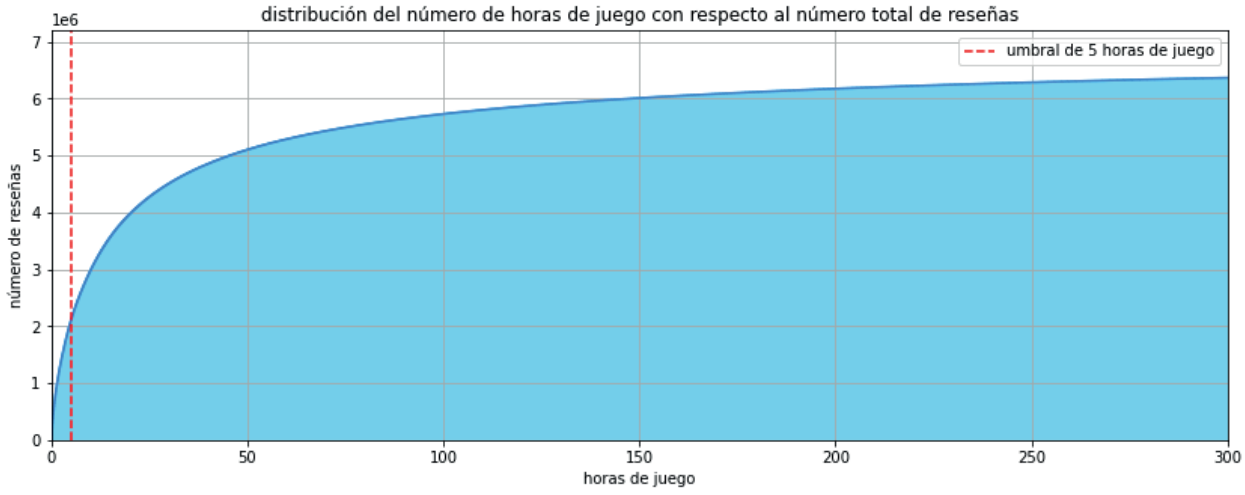
En el Dataset reviews, las entradas que no cuentan con número de horas jugadas o cuyo número de horas jugadas es cero fueron descartadas, ya que estas no son útiles para desarrollar el algoritmo de recomendación. También se eliminaron todas las entradas que contienen al usuario y al juego por duplicado, aún cuando las demás características sean diferentes. Esto, debido a que se tomaron los datos de un usuario jugando un mismo

juego en diferentes momentos. En este caso se conservó el número de horas mayor, como la variable determinante para definir la preferencia de un usuario por un juego, como se verá posteriormente.

En el Dataset Games se eliminaron las entradas que no tenían título o código de identificación del juego, además las identificaciones de juego duplicadas también fueron eliminadas. Luego de realizado este preprocesamiento inicial, el Dataset Reviews quedó con 6.859.848 entradas y 4 características, y el Dataset Games quedó con 30.083 entradas y 7 características.

Para llevar a cabo el preprocesamiento de los datos se analizó la distribución de algunas características del Dataset Reviews y a partir de los resultados obtenidos fueron tomadas decisiones para acortar dicho Dataset a uno con una distribución mucho más homogénea, de tal manera que los sistemas de recomendación diseñados se adaptaran de la mejor manera a dicho conjunto de datos.

Luego de ello se analizó la distribución de horas de juego con respecto a las reseñas totales y utilizando la distribución presentada en la siguiente gráfica, se logra determinar el valor adecuado para el umbral del número de horas de juego que definió posteriormente la preferencia de un usuario por un juego.

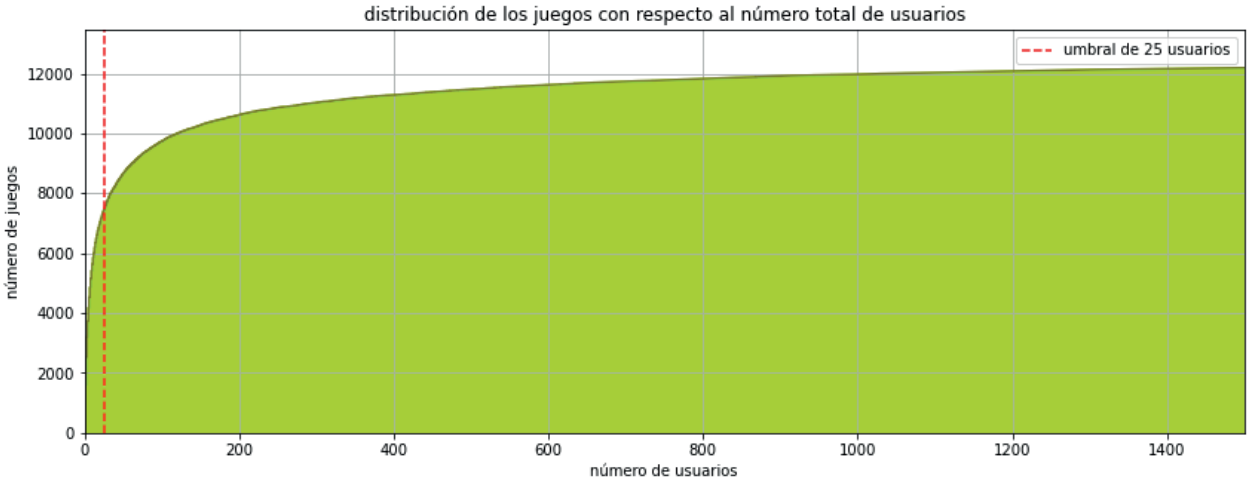


El umbral de preferencia de un usuario por un juego es de suma importancia, ya que al tratarse de un dataset de feedback implícito, no se cuenta con una calificación explícita de un usuario con respecto a un juego (sea de su gusto o no); si no con una o varias

variables que implícitamente reflejan la preferencia a través de la interacción entre el usuario y el juego. En este caso la variable utilizada fue el número de horas de juego y el umbral que definió la preferencia de un usuario por un juego fue de 5 horas. Así se puede observar que el 70% de las reseñas están ubicadas sobre el umbral propuesto, lo que deja un dataset extenso con preferencia positivas definidas.

En un sistema de recomendación implícito la preferencia es una variable binaria que toma los valores 1 (si hay preferencia) o 0 (si no la hay). Es por ello que las reseñas con menos de 5 horas de juego, al igual que todas las interacciones faltantes entre usuarios y juegos tomarán el valor de 0, por ello fueron eliminadas estas reseñas y con las reseñas restantes se construyó la preferencia positiva. El Dataset Reviews restante quedó con 4.789.783 entradas y 4 características.

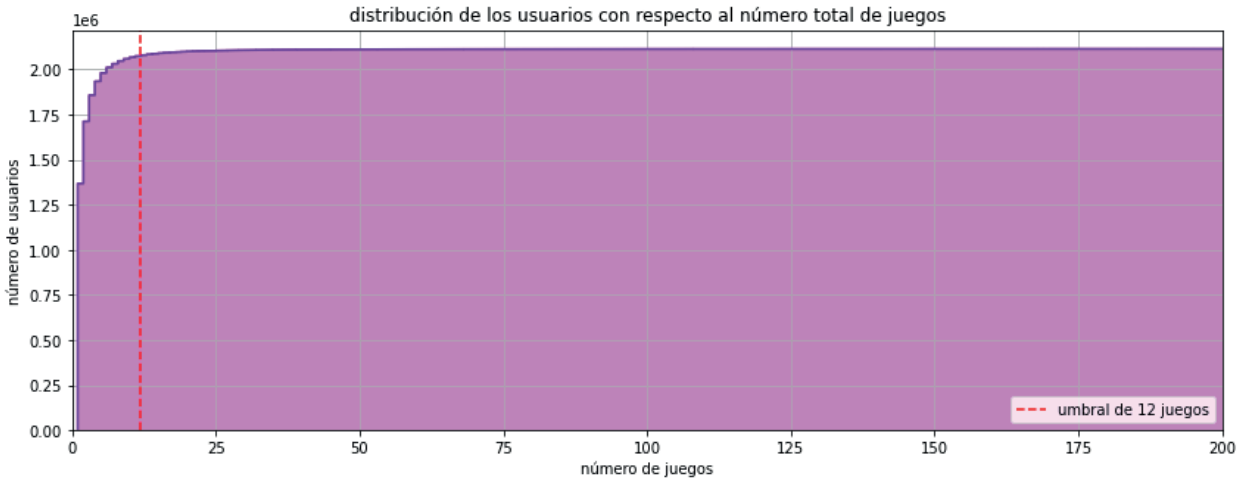
Ahora se analizará la distribución de los productos en el Dataset Reviews, que quedó con 12.822 productos diferentes, con respecto al total de usuarios. Utilizando esta distribución se determinó el valor adecuado para el umbral del número de usuarios por juego, aplicada posteriormente para eliminar los juegos menos populares y representativos.



Tomando un umbral de 25 usuarios por juego se evidenció la existencia de 7.380 juegos con menos de 25 usuarios y si bien esto corresponde al 60% de los juegos existentes, dichos juegos corresponden a 49.648 reseñas que es solo el 1% del total de reseñas con las que contamos. Es por ello que se definió el umbral del número de usuarios por juego en 25, fueron eliminadas las reseñas que correspondian a los juegos poco populares y el

Dataset Reviews restante consta de 4740135 entradas y 4 características. De igual manera debemos eliminar estos juegos del Dataset Games, quedando este con 5442 entradas y 7 características.

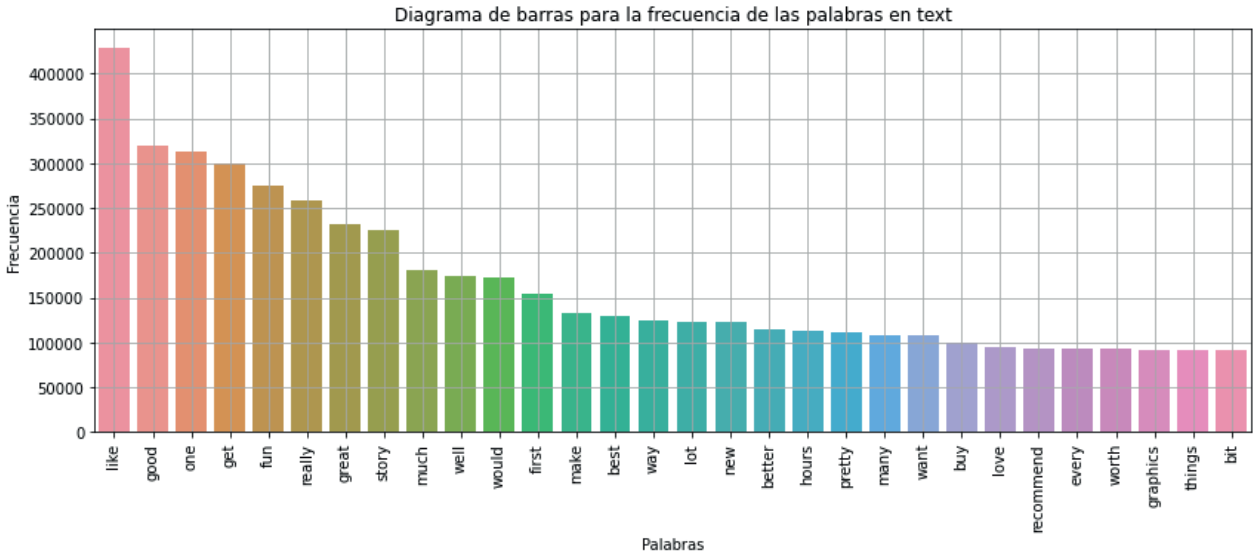
A continuación se revisará la distribución de los usuarios en el Dataset Reviews, que cuenta con 2.113.932 usuarios, con respecto al número total de productos. Al utilizar esta distribución, se determinó el valor adecuado para el umbral del número de juegos por usuario que posteriormente se aplicó para eliminar los usuarios con muy poco historial, debido a que se dificultó encontrar recomendaciones para los mismos.



Considerando el umbral propuesto de 12 productos por usuario, se evidenció que esto corresponde a 40.748 usuarios,que si bien representan solo el 2% de los usuarios, componen el 32% de las reseñas. Por lo tanto este umbral generó un dataset mucho más homogéneo y se pudieron mejorar los resultados del modelo generado en el proyecto anterior. Es por ello que se define el umbral del número de juegos por usuario en 12. Eliminando los usuarios con poco historia, para obtener un Dataset Reviews restante que quedó con 997.597 entradas y 4 características.

Ahora se hará un análisis utilizando herramientas del procesamiento de lenguaje natural de la característica text bajo la hipótesis que todas las reseñas presentadas revelan una preferencia positiva, algo que se busca corroborar con este.

A continuación presentamos los diagramas de barras para la frecuencia de las palabras y los bigramas en esta característica.



Se puede observar que entre las palabras más comunes están like, good, one, fun, great, best, better, pretty, love, worth, entre otros, que tienen un sentido positivo, lo que da validez a la hipótesis planteada.

Posterior al análisis y limpieza del Dataset, buscando siempre darle más homogeneidad, se construyeron los códigos de identificación adecuados para usuarios y productos. Tal como se había presentado en el segmento anterior, los usuarios tenían mayormente valores nulos en tal característica y por ello se eliminaron. En su lugar se generó una codificación que va del 0 al 40.747. Por su parte los productos, si bien tenían todos un código de identificación, este no era consecutivo y dificultaba la implementación de los modelos de recomendación, por ello se eliminaron y se generó una codificación que va del 0 al 5.441.

Finalmente se debió generar una columna para construir la matriz de utilidad con entradas binarias (1 o 0) según la preferencia positiva o negativa de un usuario por un juego. Debido al preprocesamiento previo todas las preferencias en el Dataset Reviews fueron positivas y por ello la columna generada contenía solo unos.

Posteriormente se construyó una matriz de contenidos para los juegos, utilizando las características del Dataset Games. Para ello se unieron las características, géneros, etiquetas y especificaciones y con su contenido se realizaron variables del tipo one hot

encoder. Para los desarrolladores se consideraron aquellos con 5 o máss juegos, con el fin de construir variables del mismo tipo. Finalmente con los títulos se aplicó procesamiento de lenguaje natural, donde se seleccionaron las 600 palabras más comunes para vectorizarlas, llevándolas a variables one hot encoder también. De esta manera se obtuvo un Dataset de contenidos con 5.442 entradas y 1.085 características.

Formulación y desarrollo de los sistemas de recomendación

Como parte del proyecto se desarrolló un algoritmo basado en contenidos (utilizando las características de los juegos proveídas en el Dataset Games), dos algoritmos de filtro colaborativo basados en memoria (uno utilizando medida de similitud coseno y otro BM25) y un algoritmo de filtro colaborativo basado en modelo que hace uso de la factorización matricial. Además se comparó el rendimiento de estos enfoques con un caso de referencia, que corresponde a la recomendación según el Top de popularidad de los ítems.

La precisión media promedio a K items (MAP@K) fue la métrica utilizada para evaluar el desempeño del modelo. A diferencia de otras métricas que no distinguen la posición de las recomendaciones relevantes, esta métrica permitió darle más peso a los errores que se presentaban más arriba en la lista de recomendaciones. Además de ser una métrica muy útil para los datasets binarios, cómo en este nuestro caso, debido a la función que introduce de relevancia.

Los modelos se evaluaron utilizando la métrica elegida y desarrollando un procedimiento de validación cruzada (efectuando una iteración por cada usuario en el dataset de prueba). Los algoritmos que solo tenian un hiperparámetro podían ser fácilmente optimizados por una búsqueda exhaustiva de valores, mientras que para aquellos que poseían varios hipeparámetros se desarrollaró un procedimiento de optimización bayesianautilizando procesos gausianos. El propósito de este procedimiento fue aproximar la función de costo como la superposición de funciones gausianas y a partir de este enfoque evaluar la métrica obtenida con cada iteración para buscar la configuración de hiperparámetros que minimizara la función de costo.

Para el desarrollo de los modelos de filtro colaborativo se hizo uso de la librería Implicit que tiene implementaciones para estos, y para la optimización gausiana se hizo uso de la librería Scikit-optimize. Para el desarrollo del quinto modelo basado en contenidos, se desarrolló una clase con sus funciones y atributos correspondientes para entrenar, recomendar y hacer la evaluación para la métrica elegida, debido a que no se encontraron librerías que implementaran dicho modelo.

01. Caso Benchmark por popularidad

El primer modelo desarrollado fue el caso benchmark o línea base, dado por la recomendación de los juegos según su top de popularidad.

02. KNN con similitud coseno

El segundo fue un modelo basado en memoria, que hace uso de la similitud coseno. Este posee un solo hiperparámetro de tipo entero (K), por ello se desarrolló la optimización del modelo por medio de una iteración definida a lo largo de dicho hiperparámetro, obteniendo un valor de K igual a 6.

03. KNN con similitud BM25

El tercer modelo fue también basado en memoria, con 3 hiperparámetros (K entero, K1 y B reales), por ello se desarrolló un procedimiento de optimización bayesiana para obtener los mejores valores de estos. Debido a que el modelo de optimización requiere el rango y la naturaleza del hiperparámetro, se construyó una función de evaluación que tomó los valores para estos hiperparámetros y obtuvo un valor para la métrica. Además se generaron condiciones de parada para detener la optimización, donde se determinó que si se obtenia una diferencia en las métricas menor a 0.0015 o si se superaba un tiempo de ejecución de 20 minutos, lo valores de K serían igual a 200, K1 igual a 0.1156 y B igual a 1.3907.

04. Mínimos cuadrados alternantes

El modelo de mínimos cuadrados alternantes se le asignaron 4 hiperparámetros (alpha, factor latente e iteraciones son enteros, regularización es real). De nuevo se desarrolló un procedimiento de optimización bayesiana. Las condiciones de parada en este caso fueron por diferencia entre los resultados de las métricas (menor a 0.0015) o si superase un tiempo de ejecución de 5 horas (se consideraba un tiempo mucho mayor ya que cada iteración podía tomar mucho más tiempo). Se obtuvieron valores de alpha igual a 31, factor latente igual a 347, regularización igual a 251.3753 e iteraciones igual a 150.

05. Basado en contenidos

Para el desarrollo de este modelo se construyó una clase llamada BasedContent para incorporar los métodos fit() para entrenar el modelo y recomend() para generar las recomendaciones para un usuario específico, de manera similar a como funcionan los métodos contenidos en las librerías conocidas de sistemas de recomendación como Implicit, Surprise y LightFM.

A continuación se desarrolló la optimización del hiperparámetro K del algoritmo, para lo cual se realizó una iteración definida a lo largo de dicho hiperparámetro, obteniendo un valor de 50 para K.

Comparativo del desempeño de los modelos

A continuación presentamos una tabla con los resultados de la métrica de evaluación para cada uno de los modelos planteados. En este vemos que el algoritmo de filtro colaborativo basado en modelo , de mínimos cuadrados alternantes, es el que presenta mejores resultados.

Modelo	Métrica
ALS	0.065455
KNN con similitud BM25	0.057390
Basado en contenidos	0.055457
KNN con similitud coseno	0.050416
Caso benchmark	0.028603

Desarrollo de las predicciones para el modelo ALS

Se elijió un usuario al azar, es este caso el 33.706, se observó su historial de juegos y las recomendaciones las recomendaciones sugeridas por el modelo.

```
Usuario elegido:  andyfinn2012

Juegos del usuario :

['Batman™: Arkham Origins Blackgate - Deluxe Edition',
 'DC Universe™ Online',
 'The Ship: Murder Party',
 'Kingdom Rush',
 'Reus',
 'Rocketbirds: Hardboiled Chicken',
 'LEGO The Lord of the Rings',
 'STAR WARS™ Knights of the Old Republic™ II - The Sith Lords™',
 'STAR WARS™ Empire at War - Gold Pack',
 "Zombie Tycoon 2: Brainhov's Revenge",
 'PAYDAY™ The Heist',
 'STAR WARS™ - Knights of the Old Republic™',
 'HELLDIVERS™',
 'Risk of Rain',
 'LEGO® Marvel™ Super Heroes',
 'Battlefield: Bad Company™ 2',
 'STAR WARS™ Republic Commando™']
```

```
Recomendaciones por similitud:

Batman™: Arkham Origins
Star Wars: Battlefront 2 (Classic, 2005)
Batman™: Arkham Knight
LEGO® Batman™3: Beyond Gotham
STAR WARS™ Jedi Knight - Jedi Academy™
LEGO® Star Wars™ - The Complete Saga
State of Decay
Gunpoint
Half-Life 2
LEGO® The Hobbit™
FTL: Faster Than Light
Batman: Arkham Asylum Game of the Year Edition
Sins of a Solar Empire®: Rebellion
Dungeon of the Endless™
Awesomenauts - the 2D moba
Left 4 Dead
Magicka
STAR WARS™ - The Force Unleashed™ Ultimate Sith Edition
Metro 2033 Redux
ORION: Prelude
```

Se puede ver como las principales recomendaciones para este jugador comparten características claves con sus productos jugados, como la temática de superhéroes y las historias de ciencia ficción.