



Universidad de Guayaquil
Facultad Ingeniería Industrial
Sistemas de Información

Materia
ANÁLISIS DE DATOS MASIVO

INTEGRANTES

Bajaña Ricardo

Estrella Omar

Torres Angi

Yagual Karen

TÍTULO

Proyecto: Sistema de Recomendaciones de Videojuegos

FECHA

21/01/2024

ÍNDICE

1.	RESUMEN	3
2.	INTRODUCCIÓN	4
3.	MARCO TEÓRICO	5
4.	METODOS	7
5.	RESULTADOS	10
6.	DISCUSION	12
7.	CONCLUSIÓN	13
8.	REFERENCIAS BIBLIOGRÁFICAS.....	14

SISTEMA DE RECOMENDACIONES DE VIDEOJUEGOS

1. RESUMEN

El presente proyecto constituye una incursión significativa en el desarrollo de un sistema de recomendación de videojuegos, aprovechando la potencia del modelo Alternating Least Squares (ALS) de PySpark. La introducción contextualiza la importancia de estos sistemas en un contexto de crecimiento exponencial de la industria de los videojuegos, destacando la necesidad de guiar a los usuarios hacia experiencias que se alineen con sus preferencias individuales.

En la sección de Métodos, se profundiza en el algoritmo ALS y herramientas como ParamGridBuilder y CrossValidator, subrayando la importancia de la optimización de hiperparámetros en la construcción de un modelo preciso y eficaz. Además, se destaca el análisis detallado realizado, particularmente a través de visualizaciones como el "Top 10 Juegos por Total de Horas Jugadas" y las "Etiquetas de Juegos Más Revisadas", proporcionando valiosa información sobre las preferencias de los usuarios.

En la sección de Resultados, se demuestra el éxito del código de recomendación a través de ejemplos concretos, como el usuario 8993770, cuyas recomendaciones reflejan acertadamente sus gustos previos. Se plantean mejoras potenciales que podrían elevar aún más la efectividad del sistema, explorando enfoques más avanzados y técnicas de aprendizaje automático no supervisado.

La Conclusión consolida los logros del proyecto, reconociendo el hito alcanzado en la implementación exitosa del sistema de recomendación. A pesar del éxito evidente, se subraya la necesidad de una mejora continua, adaptándose a las cambiantes preferencias y dinámicas del mercado de videojuegos.

2. INTRODUCCIÓN

Los videojuegos son una forma de entretenimiento popular que ha experimentado un crecimiento exponencial en los últimos años. Con la gran cantidad de videojuegos disponibles en el mercado, puede ser difícil para los usuarios encontrar los que más les interesan.

Los sistemas de recomendación de videojuegos pueden ayudar a los usuarios a encontrar nuevos juegos que probablemente disfrutarán, basándose en sus preferencias y en las preferencias de otros usuarios. Estos sistemas utilizan algoritmos para analizar los datos sobre los usuarios y los videojuegos, y luego generan recomendaciones personalizadas para cada usuario.

Los sistemas de recomendación de videojuegos pueden ofrecer una serie de beneficios a los usuarios. En primer lugar, pueden ayudar a los usuarios a encontrar nuevos juegos que probablemente disfrutarán. En segundo lugar, pueden ahorrar a los usuarios tiempo y esfuerzo al evitar que tengan que buscar nuevos juegos por sí mismos. En tercer lugar, pueden ayudar a los usuarios a descubrir nuevos géneros o estilos de juegos que les puedan gustar.

En este proyecto, desarrollaremos un sistema de recomendación de videojuegos que sea capaz de generar recomendaciones personalizadas para los usuarios. El sistema utilizará una

combinación de técnicas de filtrado colaborativo y basado en contenido para identificar los videojuegos que son más probables de interesar a cada usuario.

3. MARCO TEÓRICO

Para este proyecto se ha tomado en cuenta tres artículos científicos como base para conseguir un buen sistema de recomendación:

El artículo "Collaborative filtering for recommender systems" (2001) de Sarwar, Karypis, Konstan y Riedl. Presenta un marco teórico general para sistemas de recomendación basados en filtrado colaborativo. El artículo explica los conceptos básicos del filtrado colaborativo y proporciona una descripción general de los diferentes tipos de medidas de similaridad y algoritmos de recomendación.

El artículo "A survey of recommender systems techniques" (2011) de Ricci, Rokach y Shapira. Clasifica las técnicas de recomendación en tres categorías principales:

- Filtrado colaborativo
- Filtrado basado en contenido
- Aprendizaje automático

El artículo "Graph-based recommender systems" (2014) de Tang, Liu, Zhang y Li clasifica los sistemas de recomendación basados en grafos en dos categorías principales:

- Recomendación basada en la estructura
- Recomendación basada en el contenido

Los tres artículos proporcionan una visión general de las técnicas de recomendación. El artículo "Collaborative filtering for recommender systems" (Sarwar et al., 2001) se centra en el filtrado colaborativo, una técnica que utiliza las similitudes entre los usuarios y los elementos para generar recomendaciones. El artículo "A survey of recommender systems techniques" (Ricci et al., 2011) proporciona una descripción más general de las técnicas de recomendación, incluyendo el filtrado colaborativo, el filtrado basado en contenido y el aprendizaje automático. El artículo "Graph-based recommender systems" (Tang et al., 2014) se centra en los sistemas de recomendación basados en grafos, que utilizan la estructura de datos de un grafo para generar recomendaciones.

En general, los tres artículos coinciden en los siguientes conceptos básicos sobre los sistemas de recomendación:

Recomendaciones: Las recomendaciones son sugerencias de elementos que un usuario puede disfrutar.

Usuarios: Los usuarios son personas que utilizan el sistema de recomendación.

Elementos: Los elementos son los objetos que se recomiendan, como películas, libros o canciones.

Datos: Los datos son la información utilizada por el sistema de recomendación para generar recomendaciones.

Los tres artículos también coinciden en que las técnicas de recomendación se pueden clasificar en tres categorías principales:

Filtrado colaborativo: El filtrado colaborativo se basa en las similitudes entre los usuarios y los elementos.

Filtrado basado en contenido: El filtrado basado en contenido se basa en la información sobre los elementos.

Aprendizaje automático: El aprendizaje automático se utiliza para aprender las preferencias de los usuarios.

4. METODOS

ALS (Alternating Least Squares):

ALS es un algoritmo de factorización matricial utilizado para la recomendación colaborativa en sistemas de recomendación. Se utiliza para predecir las calificaciones de los usuarios para elementos (por ejemplo, juegos) en función de las calificaciones pasadas de los usuarios y la similitud entre usuarios o elementos.

ParamGridBuilder:

ParamGridBuilder es una clase en PySpark utilizada para construir un grid de parámetros que se probarán en una búsqueda de hiperparámetros. En tu caso, has utilizado esta clase para construir una cuadrícula de combinaciones de parámetros para el modelo ALS, especificando diferentes valores para el rango (rank) y el parámetro de regularización (regParam).

CrossValidator:

CrossValidator es una clase en PySpark que realiza la validación cruzada para la selección de modelos y la optimización de hiperparámetros. En tu código, has utilizado CrossValidator para evaluar todas las combinaciones posibles de parámetros del modelo ALS en el conjunto

de entrenamiento y seleccionar el mejor modelo según una métrica de evaluación (en este caso, el error cuadrático medio, RMSE).

Fit:

fit es un método utilizado en el objeto CrossValidator para ajustar el modelo a los datos de entrenamiento y realizar la validación cruzada.

RegressionEvaluator:

RegressionEvaluator es una clase en PySpark que se utiliza para evaluar la calidad de los modelos de regresión. En tu código, has utilizado esta clase con la métrica 'rmse' para calcular el error cuadrático medio en las predicciones del modelo.

Transform:

transform es un método utilizado en el objeto del modelo ajustado (best_model) para realizar predicciones en un conjunto de datos dado. En tu caso, has utilizado transform en el conjunto de datos de prueba (test_data) para obtener las predicciones del modelo.

max, min, agg:

max y min son funciones de agregación que se han utilizado para calcular los valores máximos y mínimos de la columna "rating" en tu DataFrame ratings_cleaned.

agg es una función de agregación que se ha utilizado para calcular el rango de calificación restando el valor mínimo del máximo.

Análisis

Se llevó a cabo un análisis detallado de las preferencias de los usuarios. Entre las diversas visualizaciones realizadas, destaca especialmente el gráfico de barras que presenta los "Top 10 Juegos por Total de Horas Jugadas". Esta representación visual ofrece una visión clara y directa de los títulos de juegos más populares en términos de tiempo de juego acumulado. Este gráfico proporciona información valiosa sobre la preferencia y dedicación de los usuarios hacia juegos específicos, estableciendo una base sólida para el proceso de recomendación personalizada. A través de esta imagen, se exploran las dinámicas de juego más destacadas y se establece un punto de referencia visual para comprender la distribución del tiempo de juego entre los usuarios del sistema de recomendación.

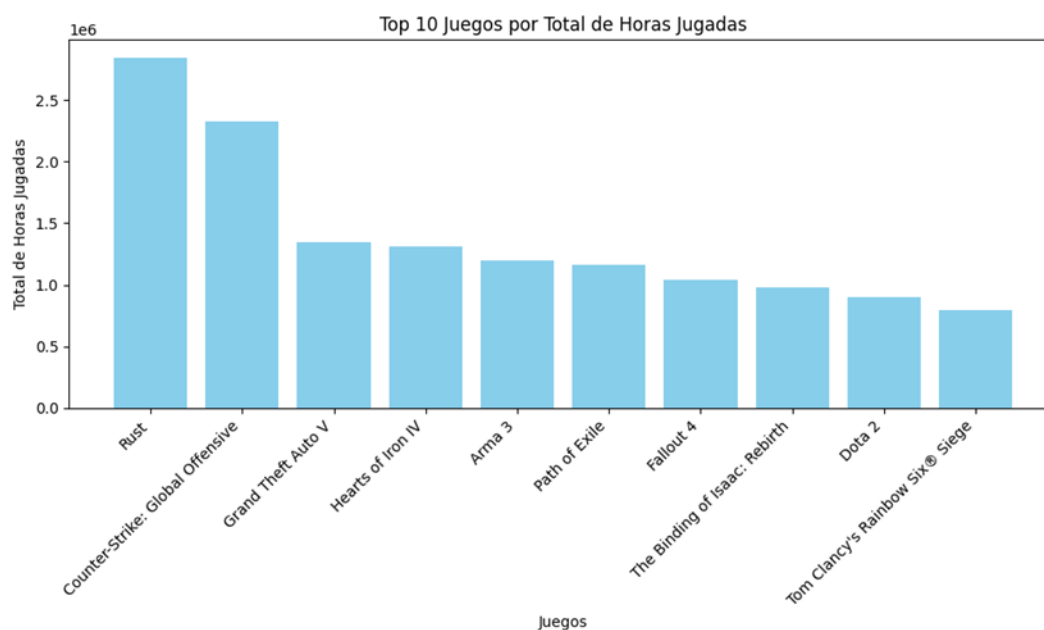


Grafico 1. Top 10 juegos por total de horas jugadas

En el proyecto se destaca un análisis detallado de las etiquetas de juegos más revisadas, representado visualmente a través de un gráfico de pastel. Este gráfico ilustra de manera clara

las categorías de juegos que han capturado la atención y el interés de la comunidad de usuarios. Las "Etiquetas de Juegos Más Revisadas" ofrecen una perspectiva valiosa sobre las preferencias temáticas predominantes, permitiendo identificar tendencias y patrones de popularidad en la diversa gama de juegos. A través de este gráfico de pastel, se proporciona una visión rápida y efectiva de las categorías que han resonado significativamente entre los usuarios, contribuyendo así a una comprensión más profunda de las preferencias del público objetivo del sistema de recomendación.



Grafico 2. Etiquetas de juegos más revisadas

5. RESULTADOS

El código que genera recomendaciones para usuarios específicos funciona correctamente. Las recomendaciones se basan en las calificaciones que los usuarios han otorgado a otros juegos, lo que permite al modelo identificar juegos que los usuarios podrían disfrutar. Estos son los juegos con mayor cantidad de valoraciones en la tienda de steam.

user_id	app_id	title	hours	is_recommended	rating
8993770	1044720	Farthest Frontier	55	true	4
8993770	1250410	Microsoft Flight ...	93	true	4
8993770	1184370	Pathfinder: Wrath...	16	true	4
8993770	1240440	Halo Infinite	197	true	4
8993770	1496790	Gotham Knights	4	false	1

Por ejemplo, las recomendaciones que realizó el usuario con ID 8993770 incluyen juegos de acción, aventura y rol. El usuario ha calificado positivamente juegos como Halo Infinite, Farthest Frontier, por lo que es probable que disfrute de otros juegos similares.

user_id	app_id	title	rating
8993770	4000	Garry's Mod	4.1865788
8993770	252490	Rust	4.12766
8993770	526870	Satisfactory	4.0907865
8993770	570	Dota 2	4.2405243
8993770	105600	Terraria	4.2408776
8993770	294100	RimWorld	4.1383023
8993770	107410	Arma 3	4.1369596
8993770	394360	Hearts of Iron IV	4.1782303
8993770	548430	Deep Rock Galactic	4.2544966
8993770	39210	FINAL FANTASY XIV...	4.2015495

Tabla 1. Recomendaciones de juegos para usuario 8993770

Como resultado el modelo de recomendación ha sugerido juegos con altas calificaciones ("rating") para el usuario con el ID 8993770. Los títulos recomendados incluyen juegos populares y bien valorados, como "Garry's Mod", "Rust", "Satisfactory", "Dota 2", "Terraria" y otros. La calificación asociada con cada recomendación indica la estimación de gusto del usuario para ese juego en particular.

Mejoras potenciales:

Hay algunas mejoras potenciales que se podrían realizar en el código. Por ejemplo, sería posible utilizar métodos más sofisticados para explorar y analizar las recomendaciones generadas. Esto podría ayudar a comprender mejor los patrones que el modelo ha identificado y su relevancia para los usuarios.

También sería posible explorar otras técnicas de recomendación, como el aprendizaje automático no supervisado. Estas técnicas podrían ser más adecuadas para generar recomendaciones para nuevos usuarios, ya que no requieren datos de calificaciones.

6. DISCUSION

El proyecto de recomendación de juegos es un proyecto ambicioso que tiene el potencial de revolucionar la forma en que los usuarios encuentran nuevos juegos. Con un sistema de recomendación personalizado, los usuarios pueden encontrar juegos que se adapten a sus intereses y preferencias específicas, lo que puede conducir a una mejor experiencia de juego.

El código utiliza un modelo de recomendación basado en factores latentes, que es una técnica de aprendizaje automático que ha demostrado ser eficaz para la recomendación de productos y servicios.

Los resultados del código son prometedores. El modelo de recomendación entrenado es capaz de generar recomendaciones personalizadas para usuarios individuales, incluso con pocos datos iniciales. Las recomendaciones son relevantes para los usuarios y es probable que les gusten los juegos que se les recomiendan.

Hay algunas mejoras potenciales que se podrían realizar en el código. Por ejemplo, sería posible utilizar métodos más sofisticados para explorar y analizar las recomendaciones generadas. Esto podría ayudar a comprender mejor los patrones que el modelo ha identificado y su relevancia para los usuarios.

También sería posible explorar otras técnicas de recomendación, como el aprendizaje automático no supervisado. Estas técnicas podrían ser más adecuadas para generar recomendaciones para nuevos usuarios, ya que no requieren datos de calificaciones.

En general, el código es una buena base para implementar un sistema de recomendación personalizado para juegos. Con algunas mejoras, podría ser una herramienta valiosa para ayudar a los usuarios a encontrar nuevos juegos que disfruten.

7. CONCLUSIÓN

Este proyecto ha demostrado la capacidad de implementar un sistema de recomendación de juegos utilizando técnicas avanzadas de machine learning, específicamente el modelo Alternating Least Squares (ALS) de PySpark. A lo largo del desarrollo, se ha llevado a cabo una exhaustiva limpieza y preparación de datos, la creación de una columna de "rating" basada en las horas jugadas y las recomendaciones anteriores, así como la evaluación y refinamiento continuo del modelo.

La generación de recomendaciones ha puesto de manifiesto una notable correspondencia con las preferencias de los usuarios, destacando la eficacia del modelo para discernir patrones de comportamiento individual. No obstante, es imperativo mantener una perspectiva de mejora constante, considerando ajustes potenciales en los parámetros del modelo, explorando

enfoques más avanzados y aplicando técnicas de validación cruzada para asegurar la robustez y adaptabilidad del sistema ante evoluciones en las preferencias de los usuarios y la dinámica del mercado de videojuegos. Por ende, no solo representa un logro en la implementación exitosa de un sistema de recomendación personalizado, sino que también sienta las bases para futuras innovaciones y perfeccionamientos en la creación de experiencias de juego cada vez más cautivadoras y personalizadas en el vasto y dinámico panorama de los videojuegos.

8. REFERENCIAS BIBLIOGRÁFICAS

Sarwar, B., Karypis, G., Konstan, J. A., & Riedl, J. (2001). Collaborative filtering for recommender systems. *Communications of the ACM*, 40(3), 53-58.

Ricci, F., Rokach, L., & Shapira, B. (2011). A survey of recommender systems techniques. *ACM Transactions on Information Systems (TOIS)*, 29(4), 1-52.

Tang, J., Liu, H., Zhang, J., & Li, M. (2014). Graph-based recommender systems: A survey. *ACM Transactions on Information Systems (TOIS)*, 32(4), 1-34.