



Introducción

En los últimos años, la industria de los juegos de mesa ha visto un fuerte auge con el lanzamiento de cientos de títulos nuevos, y se espera que este crecimiento continúe^[4]. Esto genera una dificultad para los usuarios, ya que es cada vez más difícil encontrar aquellos juegos que se ajusten mejor a sus preferencias.

Como solución a esto, se propone la creación de un sistema recomendador de juegos de mesa, el cual recomendará títulos relevantes para un usuario a partir de sus preferencias previas.

Dataset

Se utiliza la base de datos Board Games Database^[2] de Kaggle

- 22 000 Juegos (y sus atributos)
- 411 000 Usuarios
- 19 000 000 calificaciones de juegos por usuarios

Se usa un *sample* constante del 1% de las calificaciones de juegos, lo que implica también una reducción en la cantidad de usuarios y juegos.

De ésta base de datos, se usan las siguientes tablas:

- **user_ratings**: Muestra todos los ratings que cada usuario le ha dado a juegos.
- **games**: Presenta características de cada juego, como descripción, lanzamiento, entre otras.
- **themes**: Indica el género de cada juego.

Algoritmos

LightFM, que aplica factorización matricial de alguna matriz de ratings, creando vectores de factores latentes entre usuarios y juegos. También se usa TF-IDF, que genera una matriz a partir de la frecuencia y novedad de palabras en las descripciones. Por último, es usado un BERT, modelo de lenguaje que vectoriza las descripciones usando redes neuronales, incluyendo frecuencia, contexto y significado.

Metodología

Se hace uso de las descripciones de los juegos en la tabla *games*. Estos son *strings* de variadas longitudes que describen las características de cada juego.

A partir de esta, se obtienen dos matrices de características. La primera es la matriz del método TF-IDF, limitada a 5000 características por capacidad computacional. La segunda es la matriz de BERT, que utiliza una red neuronal para generar factores latentes para cada las descripciones.

Con lo anterior, se implementan tres modelos de factorización matricial con la librería *lightFM*^[3] para Python. Estos modelos se entrenan con la matriz de interacciones usuario-juego que se obtiene en base a la tabla *user_ratings* y cada modelo usa distinta metadata.

Para cada modelo se usa como metadata de los ítems (*item features*) una de las siguientes:

- 1) Género de los juegos, de la tabla *themes*
- 2) Características TF-IDF, reducidas con truncated SVD a 100 componentes.
- 3) Características BERT. La matriz se reduce usando PCA a 50 componentes.

Una vez entrenados, los modelos de LightFM toman como input las interacciones de un usuario para generar recomendaciones para este. Se obtienen las métricas MAP@k, nDCG@k (con k = 5, 10 y 15), diversidad y novedad para cada uno de los modelos.

Resultados

Modelo	MAP	nDCG	Novedad	Diversidad
Most popular	0,082	0,206	2,074	0,480
Best Ranked	0,000	0,000	12,964	0,577
LightFM	0,111	0,207	3,729	0,729
LightFM (TF-IDF)	0,082	0,122	2,501	1,000
LightFM (BERT)	0,125	0,179	2,550	1,000

Tabla 1: Resultados para k = 5

Modelo	MAP	nDCG	Novedad	Diversidad
Most popular	0,071	0,208	2,305	0,560
Best Ranked	0,000	0,000	12,964	0,577
LightFM	0,107	0,178	3,020	0,682
LightFM (TF-IDF)	0,073	0,134	2,744	0,705
LightFM (BERT)	0,051	0,102	2,742	0,697

Tabla 2: Resultados para k = 10

Modelo	MAP	nDCG	Novedad	Diversidad
Most popular	0,067	0,216	2,433	0,572
Best Ranked	0,000	0,000	12,964	0,577
LightFM	0,145	0,195	3,329	1,000
LightFM (TF-IDF)	0,052	0,108	2,870	0,692
LightFM (BERT)	0,042	0,099	2,862	1,000

Tabla 3: Resultados para k = 15

Observaciones:

El valor *k* indica la cantidad de predicciones hechas por cada modelo, a las cuales se le aplican las métricas indicadas. Las predicciones se obtienen para una *sample* de 100 usuarios del *test set* (por temas de rendimiento).

Para k = 5:

- LightFM con *themes*, con BERT y Most Popular obtienen los mejores resultados.

Para k = 10:

- LightFM con TF-IDF mejor que con BERT, pero ambas peor que con *themes*.

Para k = 15

- LightFM con *themes* entrega la mejor combinación de métricas

Además, para todos los casos:

- Best Ranked tiene el peor rendimiento en MAP y nDCG, pero ofrece la mayor novedad

Conclusiones

De los resultados se pueden hacer diversas conclusiones:

LightFM es un método que se adapta al tipo de solución necesario para esta tarea, ya que es basado en contenido. Esto le permite incorporar *features* adicionales a los *ratings*, como la descripción de los juegos, presente en el dataset utilizado. Gracias a esto, reduce el problema del *cold start*.

Sin embargo, las métricas obtenidas no presentan un buen rendimiento. Aquellas basadas en un vector de verdad (*ground truth*) no lograron un puntaje mejor a 0,25 (siendo 1 el máximo y 0 el mínimo). El equipo estima que la principal causa es el *sampling* implementado sobre el dataset, el que solo considera el 1% de los ratings hechos por los usuarios, causando que un gran número de ítems (juegos), que potencialmente son buena recomendación para un usuario, queden fuera del *ground truth*.

Existirían diversas oportunidades de mejora para aumentar el rendimiento. La primera es usar más cantidad de datos para los modelos de LightFM, con lo que aumentaría drásticamente el rendimiento y la cantidad de información para entrenar y probar, y con ello, mejorarían las métricas de MAP y nDCG. La segunda oportunidad estaría en preprocesar los textos de descripción de los juegos a la hora de obtener las características TF-IDF y BERT, ya que reduciría el ruido, la dimensionalidad y la ambigüedad de los datos. Con ello, los métodos podrían luego enfocarse en las palabras y texto relevante.

Referencias

- [1] Agrawal, A. (2019, 18 febrero). Solving business usecases by recommender system using lightFM. *Medium*.
- [2] Board Game Database from BoardGameGeek. (2022, 17 enero). *Kaggle*.
- [3] Maciej Kula (2015). Metadata Embeddings for User and Item Cold-start Recommendations.
- [4] Fortune Business Insight (2025, 16 junio). Tamaño de mercado de juegos de mesa [...]