
Judge a Book by its Cover

Recomendación Multimodal de Libros

Nicolás Herrera Vicente Steidle Lucas Vidal

Abstract

En la era digital, el volumen abrumador de contenido disponible ha transformado la forma en que los usuarios descubren y consumen información. Este fenómeno es particularmente evidente en el ámbito de los libros, donde plataformas como Goodreads ofrecen catálogos masivos que pueden dificultar, en lugar de facilitar, la experiencia de lectura. Para abordar este desafío, los sistemas de recomendación se han convertido en herramientas esenciales para personalizar y optimizar el descubrimiento de contenido.

Este trabajo presenta un estudio comparativo de modelos de recomendación de libros multimodales, integrando enfoques tradicionales y modernos, incluyendo filtrado colaborativo, métodos basados en contenido y modelos de aprendizaje profundo. En particular, se explora cómo la información textual y visual, como las descripciones y portadas de los libros, puede aprovecharse de manera conjunta para mejorar la calidad de las recomendaciones. Se implementan y evalúan diversos modelos, incluyendo baselines (ALS, BPR, IBCF), arquitecturas de filtrado colaborativo neuronal y métodos basados en contenido utilizando embeddings de BERT, ResNet y VGG16.

Nuestros resultados muestran que la combinación de contenido proveniente de múltiples modalidades permite generar recomendaciones más precisas y personalizadas en comparación con enfoques unimodales. El estudio confirma que los elementos visuales contienen características semánticamente relevantes que complementan las descripciones textuales. Esto sugiere que la modelación multimodal puede proporcionar una comprensión más integral de las preferencias de los usuarios en dominios donde la apariencia y el contenido están estrechamente relacionados.

1. Introducción

En la actualidad, los sistemas de recomendación juegan un rol fundamental en la experiencia de usuario dentro de las plataformas digitales, ayudando a los usuarios a navegar catálogos extensos y a descubrir contenido relevante de manera eficiente. Esta problemática de sobrecarga de información es especialmente notoria en el dominio de los libros, donde plataformas como Goodreads ofrecen decenas de miles de títulos disponibles, dificultando la tarea de encontrar nuevas lecturas que se alineen con los intereses particulares de cada lector.

El diseño de sistemas de recomendación efectivos requiere comprender tanto las preferencias explícitas de los usuarios (por ejemplo, interacciones pasadas o valoraciones), como también las características intrínsecas del contenido disponible. Si bien los enfoques clásicos de filtrado colaborativo han demostrado ser efectivos al capturar patrones de comportamiento colectivo, dichos modelos tienden a ignorar información valiosa contenida en el contenido textual y visual de los ítems. **Esta limitación es especialmente relevante en dominios como la literatura, donde las descripciones y portadas de los libros influyen significativamente en la decisión de lectura.**

En este contexto, la presente investigación realiza un estudio comparativo de modelos de recomendación multimodal de libros, integrando señales textuales y visuales mediante técnicas de aprendizaje profundo y métodos basados en contenido. En particular, se exploran distintos esquemas de representación para las descripciones (BERT, BERT Large) y portadas (VGG16, ResNet50), y se propone una arquitectura de recomendación basada en perfiles de usuario construidos a partir de embeddings promedio. Además, se contrastan estos enfoques con métodos colaborativos clásicos (ALS, BPR, IBCF), con modelos multimodales (NCF, VBPR), evaluando su rendimiento sobre datos reales obtenidos de Goodreads.

Las contribuciones principales de este trabajo son las siguientes:

- Se analiza un conjunto representativo de modelos de recomendación que incorporan tanto información textual como visual de los libros.

- Se comparan múltiples estrategias de representación de ítems y perfiles de usuario, evaluando su impacto en la calidad de las recomendaciones.
- Se presenta un análisis experimental exhaustivo sobre un conjunto de datos real, evaluando los modelos bajo métricas estándar de recomendación como *Precision*, *Recall*, *NDCG*, *MAP*, *Intra-List Diversity* y *Novelty*.

Este estudio demuestra que el uso conjunto de información multimodal permite capturar mejor las preferencias latentes de los usuarios, lo que se traduce en recomendaciones más precisas y personalizadas. Además, abre el campo para futuras investigaciones en sistemas de recomendación que integren múltiples fuentes de información de manera eficiente y escalable.

2. Trabajo Relacionado

El campo de los sistemas de recomendación ha sido ampliamente explorado, con enfoques que van desde métodos heurísticos simples hasta modelos basados en aprendizaje profundo. En esta sección se revisan las líneas principales de investigación que sirven como base para el presente trabajo.

2.1. Filtrado Colaborativo

El filtrado colaborativo es una de las estrategias más comunes en sistemas de recomendación. Este enfoque se basa en la premisa de que los usuarios que han tenido comportamientos similares en el pasado, probablemente compartan preferencias en el futuro. Entre los algoritmos más conocidos se encuentran el filtrado basado en ítems (IBCF), y los modelos de factorización de matrices como Alternating Least Squares (ALS) (Zachariah et al., 2012) y Bayesian Personalized Ranking (BPR) (Rendle et al., 2012).

Si bien estos métodos logran capturar patrones latentes a partir de datos de interacciones usuario-ítem, presentan limitaciones en contextos donde el número de interacciones es escaso (problema del *cold start*). Por otro lado, estos modelos no aprovechan la valiosa información de los ítems cuando lo está disponible.

2.2. Recomendación Basada en Contenido

Los métodos basados en contenido buscan representar explícitamente las características de los ítems mediante descripciones, etiquetas o metadatos. En el dominio de los libros, el uso de información textual como sinopsis, título y autor permite generar recomendaciones personalizadas sin depender del comportamiento de otros usuarios.

Recientemente, modelos de lenguaje preentrenados como BERT han sido empleados para obtener representaciones semánticas de textos largos con gran efectividad (Devlin

et al., 2018). Estos embeddings han sido aplicados exitosamente en sistemas de recomendación textual al medir similitud entre ítems o entre perfiles de usuario e ítems (Javaji & Sarode, 2023).

2.3. Modelos Multimodales

El uso combinado de distintos tipos de información, como texto e imagen, ha demostrado ser prometedor para capturar una representación más rica de los ítems. Modelos multimodales, como CLIP y SigLIP, permiten mapear texto e imágenes en un espacio vectorial compartido, facilitando tareas como clasificación y recomendación (Radford et al., 2021).

Particularmente en el caso de libros, estudios han mostrado que las portadas pueden contener señales visuales útiles para predecir género o estilo narrativo, aunque aún existe espacio para mejora (Iwana & Uchida, 2016).

3. Dataset y Preprocesamiento

Para la experimentación de este trabajo, se utilizó un subconjunto del dataset de Goodreads, una plataforma y comunidad virtual ampliamente utilizada por lectores para calificar, reseñar y descubrir libros. El conjunto de datos empleado contiene información sobre 4287 libros y las interacciones implícitas de más de 52.000 usuarios, organizadas en un conjunto de entrenamiento y otro de prueba.

Cada fila del dataset incluye metadatos como el título, autor, año de publicación, número de valoraciones, descripción textual y la URL de la imagen de portada. Esta información se utilizó para construir representaciones semánticas mediante modelos preentrenados como BERT (para texto) y ResNet/VGG (para imágenes). Las portadas fueron descargadas y normalizadas en tamaño y formato para su procesamiento por redes convolucionales. Por otro lado, las interacciones usuario-libro están representadas como datos implícitos, es decir, únicamente indican si el usuario ha interactuado con un libro sin contener una calificación explícita.

Dado que algunos de los modelos utilizados generan embeddings de alta dimensionalidad (por ejemplo, BERT produce vectores de 768 o 1024 dimensiones, y ResNet hasta 2048), se aplicó *Principal Component Analysis* (PCA) como técnica de reducción de dimensionalidad (Shlens, 2014). Esta reducción fue utilizada en los modelos Factorization Machines y Neural Collaborative Filtering para disminuir significativamente el tiempo de entrenamiento sin afectar de forma notable el rendimiento del sistema, incluso mejorándolo en algunas ocasiones.

4. Modelos

4.1. Modelos Baseline

Se implementaron los modelos baseline de ALS, BPR, e IBCF. Todos estos modelos utilizan como datos solamente las interacciones entre usuarios y libros, sin considerar ningún tipo de información adicional. Además, se implementaron recomendaciones aleatorias, y recomendaciones basadas en los libros más populares.

4.2. Modelos Multimodales

Para incorporar información adicional de los libros, se implementaron modelos multimodales. Cada uno de ellos se entrenó utilizando distintos conjuntos de features: solo con interacciones, con embeddings adicionales generados por BERT, o features adicionales de las portadas generadas por VGG16 o ResNet50.

4.2.1. FACTORIZATION MACHINES (FM)

Se implementó el modelo FM, con features de ítems dados. Para esto, y considerando la complejidad de cómputo del algoritmo, se tomaron los embeddings generados por BERT de cada libro, y se aplicó PCA para reducir su dimensionalidad.

4.2.2. NEURAL COLLABORATIVE FILTERING (NCF)

Para NCF se implementaron dos variantes, detalladas a continuación.

1. Vanilla NCF: se aprende de interacciones implícitas únicamente, entrenando dos ramas de embeddings—una GMF (producto elemento a elemento) y una MLP (concatenación y capas densas)—cuyas salidas se fusionan en una capa final con sigmoide. Se usa negative sampling y se optimiza binary cross-entropy.
2. NCF con embeddings adicionales: sobre la arquitectura vanilla se incorpora, para cada ítem, un vector de features precomputado (por ejemplo, embeddings de BERT o ResNet de portadas) mediante un layer:

```
Embedding(input_dim=num_items,
          output_dim=D,
          embeddings_initializer=Constant(embeddings),
          trainable=False)
```

que hace un `gather` de tamaño (`batch_size, D`). Estos features se concatenan a los embeddings aprendidos antes de las capas MLP, permitiendo al modelo aprovechar tanto señales colaborativas como contenido.

4.2.3. VISUAL BAYESIAN PERSONALIZED RANKING (VBPR)

El modelo VBPR está diseñado para incorporar features visuales de los ítems al modelo BPR de base. Esto se puede extender para utilizar cualquier tipo de información adicional sobre los ítems, como los embeddings generados por BERT. Para manejar features de alta dimensionalidad (los features generados por VGG16 son de 4096 dimensiones), VBPR implementa una capa aprendida de proyección para reducir dimensionalidad a una cantidad d . En este caso, se define $d = 256$, mientras que se utilizan $k = 64$ dimensiones para la matriz de factores latentes de usuarios e ítems de BPR.

5. Evaluación

5.1. Métricas Utilizadas

Para medir el rendimiento de los modelos de recomendación propuestos, se utilizaron métricas estándar de evaluación en sistemas de recomendación. Estas métricas se enfocan en evaluar tanto la precisión de las recomendaciones como su diversidad y novedad:

- **P@K (*Precision*)**: mide la proporción de ítems relevantes entre los primeros K recomendados.
- **NDCG@K (*Normalized Discounted Cumulative Gain*)**: considera no solo la relevancia de los ítems recomendados, sino también su posición en la lista, penalizando ítems relevantes que se encuentran en posiciones inferiores.
- **MAP@K (*Mean Average Precision*)**: calcula la precisión promedio a lo largo de las posiciones relevantes, y luego promedia sobre todos los usuarios.
- **AUC (*Area Under the ROC Curve*)**: evalúa la capacidad del modelo para asignar puntuaciones más altas a ítems relevantes que a irrelevantes.
- **Div@K (*Intra-List Diversity*)**: mide qué tan diferentes son entre sí los ítems recomendados, favoreciendo listas menos redundantes.
- **Nov@K (*Novelty*)**: captura la capacidad del modelo de recomendar ítems nuevos y desconocidos para el usuario.

5.2. Protocolo de Evaluación

Se utilizó un esquema de evaluación *offline*, utilizando datos históricos para el entrenamiento y validación. Para cada usuario, se utilizaron sus interacciones pasadas como conjunto de entrenamiento, y se reservó un subconjunto para el conjunto de validación. Los modelos generan una lista de

recomendaciones para cada usuario, las cuales son posteriormente comparados con los ítems del conjunto de prueba para su evaluación.

En todos los experimentos se generaron top- K recomendaciones, con $K = 5, 10, 20$, para realizar un análisis integral. Los ítems ya vistos por el usuario en entrenamiento fueron excluidos de las listas de recomendación. Los resultados presentados corresponden al promedio de cada métrica sobre todos los usuarios evaluados. Esta metodología permite comparar de manera justa diferentes enfoques de recomendación, tanto colaborativos como basados en contenido.

6. Resultados y Análisis

En esta sección se presentan los resultados obtenidos por los distintos modelos evaluados en el problema de recomendación de libros. En la Tabla 1 se resumen los resultados obtenidos presentando los modelos con los mejores resultados por categoría, junto con las métricas claves $P@K$, $NDCG@K$ y $Nov@K$. Los resultados completos de todos los modelos se pueden observar en la Tabla 7 en la sección de Anexos, junto con sus desempeños en cada métrica evaluada.

6.1. Análisis de sensibilidad

En la Tabla 2 se comparan los modelos que utilizan distintos conjuntos de información adicional. Se puede observar que en términos de $P@10$ y $NDCG@10$, utilizar información adicional puede deteriorar el rendimiento en el caso de FM y NCF, mientras que en VBPR mejora. Esto se puede deber a que la información adicional, como los features extraídos de las portadas, pueden introducir ruido al modelo por sobre los datos de interacciones pasadas entre usuarios y libros. En términos de $Nov@10$, pareciera que hay una mejora para los modelos que utilizan información adicional. Esto también se puede relacionar con el ruido que introducen estos datos, pero también con que ahora los modelos pueden recomendar no solo en base a interacciones pasadas, sino que por ejemplo en base a descripciones o portadas similares.

Otra observación es que más información no necesariamente se traduce en mejores recomendaciones. Para NCF es mejor trabajar solamente con los embeddings de BERT que combinar estos con las features de ResNet50, y para VBPR es mejor trabajar solamente con los features de VGG16 que combinar estos con los embeddings de BERT.

6.2. Análisis de resultados

Exactitud vs. diversidad y novedad. ALS, como representante de los métodos colaborativos clásicos, ofrece la mejor calidad de ranking ($P@K$ y $NDCG@K$) en todos los escenarios de K . Sin embargo, sacrifica diversidad y novedad, pues sus recomendaciones tienden a converger en

Table 1. Resumen comparativo de métodos evaluados a diferentes valores de K .

| Método | P@5 | NDCG@5 | Nov@5 |
|---------------------|---------------|---------------|---------------|
| Random | 0.0040 | 0.0050 | 0.9639 |
| ALS | 0.2660 | 0.2907 | 0.8243 |
| LightFM (warp) | 0.0459 | 0.1523 | 0.4974 |
| LightFM (ResNet) | 0.0559 | 0.1145 | 0.5214 |
| NCF (BERT) | 0.1340 | 0.1495 | 0.6285 |
| NCF (BERT & ResNet) | 0.1460 | 0.1597 | 0.6395 |
| VBPR (VGG16) | 0.1820 | 0.2069 | 0.6875 |
| VBPR (BERT & VGG16) | 0.1920 | 0.2141 | 0.6395 |

(a) $K = 5$

| Método | P@10 | NDCG@10 | Nov@10 |
|---------------------|---------------|---------------|---------------|
| Random | 0.0030 | 0.0027 | 0.9692 |
| ALS | 0.1930 | 0.2309 | 0.8311 |
| LightFM (warp) | 0.0450 | 0.1472 | 0.5543 |
| LightFM (ResNet) | 0.0490 | 0.1029 | 0.5724 |
| NCF (Vanilla) | 0.1320 | 0.1531 | 0.6243 |
| NCF (BERT) | 0.1340 | 0.1433 | 0.6591 |
| VBPR (VGG16) | 0.1650 | 0.1810 | 0.7109 |
| VBPR (BERT & VGG16) | 0.1550 | 0.1785 | 0.7228 |

(b) $K = 10$

| Método | P@20 | NDCG@20 | Nov@20 |
|---------------------|---------------|---------------|---------------|
| Random | 0.0030 | 0.0020 | 0.9687 |
| ALS | 0.3020 | 0.2918 | 0.8438 |
| LightFM (BERT) | 0.0420 | 0.1268 | 0.5794 |
| LightFM (ResNet) | 0.0430 | 0.1109 | 0.4633 |
| NCF (Vanilla) | 0.1055 | 0.1890 | 0.6716 |
| NCF (BERT) | 0.1110 | 0.1924 | 0.6897 |
| VBPR (VGG16) | 0.1205 | 0.2356 | 0.7513 |
| VBPR (BERT & VGG16) | 0.1275 | 0.2369 | 0.7567 |

(c) $K = 20$

ítems populares. Por el contrario, *BPR* produce listas con altos niveles de novedad, lo que lo hace apropiado cuando el objetivo es exponer al usuario a contenido menos conocido.

Impacto de la multimodalidad. Los modelos multimodales (p. ej., VBPR con embeddings de BERT y VGG16) destacan por su elevado AUC y diversidad (ver Tabla 4). Esto confirma que incorporar características visuales y textuales simultáneamente enriquece el espacio latente, generando recomendaciones más variadas sin perder demasiada precisión.

Ordenamiento global (MAP). LightFM con pérdida WARP sobresale en $MAP@K$, especialmente en $K = 5$ (0.2783) y $K = 10$ (0.2793), lo que indica que, aunque

Table 2. Resumen comparativo de métodos multimodales para $K = 10$.

| Método | P@10 | NDCG@10 | Nov@10 |
|---------------------|--------|---------|--------|
| LightFM (warp) | 0.0450 | 0.1472 | 0.5543 |
| LightFM (BERT) | 0.0440 | 0.0936 | 0.5175 |
| LightFM (ResNet) | 0.0490 | 0.1029 | 0.5724 |
| NCF (Vanilla) | 0.1320 | 0.1531 | 0.6243 |
| NCF (BERT) | 0.1340 | 0.1433 | 0.6591 |
| NCF (ResNet) | 0.1180 | 0.1254 | 0.7228 |
| NCF (BERT & ResNet) | 0.1150 | 0.1344 | 0.6785 |
| BPR | 0.0890 | 0.0984 | 0.9293 |
| VBPR (BERT) | 0.1460 | 0.1609 | 0.7166 |
| VBPR (VGG16) | 0.1650 | 0.1810 | 0.7109 |
| VBPR (BERT & VGG16) | 0.1550 | 0.1785 | 0.7228 |

no coloca tantos ítems relevantes en el top- K como ALS, ordena mejor los ítems relevantes a lo largo de toda la lista. Esto puede resultar ventajoso en interfaces donde se muestre más allá del primer bloque de recomendaciones.

Modelos neuronales (NCF). Las variantes de NCF quedan en posiciones intermedias: ofrecen un buen equilibrio entre precisión y novedad, pero no superan ni a ALS en exactitud pura ni a VBPR en diversidad.

6.3. Análisis de ejemplos de recomendación

De las Tablas 3 a 6 del Anexo, es posible ver que NCF en todas sus versiones recomienda el libro *The Great Gatsby* para el usuario dado, lo que puede indicar que la información colaborativa se mantiene en todos los modelos. En contraste, el libro *Anna Karenina* es recomendado por NCF Vanilla y con features de ResNet50, mientras que el libro *The Goldfinch* es recomendado solamente por NCF con información adicional, pero no Vanilla. Esto puede apuntar a las diferencias entre los modelos y cómo utilizan las señales de la información adicional, sean embeddings o features. A pesar de lo anterior, no hay patrones claros en las distintas recomendaciones.

7. Conclusiones

En este estudio comparativo de métodos de recomendación de libros revela que no existe un único modelo óptimo. La elección de un modelo depende del objetivo de la recomendación y, a partir de este, definir la métrica más relevante que se quiera optimizar. En la recomendación de libros sobre este dataset, **ALS** es insuperable en términos de precisión (P@K, NDCG@K), lo que es ideal cuando la prioridad es maximizar la relevancia de las primeras posiciones de la recomendación. **BPR** es la mejor opción para maximizar novedad sin bajar demasiado la precisión, siendo solo superado por la recomendación aleatoria en novedad. Por otro

lado, métodos multimodales como **VBPR (BERT+VGG16)** logran un compromiso entre diversidad y discriminación global (AUC), demostrando el valor de combinar señales textuales y visuales. Finalmente, el modelo **LightFM (warp)** destaca en MAP@K, lo que significa que hace rankings globales más consistentes. Esto es útil cuando la interfaz de recomendación despliega más de K ítems.

Este estudio abre camino a interesantes futuras investigaciones. Una de ellas es la incorporación de modelos multimodales avanzados como CLIP o SigLIP, los cuales permiten mapear texto e imágenes en un espacio semántico compartido (Radford et al., 2021). Otra línea prometedora es el desarrollo de modelos de ensamble dinámicos, en los que la combinación entre distintos modelos (colaborativo, textual, visual) se ajuste automáticamente para cada usuario según su historial o perfil.

References

- Devlin, J., Chang, M., Lee, K., and Toutanova, K. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL <http://arxiv.org/abs/1810.04805>.
- Iwana, B. K. and Uchida, S. Judging a book by its cover. *CoRR*, abs/1610.09204, 2016. URL <http://arxiv.org/abs/1610.09204>.
- Javaji, S. R. and Sarode, K. Multi-bert for embeddings for recommendation system, 2023. URL <https://arxiv.org/abs/2308.13050>.
- Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. Learning transferable visual models from natural language supervision. *CoRR*, abs/2103.00020, 2021. URL <https://arxiv.org/abs/2103.00020>.
- Rendle, S., Freudenthaler, C., Gantner, Z., and Schmidt-Thieme, L. Bpr: Bayesian personalized ranking from implicit feedback, 2012. URL <https://arxiv.org/abs/1205.2618>.
- Shlens, J. A tutorial on principal component analysis, 2014. URL <https://arxiv.org/abs/1404.1100>.
- Zachariah, D., Sundin, M., Jansson, M., and Chatterjee, S. Alternating least-squares for low-rank matrix reconstruction. *IEEE Signal Processing Letters*, 19(4):231–234, April 2012. ISSN 1558-2361. doi: 10.1109/lsp.2012.2188026. URL <http://dx.doi.org/10.1109/LSP.2012.2188026>.

Anexos

Table 3. Ejemplos de Interacciones de usuario

| Título | Portada |
|--|--|
| 1. The Alchemist |  |
| 2. The Curious Incident of the Dog in the Night-Time |  |
| 3. Girl with a Pearl Earring |  |
| 4. Pride and Prejudice |  |
| 5. To Kill a Mockingbird |  |

Table 4. Ejemplos de recomendación para NCF Vanilla

| Título | Portada |
|----------------------------------|---|
| 1. The Great Gatsby |  |
| 2. The Time Traveler's Wife |  |
| 3. Anna Karenina |  |
| 4. The Joy Luck Club |  |
| 5. One Hundred Years of Solitude |  |

Table 5. Ejemplos de recomendación para NCF con Resnet



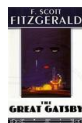

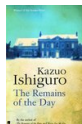
| Título | Portada |
|---------------------------|---|
| 1. Anna Karenina |  |
| 2. The Goldfinch |  |
| 3. The Great Gatsby |  |
| 4. Cutting for Stone |  |
| 5. The Remains of the Day |  |

Table 6. Ejemplos de recomendación para NCF con BERT & ResNet

| Título | Portada |
|---------------------------|---|
| 1. Gone Girl |  |
| 2. The Girl on the Train |  |
| 3. The Great Gatsby |  |
| 4. The Goldfinch |  |
| 5. The Catcher in the Rye |  |

Table 7. Comparación de métodos evaluados a diferentes valores de K .

| Método | P@5 | NDCG@5 | MAP@5 | AUC | Div@5 | Nov@5 |
|---------------------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Random | 0.0040 | 0.0050 | 0.0026 | 0.5004 | 0.3164 | 0.9639 |
| Most Popular | 0.1000 | 0.1091 | 0.0634 | 0.5244 | 0.2893 | 0.2481 |
| ALS | 0.2660 | 0.2907 | 0.1881 | 0.9198 | 0.2774 | 0.8243 |
| BPR | 0.1040 | 0.1108 | 0.0592 | 0.9183 | 0.2794 | 0.9245 |
| IBCF | 0.1840 | 0.1933 | 0.1229 | 0.8279 | 0.3239 | 0.5571 |
| LightFM (warp) | 0.0459 | 0.1523 | 0.2783 | 0.9409 | 0.3209 | 0.4974 |
| LightFM (w. BERT) | 0.0440 | 0.1109 | 0.2226 | 0.9393 | 0.3110 | 0.4633 |
| LightFM (w. ResNet) | 0.0559 | 0.1145 | 0.2276 | 0.9398 | 0.2988 | 0.5214 |
| NCF (Vanilla) | 0.1320 | 0.1531 | 0.0913 | 0.9579 | 0.3053 | 0.5888 |
| NCF (w. BERT embeddings) | 0.1340 | 0.1495 | 0.0840 | 0.9563 | 0.3062 | 0.6285 |
| NCF (w. ResNet images) | 0.1140 | 0.1290 | 0.0735 | 0.9532 | 0.3104 | 0.7012 |
| NCF (w. BERT and ResNet) | 0.1460 | 0.1597 | 0.0945 | 0.9592 | 0.3125 | 0.6395 |
| VBPR (w. BERT embeddings) | 0.1780 | 0.1873 | 0.1161 | 0.9628 | 0.2916 | 0.6976 |
| VBPR (w. VGG16 images) | 0.1820 | 0.2069 | 0.1217 | 0.9607 | 0.6462 | 0.6875 |
| VBPR (w. BERT and VGG16) | 0.1920 | 0.2141 | 0.1275 | 0.9635 | 0.9515 | 0.6395 |

 (a) $K = 5$

| Método | P@10 | NDCG@10 | MAP@10 | AUC | Div@10 | Nov@10 |
|---------------------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Random | 0.0030 | 0.0027 | 0.0007 | 0.5003 | 0.3228 | 0.9692 |
| Most Popular | 0.0760 | 0.0890 | 0.0373 | 0.5369 | 0.3283 | 0.3235 |
| ALS | 0.1930 | 0.2309 | 0.1148 | 0.9198 | 0.2814 | 0.8311 |
| BPR | 0.0890 | 0.0984 | 0.0382 | 0.9183 | 0.2808 | 0.9293 |
| IBCF | 0.1600 | 0.1738 | 0.0815 | 0.8279 | 0.3212 | 0.5933 |
| LightFM (warp) | 0.0450 | 0.1472 | 0.2793 | 0.9409 | 0.3115 | 0.5543 |
| LightFM (w. BERT) | 0.0440 | 0.0936 | 0.2155 | 0.9393 | 0.3255 | 0.5175 |
| LightFM (w. ResNet) | 0.0490 | 0.1029 | 0.2451 | 0.9398 | 0.2975 | 0.5724 |
| NCF (Vanilla) | 0.1320 | 0.1531 | 0.0621 | 0.9579 | 0.3041 | 0.6243 |
| NCF (w. BERT embeddings) | 0.1340 | 0.1433 | 0.0585 | 0.9563 | 0.3097 | 0.6591 |
| NCF (w. ResNet images) | 0.1180 | 0.1254 | 0.0522 | 0.9532 | 0.3080 | 0.7228 |
| NCF (w. BERT and ResNet) | 0.1150 | 0.1344 | 0.0593 | 0.9592 | 0.3070 | 0.6785 |
| VBPR (w. BERT embeddings) | 0.1460 | 0.1609 | 0.0734 | 0.9628 | 0.3007 | 0.7166 |
| VBPR (w. VGG16 images) | 0.1650 | 0.1810 | 0.0837 | 0.9607 | 0.6543 | 0.7109 |
| VBPR (w. BERT and VGG16) | 0.1550 | 0.1785 | 0.0813 | 0.9635 | 0.9550 | 0.7228 |

 (b) $K = 10$

| Método | P@20 | NDCG@20 | MAP@20 | AUC | Div@20 | Nov@20 |
|---------------------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Random | 0.0030 | 0.0020 | 0.0003 | 0.4991 | 0.3211 | 0.9687 |
| Most Popular | 0.1290 | 0.1183 | 0.0445 | 0.5623 | 0.3362 | 0.4214 |
| ALS | 0.3020 | 0.2918 | 0.1416 | 0.9198 | 0.2847 | 0.8438 |
| BPR | 0.1440 | 0.1294 | 0.0474 | 0.9183 | 0.2847 | 0.9361 |
| IBCF | 0.2470 | 0.2216 | 0.0997 | 0.8279 | 0.3134 | 0.6534 |
| LightFM (warp) | 0.0410 | 0.1472 | 0.2141 | 0.9409 | 0.3039 | 0.6359 |
| LightFM (w. BERT) | 0.0420 | 0.1268 | 0.2034 | 0.9393 | 0.3257 | 0.5794 |
| LightFM (w. ResNet) | 0.0430 | 0.1109 | 0.2226 | 0.9398 | 0.3110 | 0.4633 |
| NCF (Vanilla) | 0.1055 | 0.1890 | 0.0788 | 0.9579 | 0.3058 | 0.6716 |
| NCF (w. BERT embeddings) | 0.1110 | 0.1924 | 0.0751 | 0.9563 | 0.3109 | 0.6897 |
| NCF (w. ResNet images) | 0.0990 | 0.1715 | 0.0672 | 0.9532 | 0.3089 | 0.7407 |
| NCF (w. BERT and ResNet) | 0.0975 | 0.1788 | 0.0735 | 0.9592 | 0.3085 | 0.7216 |
| VBPR (w. BERT embeddings) | 0.1285 | 0.2223 | 0.0970 | 0.9628 | 0.3026 | 0.7456 |
| VBPR (w. VGG16 images) | 0.1205 | 0.2356 | 0.0977 | 0.9607 | 0.6549 | 0.7513 |
| VBPR (w. BERT and VGG16) | 0.1275 | 0.2369 | 0.1023 | 0.9635 | 0.9636 | 0.7567 |

 (c) $K = 20$

Repositorio

[Repositorio de GitHub](#)