



UA

**RecGraf : Recomendaciones Basadas en Grafos para PYMEs**

*Alexander Andonov Aracil*  
**DNI: 74526740L**

*Daniel Asensi Roch*  
**DNI: 48776120C**

**11 de noviembre de 2023**

# Índice

<b>1. Introducción</b>	<b>2</b>
1.1. Motivación . . . . .	2
1.2. Objetivos . . . . .	3
<b>2. Estudio de mercado</b>	<b>4</b>
2.1. Elección del método . . . . .	5
2.2. El mercado actual . . . . .	5
2.2.1. Productos similares . . . . .	5
<b>3. Metodología</b>	<b>7</b>
3.1. LightGCN . . . . .	7
3.1.1. Señal colaborativa . . . . .	7
3.1.2. Arquitectura . . . . .	8
3.2. Datasets . . . . .	10
3.2.1. MovieLens . . . . .	10
3.2.2. Amazon-Book . . . . .	11
3.3. Implementaciones del modelo . . . . .	12
<b>4. Resultados</b>	<b>13</b>
4.1. Análisis inicial del modelo . . . . .	13
4.1.1. Análisis cualitativo . . . . .	13
4.2. Posibles optimizaciones . . . . .	14
4.2.1. Adam (Adaptive Moment Estimation) . . . . .	14
4.2.2. SGD (Stochastic Gradient Descent) con Momentum . . . . .	15
4.2.3. RMSprop (Root Mean Square Propagation) . . . . .	15
4.2.4. Adadelta . . . . .	16
4.2.5. Adagrad (Adaptive Gradient Algorithm) . . . . .	16
4.2.6. Conclusiones . . . . .	17
4.3. Análisis de la incertidumbre . . . . .	18
4.3.1. Naturaleza Probabilística de las Puntuaciones . . . . .	18
4.3.2. Grafos Bipartitos y Estructuras de Alto Orden . . . . .	18
4.3.3. Ranking Basado en Puntuaciones . . . . .	18
4.3.4. Incertidumbre Fundamental . . . . .	18
<b>5. Aplicaciones</b>	<b>19</b>
<b>6. Presupuesto para la Implementación de RecGraf</b>	<b>21</b>
<b>7. Conclusiones</b>	<b>23</b>

# 1. Introducción

En el dinámico mundo del comercio electrónico de hoy, donde la tecnología y la interconexión global avanzan a pasos agigantados, destacar en medio de una competencia feroz es esencial. Una herramienta que ha demostrado ser crucial para lograr este objetivo es la capacidad de ofrecer recomendaciones de productos personalizadas y precisas. Mientras que las grandes corporaciones, con vastos recursos a su disposición, han implementado sistemas de recomendación avanzados basados en técnicas de inteligencia artificial y aprendizaje automático, las [Pequeñas y medianas empresas \(PYMES\)](#) a menudo se encuentran en desventaja [1].

Estas pequeñas y medianas empresas, a pesar de ser pilares fundamentales en la economía global, enfrentan restricciones significativas en términos de recursos, infraestructura y expertise técnico, lo que les dificulta acceder a soluciones de recomendación de vanguardia [2]. Sin embargo, su relevancia en la generación de empleo y contribución al [Producto Interior Bruto \(PIB\)](#) en muchos países subraya la necesidad de nivelar el campo de juego [3].

Con este panorama en mente, el presente documento se enfoca en el diseño, desarrollo e implementación de un sistema de recomendación de productos específicamente basado en Filtrado Colaborativo, una técnica que se centra en los patrones de comportamiento de los usuarios para hacer recomendaciones. Este enfoque ha sido elegido por su eficacia y, crucialmente, por su adaptabilidad a entornos con recursos computacionales limitados, lo que lo hace ideal para e-commerces de [PYMES](#). A través de este proyecto, aspiramos a cerrar la brecha tecnológica, ofreciendo a estas empresas una solución de recomendación que no solo sea competitiva, sino también viable y adecuada a sus capacidades y necesidades.

En las siguientes secciones, se establecerá la importancia de adaptar tecnologías avanzadas de recomendación al contexto de las [PYMES](#), se explorarán los desafíos asociados y se detallarán las motivaciones y objetivos que guían este proyecto.

## 1.1. Motivación

En el contexto contemporáneo, marcado por la creciente predominancia de la digitalización y el auge del comercio electrónico [4], se identifica una demanda creciente por sistemas que faciliten recomendaciones de productos adecuadas y personalizadas. Grandes conglomerados de comercio electrónico, como Amazon, se han beneficiado enormemente de la implementación de sistemas de recomendación avanzados, que, haciendo uso de técnicas de aprendizaje profundo, descifran patrones de compra y afinan las preferencias de los usuarios. No obstante, estas tecnologías avanzadas, pese a su eficacia probada, a menudo presuponen una robusta infraestructura computacional y una gran cantidad de datos, lo que las coloca fuera del alcance de las [PYMES](#). Estas empresas, que carecen de los recursos significativos que poseen las grandes corporaciones, encuentran dificultades para acceder a tales tecnologías.

Adicionalmente, es fundamental subrayar que las [PYMES](#), por su naturaleza y tamaño, poseen dinámicas y necesidades distintas a las de las grandes corporaciones. Mientras que las grandes empresas pueden invertir en infraestructuras complejas y en la adquisición de vastos conjuntos de datos, las [PYMES](#) a menudo operan en nichos de mercado más específi-

cos, donde la personalización y la cercanía con el cliente son esenciales. En este sentido, la falta de acceso a sistemas de recomendación avanzados no solo representa una desventaja tecnológica, sino que también limita la capacidad de estas empresas para optimizar la experiencia del cliente y adaptarse a sus preferencias de manera ágil. Por ende, surge la necesidad de desarrollar soluciones adaptadas que permitan a las **PYMES** aprovechar las ventajas del aprendizaje profundo y las redes neuronales, sin la necesidad de grandes inversiones o complejidades técnicas.

## 1.2. Objetivos

Partiendo de los problemas que existen en los sistemas de recomendación de productos en tiendas online y las posibles ventajas que pueda aportar el uso de un sistema basado en **Graph Neural Networks (GNN)**, el objetivo principal del proyecto es desarrollar un sistema de recomendaciones de menor escala, proponiendo un modelo accesible y escalable. A través de este enfoque, este trabajo aspira a proporcionar una solución práctica y efectiva, que permita a las **PYMES** implementar sistemas de recomendaciones avanzados, mejorando así su competitividad y capacidad para ofrecer servicios personalizados a sus clientes, sin la necesidad de intervenir en infraestructuras tecnológicas costosas. Para alcanzar dicho objetivo se plantean los siguientes sub-objetivos:

- **OE1:** Realizar un estudio de mercado de los productos y soluciones que se encuentran actualmente disponibles en el mercado. Tras este estudio podremos concluir los factores que harán que nuestra solución se distinga de las demás.
- **OE2:** Elaborar un diseño conceptual y arquitectónico del sistema de recomendación basado en **GNN**, definiendo las características, los componentes y la estructura del modelo, así como los métodos y técnicas de aprendizaje profundo a emplear.
- **OE3:** Implementar el modelo de **GNN** propuesto, utilizando herramientas y frameworks adecuados para la construcción de redes neuronales de grafos, y adaptando el modelo a los requisitos y limitaciones específicas de las **PYMES**.
- **OE4:** Realizar una evaluación exhaustiva del sistema de recomendación desarrollado, utilizando métricas de rendimiento pertinentes para medir la precisión, la relevancia y la eficacia de las recomendaciones generadas, y comparando los resultados con los de otros sistemas de recomendación existentes.
- **OE5:** Implementar pruebas de campo del modelo en entornos reales de **PYMES**, analizando la respuesta y el comportamiento del sistema en condiciones de operación y con datos de usuarios y productos reales, para validar su aplicabilidad y robustez en el mundo real.

## 2. Estudio de mercado

Los sistemas de recomendación de productos son herramientas esenciales en el ámbito del comercio electrónico y se clasifican, generalmente, en tres categorías principales [5], cada una con su propia metodología y enfoque para proporcionar recomendaciones a los usuarios:

- **Filtrado Colaborativo:** realiza recomendaciones basándose en los patrones de comportamiento de los usuarios. Este sistema analiza las interacciones y preferencias de los usuarios para identificar similitudes entre ellos. Como se puede apreciar en la figura 1. Este método es especialmente efectivo para descubrir intereses implícitos de los usuarios, basándose en las acciones y preferencias de usuarios similares.
- **Filtrado Basado en Contenido:** realiza recomendaciones fundamentadas en la descripción de los productos y el perfil del usuario. Este método analiza las características de los productos que el usuario ha mostrado interés previamente y busca productos similares para recomendar.
- **Sistemas de Recomendación Híbridos:** combinan elementos del Filtrado Colaborativo y el Filtrado Basado en Contenido. Estos sistemas utilizan tanto los datos del comportamiento del usuario como las características de los productos para generar recomendaciones más precisas y personalizadas, aprovechando las ventajas de ambos métodos.



Figura 1: Filtrado Colaborativo explicación visual

## 2.1. Elección del método

En el contexto de este trabajo, hemos decidido implementar un sistema basado en **Filtrado Colaborativo**. Este enfoque es el más interesante y beneficioso para las **PYMES**, ya que permite descubrir y anticipar las preferencias de los usuarios sin necesidad de disponer de extensas descripciones o características de los productos. Por otro lado, este tipo de filtrado permite adaptarse de manera efectiva a los cambios en los comportamientos y preferencias de los usuarios, lo que resulta crucial para pequeñas y medianas empresas que buscan mantenerse competitivas y relevantes en mercados en constante evolución [6].

## 2.2. El mercado actual

En el mercado actual, los sistemas de recomendación son omnipresentes y se utilizan en una variedad de campos, desde el comercio electrónico hasta las plataformas de streaming de medios, pasando por las redes sociales. Estos sistemas son vitales para aumentar la retención de usuarios y mejorar la experiencia del cliente, proporcionando sugerencias personalizadas y relevantes.

### 2.2.1. Productos similares

En el ámbito de sistemas de recomendación para **PYMES**, existen diversas soluciones que buscan satisfacer las necesidades específicas de este segmento del mercado. A continuación, se describen tres productos que son representativos en este campo:

- **Wiser Product Recommendations de Shopify:** Wiser es una aplicación de recomendación de productos integrada en la plataforma de comercio electrónico Shopify. Esta herramienta utiliza algoritmos inteligentes para analizar el comportamiento de los usuarios y sus interacciones con los productos, y en base a esto, genera recomendaciones personalizadas. Wiser es especialmente útil para **PYMES** que utilizan Shopify para sus tiendas en línea, ya que ofrece una integración sencilla y permite a estas empresas mejorar la experiencia de compra de sus clientes y potencialmente aumentar sus ventas, todo esto sin necesidad de un desarrollo interno extenso o inversiones significativas en infraestructura tecnológica [7].
- **Crossing Minds:** ofrece una solución de recomendación basada en técnicas de aprendizaje profundo y automático para proporcionar recomendaciones altamente personalizadas y precisas. Su tecnología se basa en el análisis de grandes cantidades de datos y en la identificación de patrones complejos en el comportamiento del usuario [8].
- **Dynamic Yield:** es una plataforma de optimización de la experiencia del usuario que ofrece funcionalidades de recomendación de productos entre sus soluciones. Utiliza algoritmos de aprendizaje automático para analizar el comportamiento del usuario en tiempo real y adaptar las recomendaciones y el contenido, permitiendo a estas empresas optimizar la experiencia de sus clientes y ofrecer productos y contenidos que se alinean de manera efectiva con sus preferencias y necesidades individuales [9].

- **Recommendify:** Esta es una plataforma de recomendación especializada en productos físicos para tiendas en línea. Gracias a su avanzada tecnología de aprendizaje automático, Recommendify analiza las interacciones de los usuarios, historiales de compra y preferencias para generar recomendaciones de productos altamente relevantes. Pensada especialmente para [PYMES](#), esta herramienta no sólo se integra fácilmente con las principales plataformas de comercio electrónico, sino que también ofrece un panel de control intuitivo que permite a los propietarios de tiendas ajustar las recomendaciones según necesidades específicas, promociones o inventario. Además, proporciona informes detallados sobre el rendimiento de las recomendaciones, lo que permite a las empresas ajustar sus estrategias en función de datos concretos [10].

Característica	Wiser	Crossing Minds	Dynamic Yield	Recommendify
Integración con plataformas populares	Shopify	-	Múltiples	Múltiples
Técnicas utilizadas	Algoritmos inteligentes	Aprendizaje profundo	Aprendizaje automático	Aprendizaje automático
Enfoque principal	Comportamiento del usuario	Patrones complejos	Experiencia del usuario	Productos físicos
Adaptabilidad para <a href="#">PYMES</a>	Alta	Media	Alta	Alta
Panel de control	No	No	Sí	Sí
Informes de rendimiento	No	No	Sí	Sí
Integración con otros sistemas	Baja	Media	Alta	Alta

Cuadro 1: Tabla comparativa de sistemas de recomendación de productos.

### 3. Metodología

El objetivo del método a desarrollar es obtener puntuaciones por cada posible recomendación a un determinado usuario (estas puntuaciones permitirán modelar la incertidumbre del problema). En los casos en los que se requiere una recomendación de tamaño fijo se suelen seleccionar los  $k$  ítems con mayor puntuación.

#### 3.1. LightGCN

La Red de [Graph Convolutional Network \(GCN\)](#) se ha consolidado como el estándar de referencia para el filtrado colaborativo. A pesar de su adopción, las razones detrás de su eficacia para las recomendaciones no están completamente claras. Estudios previos que han adaptado la [GCN](#) para recomendaciones no ofrecen un análisis profundo de la misma. Originalmente, la [GCN](#) fue diseñada para tareas de clasificación de grafos y viene con una variedad de operaciones de redes neuronales. No obstante, investigaciones empíricas sugieren que dos de los diseños más prevalentes en las [GCN](#) —la transformación de características y la activación no lineal— tienen un impacto limitado en el rendimiento del filtrado colaborativo. De hecho, su inclusión puede complicar el entrenamiento y reducir la eficacia de las recomendaciones. En este contexto, surge la propuesta de simplificar el diseño de la [GCN](#) para adaptarlo mejor a las recomendaciones. Se presenta un nuevo modelo denominado LightGCN, que retiene solo el componente esencial de la [GCN](#): la agregación de vecindarios para el filtrado colaborativo. De manera específica, LightGCN aprende embeddings de usuarios y productos mediante una propagación lineal en el grafo de interacción usuario-producto, y posteriormente utiliza la suma ponderada de los embeddings adquiridos en todas las capas como el embedding <sup>1</sup> definitivo. Este diseño lineal y simplificado no solo es más fácil de implementar y entrenar, sino que también ha demostrado mejoras significativas (alrededor del 16.0 % de mejora relativa en promedio) sobre el [Neural Graph Collaborative Filtering \(NGCF\)](#) —un modelo de recomendación basado en [GCN](#) avanzado— en condiciones experimentales idénticas. Se ofrecen análisis adicionales que justifican la eficiencia del LightGCN desde perspectivas analíticas y empíricas [11].

##### 3.1.1. Señal colaborativa

En el contexto del filtrado colaborativo, la **señal colaborativa** se refiere a patrones que pueden descubrirse a través del análisis de la forma en que diferentes usuarios interactúan con diferentes objetos como productos o películas.

Por ejemplo, en un sistema de recomendación de películas, si dos usuarios tienen gustos similares en películas y uno de ellos muestra preferencia por una película que el otro aún no ha visto, la señal colaborativa sugiere que probablemente a este último también le gustará

---

<sup>1</sup>Los embeddings son representaciones vectoriales en un espacio de alta dimensión, utilizadas para modelar y capturar las características y relaciones de diferentes entidades, como usuarios y productos en sistemas de recomendación. Son esenciales para identificar similitudes y diferencias en un espacio matemático, permitiendo algoritmos más eficientes y precisos en tareas como el filtrado colaborativo.



esa película. Por lo tanto, capturar la señal colaborativa implica analizar y aprender de estas interacciones para hacer recomendaciones más precisas.

Las interacciones entre usuarios e ítems pueden ser representadas como un grafo bipartito (ver figura 9a). Las técnicas avanzadas, como la propagación de etiquetas o la factorización de matrices en grafos, pueden aprovechar la estructura de alto orden de estos grafos para hacer recomendaciones. Al considerar estructuras y relaciones de alto orden en los datos (ver figura 9b), los sistemas de recomendación pueden capturar patrones más complejos y sutiles en las interacciones de los usuarios [12], lo que potencialmente lleva a recomendaciones más precisas y personalizadas.

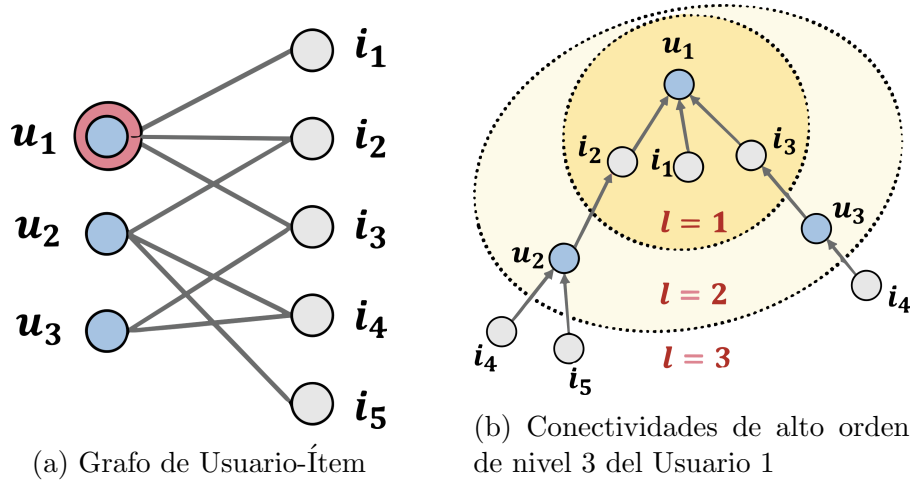


Figura 2: Imagen adaptada de <https://github.com/recommenders-team>.

### 3.1.2. Arquitectura

En esta sección se discutirán y analizarán los factores y procedimientos más relevantes de la arquitectura de una LightGCN.

**Graph Convolution** Una capa de **Light Graph Convolution (LGC)** es una suma normalizada que actualiza los valores de los *embeddings* de usuarios e ítems. Se puede expresar mediante la siguiente fórmula recursiva:

$$\begin{aligned} \mathbf{e}_u^{(k+1)} &= \sum_{i \in \mathcal{N}_u} \frac{1}{\sqrt{|\mathcal{N}_u|} \sqrt{|\mathcal{N}_i|}} \mathbf{e}_i^{(k)} \\ \mathbf{e}_i^{(k+1)} &= \sum_{u \in \mathcal{N}_i} \frac{1}{\sqrt{|\mathcal{N}_i|} \sqrt{|\mathcal{N}_u|}} \mathbf{e}_u^{(k)} \end{aligned} \quad (1)$$

Donde:

- $\mathbf{e}_u^{(k)}$ : Embedding de entrada de usuarios
- $\mathbf{e}_i^{(k)}$ : Embedding de entrada de ítems
- $\mathbf{e}_u^{(k+1)}$ : Embedding de salida de usuarios

- $e_i^{(k+1)}$ : Embedding de salida de ítems
- $\frac{1}{\sqrt{N_y}\sqrt{N_i}}$ : Término de normalización

**Combinación de las capas** La predicción del modelo es el producto interno de la combinación de los embeddings de cada capa. Para la combinación de embeddings se realiza una suma ponderada, asignando un factor de importancia a cada capa (ver ecuación 2). Finalmente se obtiene la predicción mediante el producto interno de la combinación de los embeddings de usuario e ítem (ver ecuación 3). La predicción incorporará las puntuaciones necesarias para generar las recomendaciones.

$$\mathbf{e}_u = \sum_{k=0}^K \alpha_k \mathbf{e}_u^{(k)}; \quad \mathbf{e}_i = \sum_{k=0}^K \alpha_k \mathbf{e}_i^{(k)} \quad (2)$$

$$\hat{y}_{ui} = \mathbf{e}_u^T \mathbf{e}_i \quad (3)$$

Donde:

- $K$ : Número de capas
- $\alpha_k$ : Factor de importancia de cada capa
- $e_u^{(k)}$ : Embedding de usuarios de la capa k
- $e_i^{(k)}$ : Embedding de ítems de la capa k
- $\hat{y}_{ui}$ : Predicción del modelo

El factor de importancia utilizado en [11] es uniforme para cada capa ( $\frac{1}{K+1}$ , siendo K el número de capas). Dicho factor se utilizará en la fase de extracción y comparación de resultados (ver sección 4).

**Forma Matricial** Para facilitar la implementación del modelo, es posible expresar las LGC en forma matricial (ver ecuaciones 4 y 5).

$$\mathbf{A} = \begin{pmatrix} \mathbf{0} & \mathbf{R} \\ \mathbf{R}^T & \mathbf{0} \end{pmatrix} \quad (4)$$

$$\mathbf{E}^{(k+1)} = \left( \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \right) \mathbf{E}^{(k)} \quad (5)$$

Donde:

- $R$ : Matriz de Usuario-Interacción
- $A$ : Matriz de adyacencia del grafo de Usuario-Interacción
- $D$ : Matriz de grados del grafo
- $E^{(k)}$ : Matriz de embeddings de entrada
- $E^{(k+1)}$ : Matriz de embeddings de salida

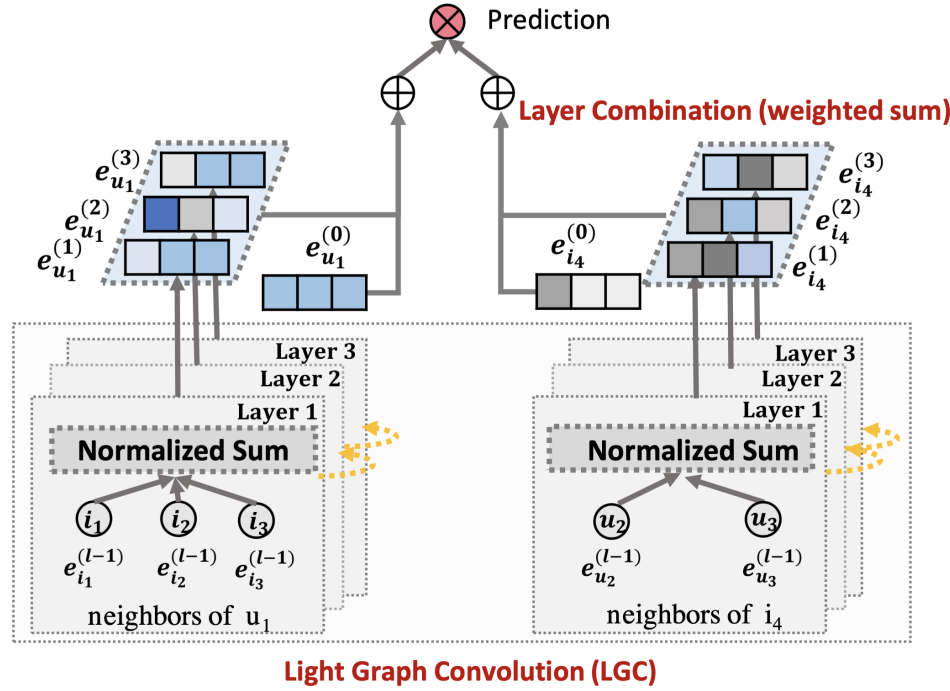


Figura 3: Diagrama de la estructura de una LightGCN. Imagen adaptada de [11].

## 3.2. Datasets

Los datasets actúan como el cimiento sobre el cual se construyen y validan modelos. Proporcionan el contexto necesario y la información cruda que alimenta los algoritmos, permitiendo la generación de insights y la toma de decisiones basadas en datos. En el ámbito de los sistemas de recomendación, la elección del dataset es crucial, ya que determina la relevancia y aplicabilidad de los resultados. En esta sección, nos sumergiremos en dos datasets prominentes que han sido ampliamente utilizados en la investigación de sistemas de recomendación: MovieLens y Amazon-books. Ambos datasets, aunque diferentes en su naturaleza y contenido, ofrecen una rica fuente de información que puede ser aprovechada para entender y predecir las preferencias de los usuarios.

### 3.2.1. MovieLens

El dataset de MovieLens es uno de los conjuntos de datos más populares utilizados en la investigación de sistemas de recomendación y análisis de datos. Fue creado por el GroupLens Research Project en la Universidad de Minnesota. [13]. Sobre este se han realizado diferentes modificaciones para obtener una mejor comprensión de los datos contenidos en el mismo. La adaptación utilizada contiene los siguientes datos:

- **user:** ID del usuario que ha proporcionado una calificación.
- **item:** ID de la película que ha sido calificada.

- **label:** Calificación otorgada por el usuario a la película. En el contexto de MovieLens, esto suele ser una calificación de 1 a 5 estrellas.
- **time:** Marca de tiempo que indica cuándo se realizó la calificación.
- **sex:** Género del usuario (F para femenino, M para masculino).
- **age:** Edad del usuario.
- **occupation:** Ocupación del usuario, que suele estar codificada como un número. En el dataset original de MovieLens, cada número corresponde a una categoría específica de ocupación (por ejemplo, "6" podría ser "administrador").
- **genre1, genre2, genre3:** Géneros asociados con la película. Parece que si una película tiene menos de tres géneros, se utiliza "missing" para llenar los espacios vacíos.

### 3.2.2. Amazon-Book

El dataset de Amazon Product Reviews es un conjunto de datos extenso que cubre reseñas de una amplia variedad de productos vendidos en Amazon. Sin embargo, para ciertas aplicaciones o investigaciones, puede ser útil centrarse en un subconjunto específico de productos. El dataset Amazon-books es precisamente una adaptación o subconjunto de este dataset más grande, centrado exclusivamente en libros vendidos en Amazon. [14] Este dataset es de gran utilidad por los siguientes puntos clave:

- **Específico para libros:** Todas las reseñas y metadatos en este dataset están relacionados con libros. Esto permite a los investigadores y desarrolladores centrarse en las particularidades y patrones específicos de las reseñas y compras de libros.
- **Reseñas y calificaciones:** Al igual que en el dataset principal, se incluyen detalles como el ID del revisor, el nombre del revisor, el texto de la reseña, la calificación general, un resumen de la reseña, y la fecha de la reseña.
- **Metadatos:** Los metadatos específicos de los libros pueden incluir información como el autor, el editor, el año de publicación, el género o categoría del libro, entre otros. Además, se pueden mantener otros metadatos generales como votos de utilidad, imágenes de los libros, y ranking de ventas.
- **Relaciones ítem-ítem:** Estas relaciones, en el contexto de libros, pueden representar patrones como "los lectores que compraron este libro también compraron estos otros libros". Estas relaciones son esenciales para el filtrado colaborativo y para entender las preferencias y patrones de lectura de los usuarios.

### 3.3. Implementaciones del modelo

Cuando se trabaja en la replicación o adaptación de un modelo o técnica propuesta en un artículo científico, es común encontrar múltiples implementaciones disponibles. Estas implementaciones pueden variar en términos de eficiencia, precisión, facilidad de uso, y otros factores. La elección de una implementación específica puede depender de varios criterios, incluyendo los recursos computacionales disponibles, el objetivo del estudio, y la familiaridad con ciertas herramientas o lenguajes.

#### Elección de LightGCN con LibRecommender:

- **Eficiencia:** LightGCN es una variante simplificada de las GCN que se centra en la eficiencia al eliminar ciertas características que no contribuyen significativamente al rendimiento del modelo. Esto lo hace menos intensivo en términos de recursos computacionales.
- **LibRecommender:** Es una biblioteca de Python diseñada específicamente para sistemas de recomendación. Al utilizar una biblioteca especializada como LibRecommender, se puede aprovechar una implementación optimizada y bien mantenida, lo que puede reducir aún más el consumo de recursos y simplificar el proceso de desarrollo. [15]

#### Implementación en Notebooks para Comprensión:

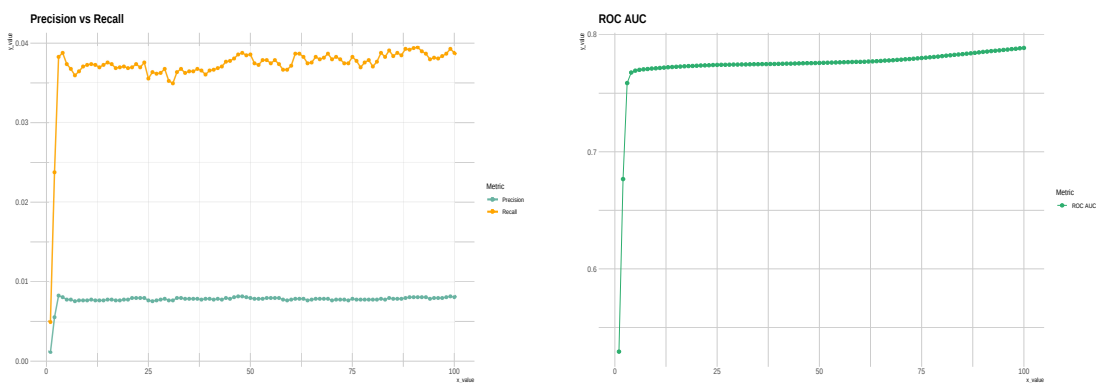
- **Interactividad:** Los notebooks, como Jupyter Notebook, ofrecen un entorno interactivo que permite ejecutar código, visualizar resultados, y documentar el proceso paso a paso. Esta interactividad es especialmente útil para entender y desglosar el funcionamiento de un modelo.
- **Desglose del Modelo:** Al implementar el modelo en un notebook, se tiene la flexibilidad de dividir el modelo en componentes más pequeños, ejecutarlos individualmente, y analizar cada paso. Esto facilita la comprensión de cómo funciona el modelo y qué impacto tiene cada componente en los resultados finales.
- **Documentación y Explicación:** Los notebooks permiten combinar código con texto enriquecido, lo que facilita la documentación del proceso, la explicación de decisiones, y la presentación de resultados. Esto es invaluable para la comprensión y comunicación del funcionamiento del modelo.

## 4. Resultados

En esta sección se realizará un análisis de funcionamiento del modelo y un estudio de las posibles mejoras en este. Además se discutirá la incertidumbre que rodea al problema que nos acecha.

### 4.1. Análisis inicial del modelo

En este apartado se realizará un análisis inicial de la implementación del modelo por parte de la librería LibRecommender. Para el análisis inicial utilizaremos 5894 usuarios y 3253 ítems del dataset MovieLens para su entrenamiento. Tras un entrenamiento de 100 épocas, las medidas cuantitativas sobre su eficacia son bastante prometedoras (ver figuras 4a y 4b).



(a) Gráfica de precisión y exactitud

(b) Gráfica del área bajo la curva ROC

Figura 4: Gráficas de medidas cuantitativas sobre la eficacia del modelo en 100 épocas

#### 4.1.1. Análisis cualitativo

Aunque las medidas cuantitativas son prometedoras, este tipo de modelos requieren de un análisis más exhaustivo para la validación de su funcionamiento. A continuación realizaremos un pequeño análisis cualitativo sobre una consulta sobre el modelo (ver tabla 2).

En cuanto a la consistencia de las recomendaciones teniendo en cuenta los géneros de película, podemos observar que: 6 son de Drama, 3 de Romance y 3 de Comedia lo que sugiere una preferencia de ese usuario por esos géneros. Otro comportamiento común en el modelo, es que recomiende películas populares aunque no se adapten a los gustos de un determinado usuario. En este caso, de las 7 películas recomendadas solo se produce esta casuística con *Star Wars: Episode IV*

k	Título	Género
1	American Beauty (1999)	Drama—Romance
2	Being John Malkovich (1999)	Comedy—Drama—Fantasy
3	Star Wars: Episode IV - A New Hope (1977)	Action—Adventure—Sci-Fi
4	Shakespeare in Love (1998)	Comedy—Drama—Romance
5	One Flew Over the Cuckoo's Nest (1975)	Drama
6	Casablanca (1942)	Drama—Romance
7	Fargo (1996)	Comedy—Crime—Drama—Thriller

Cuadro 2: Resultado de la consulta con los 7 ítems con mayor puntuación en su recomendación para el usuario con id 2211

## 4.2. Posibles optimizaciones

La elección del optimizador es un aspecto crucial en el entrenamiento de modelos de aprendizaje profundo, ya que puede influir significativamente en la convergencia del modelo y, en última instancia, en su rendimiento. En el contexto de mejorar la implementación de LightGCN, se exploraron varios optimizadores con el objetivo de identificar el que proporcionara los mejores resultados en términos de precisión y eficiencia. Los optimizadores probados fueron los disponibles en la siguiente documentación. [16]

Para cada optimizador, las gráficas de `recall_at_k_curves` proporcionan información sobre la capacidad del modelo para recuperar ítems relevantes en sus primeras  $k$  recomendaciones. Por otro lado, las gráficas de `training and validation loss curves` ofrecen insights sobre la convergencia del modelo y si había signos de sobre ajuste.

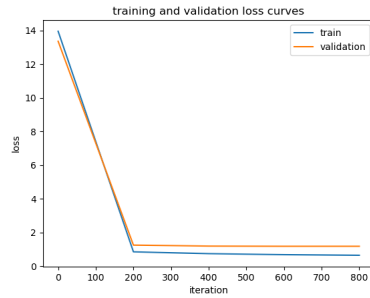
A continuación, se detallan los optimizadores probados y las métricas utilizadas para evaluar su eficacia:

### 4.2.1. Adam (Adaptive Moment Estimation)

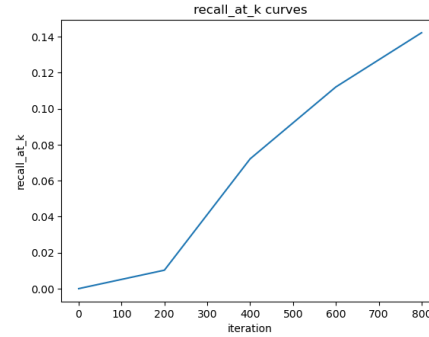
- **Configuración:**

```
optimizer = optim.Adam(model.parameters(), lr=LR, weight_decay=0.01)
```

- **Descripción:** Adam combina las ventajas de AdaGrad y RMSprop. Es conocido por ser eficiente y requerir poca memoria. El término `weight_decay` se utiliza para la regularización L2.
- **Resultados:** Se observaron las gráficas de `recall_at_k_curves` y `training and validation loss curves` para evaluar cómo Adam afecta la capacidad del modelo para hacer recomendaciones precisas y su convergencia durante el entrenamiento.



(a) Gráfica training and validation loss curves



(b) Gráfica recall\_at\_k\_curves

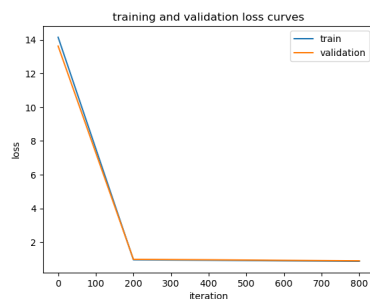
Figura 5: Gráficas obtenidas con Optimizador Adam y 1000 iteraciones

#### 4.2.2. SGD (Stochastic Gradient Descent) con Momentum

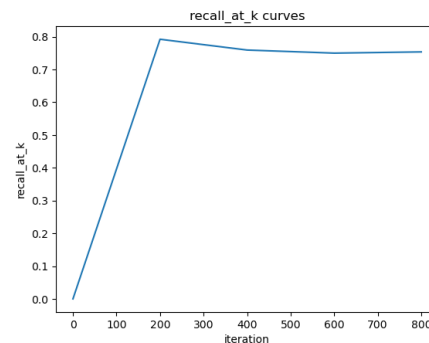
- **Configuración:**

```
optimizer = optim.SGD(model.parameters(), lr=0.01, momentum=0.9)
```

- **Descripción:** SGD es un optimizador tradicional. La adición de momentum ayuda a acelerar la convergencia y a superar mínimos locales.
- **Resultados:** Se analizaron las gráficas correspondientes para determinar si el momentum mejora la eficiencia del SGD en el contexto de LightGCN.



(a) Gráfica training and validation loss curves



(b) Gráfica recall\_at\_k\_curves

Figura 6: Gráficas obtenidas con Optimizador SDG y 1000 iteraciones

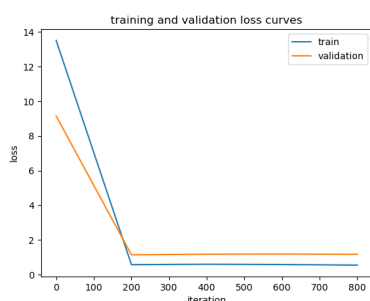
#### 4.2.3. RMSprop (Root Mean Square Propagation)

- **Configuración:**

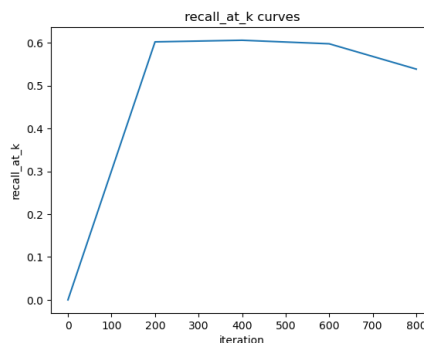


```
optimizer = optim.RMSprop(model.parameters(), lr=0.01, alpha=0.99)
```

- **Descripción:** RMSprop ajusta adaptativamente la tasa de aprendizaje durante el entrenamiento. Es especialmente útil en situaciones donde AdaGrad podría frenar demasiado la tasa de aprendizaje.
- **Resultados:** Las gráficas proporcionaron insights sobre la rapidez y estabilidad de la convergencia con RMSprop.



(a) Gráfica training and validation loss curves



(b) Gráfica recall\_at\_k curves

Figura 7: Gráficas obtenidas con Optimizador RMSprop y 1000 iteraciones

#### 4.2.4. Adadelata

- **Configuración:**

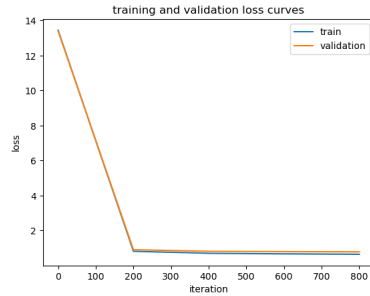
```
optimizer = optim.Adadelata(model.parameters(), lr=1.0, rho=0.9)
```

- **Descripción:** Adadelata es una extensión de AdaGrad que busca reducir su agresiva disminución de la tasa de aprendizaje.
- **Resultados:** Se evaluó si Adadelata ofrece una convergencia más estable y resultados superiores en comparación con AdaGrad.

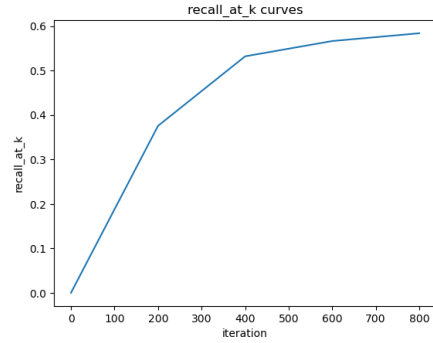
#### 4.2.5. Adagrad (Adaptive Gradient Algorithm)

- **Configuración:**

```
optimizer = optim.Adagrad(model.parameters(), lr=0.01)
```



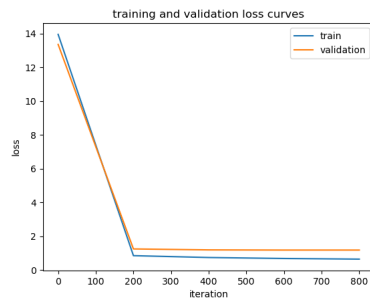
(a) Gráfica training and validation loss curves



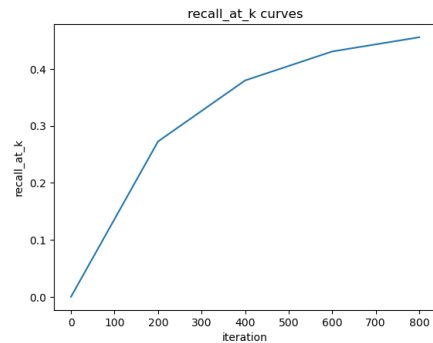
(b) Gráfica recall\_at\_k\_curves

Figura 8: Gráficas obtenidas con Optimizador Adadelata y 1000 iteraciones

- **Descripción:** Adagrad ajusta la tasa de aprendizaje de cada parámetro basándose en la historia de gradientes.
- **Resultados:** Aunque Adagrad puede ser beneficioso en ciertos escenarios, se evaluó su comportamiento específicamente con LightGCN a través de las gráficas mencionadas.



(a) Gráfica training and validation loss curves



(b) Gráfica recall\_at\_k\_curves

Figura 9: Gráficas obtenidas con Optimizador Adagrad y 1000 iteraciones

#### 4.2.6. Conclusiones

Adam, como optimizador, combina las ventajas de Momentum y RMSprop, ofreciendo una convergencia rápida y estable. Ajusta adaptativamente la tasa de aprendizaje para cada parámetro, lo que es crucial para modelos como LightGCN con diferentes embeddings y parámetros. Su robustez frente a gradientes esparcidos o ruidosos, eficiencia computacional y regularización L2 incorporada lo hacen ideal para sistemas de recomendación. Además, su flexibilidad y resultados empíricos positivos en la práctica lo han consolidado como una opción preferida en el aprendizaje profundo, incluyendo para LightGCN. Sin embargo, es vital validar su eficacia específicamente para cada problema y conjunto de datos. [17]

### 4.3. Análisis de la incertidumbre

Muchos sistemas basados en datos, existe una inherente incertidumbre en las predicciones generadas. A continuación, se analiza esta incertidumbre en el contexto del filtrado colaborativo.

#### 4.3.1. Naturaleza Probabilística de las Puntuaciones

El objetivo principal es asignar puntuaciones a posibles recomendaciones para un usuario. Estas puntuaciones, en esencia, representan la probabilidad o confianza de que un usuario interactuará o preferirá un ítem en particular. Sin embargo, dado que se basan en datos históricos y patrones observados, siempre hay un grado de incertidumbre asociado con estas puntuaciones.

Al seleccionar los  $k$  ítems con la puntuación más alta, estamos haciendo una suposición basada en la información disponible, pero no hay garantía de que estos ítems sean efectivamente los más relevantes para el usuario en un momento dado.

#### 4.3.2. Grafos Bipartitos y Estructuras de Alto Orden

Representar las interacciones entre usuarios e ítems como un grafo bipartito introduce una estructura que puede ser explotada para mejorar las recomendaciones. Sin embargo, esta estructura también introduce complejidad y potencial incertidumbre.

Las técnicas avanzadas, como la propagación de etiquetas o la factorización de matrices en grafos, buscan aprovechar las relaciones de alto orden en estos grafos. Si bien estas técnicas pueden capturar patrones más sutiles y complejos, también pueden ser más susceptibles a ruido o anomalías en los datos, lo que puede aumentar la incertidumbre.

#### 4.3.3. Ranking Basado en Puntuaciones

Las puntuaciones se utilizan para establecer un ranking de ítems para cada usuario. Sin embargo, el ranking en sí no proporciona una medida absoluta de relevancia, sino una medida relativa. Dos ítems con puntuaciones muy cercanas en el ranking pueden ser casi igualmente relevantes para el usuario, pero pequeñas variaciones en los datos o en el modelo pueden alterar su posición relativa, introduciendo incertidumbre en las recomendaciones.

#### 4.3.4. Incertidumbre Fundamental

En última instancia, el acto de recomendar ítems se basa en la suposición de que las interacciones pasadas de un usuario son indicativas de sus preferencias futuras. Sin embargo, las preferencias de los usuarios pueden cambiar con el tiempo y pueden ser influenciadas por una multitud de factores externos. Esta es una fuente fundamental de incertidumbre que ningún modelo puede eliminar completamente.

## 5. Aplicaciones

El filtrado colaborativo y los sistemas de recomendación, en general, han encontrado aplicaciones en una amplia variedad de industrias y contextos. Estos sistemas no solo están reservados para grandes corporaciones con vastos recursos computacionales, sino que también pueden ser implementados por pequeños negocios para mejorar la experiencia del cliente y aumentar las ventas. A continuación, se describen algunas aplicaciones prácticas de estos sistemas:

### Pequeños Cines

- **Recomendación Personalizada:** Basándose en las películas que un cliente ha visto anteriormente, el sistema podría recomendar próximos estrenos o películas en cartelera que se alineen con sus gustos.
- **Ofertas y Descuentos:** Si un cliente ha comprado entradas para una película de un género específico, el cine podría ofrecer descuentos para futuras películas del mismo género o películas con actores o directores similares.
- **Eventos Temáticos:** Basándose en las preferencias generales de su clientela, el cine podría organizar noches temáticas o maratones de películas populares dentro de ciertos géneros o franquicias.

### Pequeñas Librerías

- **Recomendación de Libros:** Al comprar un libro, el sistema podría recomendar otros libros del mismo autor, del mismo género o con temáticas relacionadas.
- **Promociones Cruzadas:** Si un cliente compra un libro que forma parte de una serie o trilogía, la librería podría ofrecer un descuento en los siguientes libros de la serie.
- **Eventos de Autor:** Basándose en los libros más vendidos o populares, la librería podría organizar firmas de libros o charlas con autores que sean de interés para su clientela.

### Pequeños Negocios con Sistemas de Punto de Venta (POS)

- **Recomendaciones en Tiempo Real:** Al utilizar un sistema POS, el negocio podría generar recomendaciones en tiempo real durante el proceso de compra, sugiriendo productos complementarios o relacionados.
- **Programas de Lealtad:** Basándose en las compras pasadas de un cliente, el negocio podría ofrecer descuentos personalizados o recompensas que se alineen con las preferencias del cliente.
- **Inventario y Stock:** Al analizar las tendencias de compra y las recomendaciones más populares, el negocio podría ajustar su inventario para asegurarse de tener en stock los productos más demandados.

Estas aplicaciones demuestran que, con la tecnología adecuada y una implementación cuidadosa, incluso los pequeños negocios pueden aprovechar el poder de los sistemas de recomendación para mejorar la experiencia del cliente y aumentar las ventas. Es esencial, sin embargo, que estos sistemas se adapten a las necesidades y características específicas de cada negocio para maximizar su eficacia.

## 6. Presupuesto para la Implementación de RecGraf

Este presupuesto proporciona una visión general de las fases y tareas involucradas en la implementación de RecGraf en un sistema cliente. Es esencial trabajar estrechamente con el cliente en cada fase para asegurar que el sistema cumpla con sus necesidades y expectativas.

### 1. Recolección de Datos del Cliente:

- **Descripción:** Esta fase implica la recopilación, limpieza y preprocesamiento de los datos del cliente. Es esencial obtener datos de calidad para entrenar un modelo efectivo.
- **Tareas:**
  - Reuniones con el cliente para entender la estructura y naturaleza de los datos.
  - Extracción de datos desde las fuentes proporcionadas por el cliente.
  - Limpieza y preprocesamiento de datos para eliminar valores faltantes, outliers o errores.
  - Análisis exploratorio para entender las características y distribuciones de los datos.
- **Duración:** 2 semanas
- **Costo:** 5,000 €

### 2. Entrenamiento del Modelo:

- **Descripción:** Una vez que los datos estén listos, se procederá al entrenamiento del modelo usando LibRecommender.
- **Tareas:**
  - Selección y configuración de la arquitectura del modelo.
  - División de datos en conjuntos de entrenamiento, validación y prueba.
  - Entrenamiento del modelo y ajuste de hiperparámetros.
  - Evaluación del modelo en el conjunto de validación y ajustes según sea necesario.
- **Duración:** 4 semanas
- **Costo:** 8,000€

### 3. Testeo del Modelo Entrenado:

- **Descripción:** Antes de la implementación en producción, es crucial testear el modelo en un conjunto de datos no visto para evaluar su rendimiento.
- **Tareas:**
  - Evaluación del modelo en el conjunto de prueba.
  - Comparación de los resultados con métricas de referencia o benchmarks.

- Ajustes finales basados en el rendimiento en el conjunto de prueba.
- **Duración:** 2 semanas
- **Costo:** 3,000€

#### 4. Desarrollo de la Aplicación de Observabilidad:

- **Descripción:** Esta fase implica el desarrollo de un frontend que permita a los stakeholders observar y entender el rendimiento del modelo en tiempo real.
- **Tareas:**
  - Diseño de la interfaz de usuario (UI) para la aplicación.
  - Desarrollo del frontend y conexión con la API del modelo.
  - Implementación de gráficos y métricas para monitorear el rendimiento del modelo.
  - Testeo y ajustes basados en el feedback de los stakeholders.
- **Duración:** 6 semanas
- **Costo:** 10,000€

#### 5. Integración en el Sistema del Cliente:

- **Descripción:** Una vez que todo esté listo, el modelo y la aplicación de observabilidad se integrarán en el sistema existente del cliente.
- **Tareas:**
  - Integración de la API del modelo con los sistemas del cliente.
  - Configuración y despliegue del modelo en el entorno de producción del cliente.
  - Pruebas finales para asegurar que todo funcione correctamente en el entorno del cliente.
  - Documentación y formación para el equipo del cliente sobre cómo usar y mantener el sistema.
- **Duración:** 3 semanas
- **Costo:** 6,000€

**Costo Total Estimado:** 32,000€

**Duración Total Estimada:** 17 semanas

Estas cifras son aproximadas y se basan en un proyecto de complejidad media. Es crucial obtener cotizaciones detalladas y específicas de los profesionales o empresas que llevarán a cabo el trabajo para tener una estimación más precisa. Además, es recomendable considerar un margen adicional en el presupuesto y el tiempo para imprevistos o ajustes adicionales.

## 7. Conclusiones

Este estudio ha abordado de manera exhaustiva la relevancia y aplicabilidad de los sistemas de recomendación en el ámbito de las pequeñas y medianas empresas. A través de la implementación del Filtrado Colaborativo, se ha demostrado que es posible anticipar y descubrir las preferencias de los usuarios sin depender de extensas descripciones de productos. Esta metodología no solo es innovadora, sino también adaptativa, respondiendo eficientemente a los cambios en comportamientos y gustos de los consumidores. Un aspecto crucial que se destaca en el estudio es la incertidumbre inherente en las recomendaciones, reconociendo que las interacciones pasadas de un usuario no siempre son indicativas de sus preferencias futuras. Sin embargo, a pesar de esta incertidumbre fundamental, el estudio resalta la importancia de una fase de testeo rigurosa antes de la implementación en un entorno real, garantizando así la eficacia y precisión del modelo. En un mercado en constante evolución, donde la personalización y la adaptabilidad son clave, las herramientas y enfoques presentados en este estudio ofrecen a las [PYMES](#) una ventaja competitiva, permitiéndoles ofrecer a sus clientes experiencias más personalizadas y, por ende, más satisfactorias.



## Referencias

- [1] J. L. Duarte Castillo, *Factores determinantes y críticos en empresas de servicios, para la obtención de ventajas competitivas sostenibles y transferibles a estrategias de globalización: un análisis de la industria del software...* Universitat Autònoma de Barcelona,, 2005. [Online]. Available: <https://www.tdx.cat/bitstream/handle/10803/3957/jldc1de1.pdf>
- [2] S. P. de Pablos, “Las pymes, atrapadas en el laberinto que conduce a la digitalización,” *El País*, Apr. 2023. [Online]. Available: [https://elpais.com/branded\\_content/2023-04-28/las-pymes-atrapadas-en-el-laberinto-que-conduce-a-la-digitalizacion.html](https://elpais.com/branded_content/2023-04-28/las-pymes-atrapadas-en-el-laberinto-que-conduce-a-la-digitalizacion.html)
- [3] L. V. Hervás, “¿Cómo son las pymes en España?” *Cinco Días*, Jun. 2022. [Online]. Available: [https://cincodias.elpais.com/cincodias/2022/06/24/pyme/1656070303\\_778210.html](https://cincodias.elpais.com/cincodias/2022/06/24/pyme/1656070303_778210.html)
- [4] Redacción, “El auge del eCommerce dispara la competencia - Capital,” *Capital*, Apr. 2022. [Online]. Available: <https://capital.es/2022/03/17/e-commerce-competencia-empresas>
- [5] G. Mileva, “A Helpful Overview on eCommerce Recommendation Systems,” *Influencer Marketing Hub*, Apr. 2023. [Online]. Available: <https://influencermarketinghub.com/e-commerce-recommendation-system>
- [6] A. M. del Campo Fernández-Paniagua, “Filtrado colaborativo: ¿Qué es y para qué sirve?” *Thinking for Innovation*, Jul. 2018. [Online]. Available: <https://www.iebschool.com/blog/filtrado-colaborativo-sirve-e-commerce>
- [7] “Best Shopify App for Upsell Cross Sell & AI Related... | Tienda de aplicaciones de Shopify,” Oct. 2023, [Online; accessed 7. Oct. 2023]. [Online]. Available: <https://apps.shopify.com/recommended-products-wiser?locale=es>
- [8] “Premium Recommendation Engine | Crossing Minds,” Oct. 2023, [Online; accessed 7. Oct. 2023]. [Online]. Available: <https://www.crossingminds.com>
- [9] “Experience OS by Dynamic Yield,” Sep. 2023, [Online; accessed 7. Oct. 2023]. [Online]. Available: <https://www.dynamicyield.com/experience-os>
- [10] “Recommendify - Advanced E-Commerce Personalization for Shopify,” Oct. 2023, [Online; accessed 7. Oct. 2023]. [Online]. Available: <https://recommendifyapp.com>
- [11] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, and M. Wang, “Lightgcn: Simplifying and powering graph convolution network for recommendation,” in *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, 2020, pp. 639–648. [Online]. Available: <https://arxiv.org/pdf/2002.02126v4.pdf>

- [12] X. Wang, X. He, M. Wang, F. Feng, and T.-S. Chua, “Neural graph collaborative filtering,” in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, jul 2019. [Online]. Available: <https://doi.org/10.1145%2F3331184.3331267>
- [13] Contributors to Wikimedia projects, “MovieLens - Wikipedia,” Apr. 2023, [Online; accessed 29. Oct. 2023]. [Online]. Available: <https://en.wikipedia.org/w/index.php?title=MovieLens&oldid=1151179828>
- [14] J. M. Jianmo Ni, Jiacheng Li, “Justifying Recommendations using Distantly-Labeled Reviews and Fine-Grained Aspects,” Oct. 2019, [Online; accessed 29. Oct. 2023]. [Online]. Available: <https://cseweb.ucsd.edu/~jmcauley/pdfs/emnlp19a.pdf>
- [15] “LibRecommender,” Oct. 2023, [Online; accessed 29. Oct. 2023]. [Online]. Available: <https://pypi.org/project/LibRecommender/0.0.1>
- [16] “torch.optim — PyTorch 2.1 documentation,” Oct. 2023, [Online; accessed 30. Oct. 2023]. [Online]. Available: <https://pytorch.org/docs/stable/optim.html>
- [17] “Adam — PyTorch 2.1 documentation,” Oct. 2023, [Online; accessed 30. Oct. 2023]. [Online]. Available: <https://pytorch.org/docs/stable/generated/torch.optim.Adam.html#torch.optim.Adam>

## Siglas

**GCN** Graph Convolutional Network. [7](#), [12](#)

**GNN** Graph Neural Networks. [3](#)

**LGC** Light Graph Convolution. [8](#), [9](#)

**NGCF** Neural Graph Collaborative Filtering. [7](#)

**PIB** Producto Interior Bruto. [2](#)

**PYMES** Pequeñas y medianas empresas. [2](#), [3](#), [5](#), [6](#), [23](#)