



**Desarrollo y Evaluación de Sistemas de Recomendación con Análisis de Sentimientos para  
la Optimización de Ventas en el Mercado en Línea: Un Enfoque Analítico de Datos**

Lina María Beltrán Durango

Mario Earles Otero Andrade

Monografía presentada para optar al título de Especialista en Analítica y Ciencia de Datos

Asesor

Carlos Alberto Álvarez Henao, DSc en Ing. Civil

Universidad de Antioquia

Facultad de Ingeniería

Especialización en Analítica y Ciencia de Datos

Medellín, Antioquia, Colombia

2023

Cita	(Beltrán Durango & Otero Andrade, 2023)
Referencia	Beltrán Durango, L. M., & Otero Andrade, M. E. (2023). <i>Desarrollo y Evaluación de Sistemas de Recomendación con Análisis de Sentimientos para la Optimización de Ventas en el Mercado Online: Un Enfoque Analítico de Datos</i> . Universidad de Antioquia, Medellín, Colombia.
Estilo APA 7 (2020)	



Especialización en Analítica y Ciencia de Datos, Cohorte V.

Centro de Investigación Ambientales y de Ingeniería (CIA).



Centro de Documentación Ingeniería (CENDOI)

**Repositorio Institucional:** <http://bibliotecadigital.udea.edu.co>

Universidad de Antioquia - [www.udea.edu.co](http://www.udea.edu.co)

Rector: John Jairo Arboleda Céspedes.

Decano: Julio Cesar Saldarriaga Molina

Jefe departamento: Diego José Luis Botia Valderrama

El contenido de esta obra corresponde al derecho de expresión de los autores y no compromete el pensamiento institucional de la Universidad de Antioquia ni desata su responsabilidad frente a terceros. Los autores asumen la responsabilidad por los derechos de autor y conexos.

### **Dedicatoria**

A mi madre Gloria Durango, fuente inagotable de inspiración y apoyo incondicional. Agradezco su amor, paciencia y sabiduría, que han sido mi soporte a lo largo de este viaje académico. A mis amigos, cuya amistad ha hecho más ligero el camino y ha llenado de alegría cada paso. A mis profesores, por su dedicación y enseñanzas que han enriquecido mi conocimiento. Este trabajo está dedicado a todos aquellos que han contribuido, de una manera u otra, a mi crecimiento personal y académico.

### **Agradecimientos**

Quisiera expresar mi sincero agradecimiento a Carlos Alberto Álvarez Henao, por su orientación experta, paciencia y apoyo constante a lo largo de la elaboración de esta monografía. Sus comentarios y sugerencias han sido invaluable para mejorar la calidad de este trabajo.

*Lina M. Beltrán D.*

### **Dedicatoria**

Esta significativa conquista en mi trayectoria se la dedico con todo mi corazón a mi padre, Mariano Earles Otero Benedetty, a mi madre, Maria Victoria Andrade Niño, y a mi hermano, Victor Manuel Otero Andrade. Mi gratitud es infinita por su amor, su respaldo sin condiciones, su manera especial de impulsarme a avanzar a pesar de las dificultades y por todas las lecciones de vida que me han brindado. Su contribución a este éxito es inmensurable.

A Jelitza Castillo, quien ha sido la esencia de lo extraordinario en mi vida. Mi más profundo agradecimiento por todo el amor y soporte que me has ofrecido, por darme la fortaleza para perseverar en los momentos más desafiantes y por ser una brillante luz en mis momentos más oscuros.

### **Agradecimientos**

Extiendo mi más sincera gratitud a Carlos Alberto Álvarez Henao, cuya guía experta, paciencia inagotable y apoyo incansable han sido fundamentales durante la elaboración de esta monografía. Su sabiduría y constructivas sugerencias han sido esenciales para elevar la calidad de este trabajo.

*Mario Earles Otero Andrade*

---

## Tabla de contenido

<b>Resumen.....</b>	<b>12</b>
<b>Abstract.....</b>	<b>14</b>
<b>1. Descripción del problema.....</b>	<b>16</b>
1.1. Problema de negocio.....	17
1.2. Aproximación desde la analítica de datos.....	18
1.3. Origen de los datos.....	19
1.4. Métricas de desempeño.....	19
<b>2. Objetivos.....</b>	<b>25</b>
2.1. Objetivo general.....	25
2.2. Objetivos específicos.....	26
<b>3. Datos.....</b>	<b>28</b>
3.1. Datos originales.....	28
3.2. Datasets.....	32
3.3. Analítica descriptiva.....	35
<b>4. Proceso de analítica.....</b>	<b>41</b>
4.1. Pipeline principal.....	41
4.2. Modelos.....	46
4.3. Métricas.....	48
<b>5. Metodología.....</b>	<b>51</b>
5.1. Baseline.....	51
5.2. Validación.....	54
5.3. Iteraciones y evolución.....	56
5.4. Herramientas de desarrollo.....	59
<b>6. Resultados y discusión.....</b>	<b>61</b>
6.1. Métricas.....	67
6.2. Evaluación cualitativa.....	75
6.3. Consideraciones de producción.....	77
<b>7. Conclusiones.....</b>	<b>79</b>
<b>8. Recomendaciones.....</b>	<b>81</b>
<b>Referencias.....</b>	<b>83</b>

---

## Lista de tablas

Tabla 1: Resultados de la iteración 1 para el modelo Filtrado Colaborativo usando Embedding con Keras	52
Tabla 2: Resultados SVD.....	68
Tabla 3: Resultados de modelos con Embeddings.....	69
Tabla 4: Resultados de modelos de Redes Neuronales Multicapa.....	70
Tabla 5: Resultados de modelos con Embeddings y Redes Neuronales Multicapa.....	71

## Lista de figuras

Figura 1: Representación de la estructura de un registro del conjunto de evaluaciones - Imagen tomada de “Amazon review data”.....	27
Figura 2: Representación de la estructura de un registro del conjunto de metadatos - Imagen tomada de “Amazon review data”.....	29
Figura 3: Captura de la lista de archivos con datos crudos en el repositorio de los autores de este documento en Google Drive.....	31
Figura 4: Captura de la lista de archivos con datos procesados en el repositorio de los autores de este documento en Google Drive.....	32
Figura 5: Información de la estructura del dataset procesado DataFinal_Amazon.....	32
Figura 6: Lista de pasos ejecutados en la etapa de análisis descriptivo.....	34
Figura 7: Resultados estadísticas descriptivas para columnas numéricas “Overall”, “Price” y “extracted_rank”.....	35
Figura 8: Distribución de los valores de la característica “Overall” - escala 116.....	35
Figura 9: Distribución de las categorías de productos en el conjunto de datos - escala 116.....	36
Figura 10: Matriz de correlación de las características numéricas del conjunto de datos.....	37
Figura 11: Gráfico de evolución de las revisiones a lo largo del periodo comprendido entre 2004 y 2018.....	38
Figura 12: Gráfico de la tendencia mensual de revisiones a lo largo del periodo comprendido entre 2004 y 2018.....	38
Figura 13: Arquitectura del flujo de datos empleado - Original de Maggie Mhanna: MLOps Best Practices for Data Scientist".....	39
Figura 14: Métricas de desempeño aplicadas al modelo SVD.....	46
Figura 15: Métricas de desempeño aplicadas al modelo de Filtrado Colaborativo con Embedding mediante Keras.....	47
Figura 16: Métricas de desempeño aplicadas al modelo de Filtrado Colaborativo con Embedding y Red Neuronal Multicapa mediante Keras.....	48
Figura 17: Entrenamiento del modelo “filtrado colaborativo con redes neuronales y Embeddings con Keras”.....	60
Figura 18: Resultados del modelo “filtrado colaborativo con redes neuronales y Embeddings con Keras”.....	61
Figura 19: Resultados análisis de sentimiento por producto, escala representativa numérica de -1 a 1 (Negativo, neutro, positivo).....	62
Figura 20: Clústeres de productos basados en Sentimientos.....	62
Figura 21: Resumen de resultados de las métricas de error para la iteración 5.....	66
Figura 22: Visualización gráfica de los resultados de las métricas de error para la iteración 5.....	66
Figura 23: Variación del Error Cuadrático Medio (MSE) por cantidad de épocas en relación con la iteración.....	71
Figura 24: Comparación del valor promedio de la Raíz del Error Cuadrático Medio (RMSE) en	

---

cada iteración.....	71
Figura 25: Comparación del valor promedio del Porcentaje Absoluto Medio de Error (MAPE) en cada iteración.....	71
Figura 26: Dispersión de los valores de MAE en cada iteración y cantidad de épocas.....	72
Figura 27: Comparación del valor promedio del Porcentaje Absoluto Medio de Error (MAPE) en cada iteración y tamaño de época.....	72
Figura 28: Recomendaciones generadas con el modelo embeddings batch=2048.....	75
Figura 29: Recomendaciones generadas con el modelo embeddings batch=800000.....	75
Figura 30: Recomendaciones generadas con el modelo embeddings y Red Neuronal Multicapa.....	75
Figura 31: Recomendaciones generadas con el modelo embeddings y Red Neuronal Multicapa batch=800000.....	75
Figura 32: Recomendaciones generadas con el modelo de Redes Neuronales Multicapa batch=80000.....	76
Figura 33: Recomendaciones generadas con el modelo de Redes Neuronales Multicapa batch=800000....	76
Figura 34: Recomendaciones generadas con el modelo SVD.....	76
Figura 35: Recomendaciones generadas con el modelo SVD batch=4000.....	76



### **Siglas, acrónimos y abreviaturas**

<b>RMSD</b>	Root-Mean-Square Deviation
<b>MAE</b>	Mean Absolute Error
<b>EMNLP</b>	Empirical Methods in Natural Language Processing
<b>Colab.</b>	Google Colaboratory notebooks
<b>ML</b>	Machine Learning
<b>ROI</b>	Return On Investment
<b>UdeA</b>	Universidad de Antioquia
<b>SVD</b>	Singular Value Decomposition

## Resumen

Este proyecto de monografía se enfoca en el desarrollo y evaluación de sistemas de recomendación con análisis de sentimientos para entender e identificar los patrones de comportamiento e interacción entre los usuarios y los productos en el mercado en línea, utilizando datos de reseñas de Amazon hasta el año 2018. Nuestra investigación tiene como objetivo abordar preguntas clave, utilizando datos de reseñas para mejorar las recomendaciones de productos, la aplicación del análisis de sentimientos para comprender las preferencias de los usuarios, evaluar la efectividad de diversos modelos de sistemas de recomendación y medir su rendimiento.

Hemos empleado una variedad de enfoques, que incluyen *Collaborative Filtering Systems*, *Content-Based Systems*, *Hybrid Recommendation Systems* y *Empirical Methods in Natural Language Processing*. La evaluación se realizó con las siguientes métricas MSE, RMSE, MAE, MAPE y k-fold cross-validation

Durante el proceso, enfrentamos desafíos relacionados con la calidad y volumen de los datos, incluida la capacidad de procesamiento en ambientes de desarrollo experimentales como Google Colaboratory, así como la selección adecuada de modelos. Sin embargo, superamos estos obstáculos mediante técnicas de preprocesamiento y ajuste de hiperparámetros.

Los resultados revelaron mejoras en la precisión de las recomendaciones y la comprensión de las preferencias de los usuarios. Los modelos híbridos y los enfoques basados en contenido mostraron un rendimiento particularmente prometedor. Esta investigación presenta un enfoque analítico de datos para respaldar el entendimiento de las interacciones usuario - producto en mercados en línea y así aprovechar las propiedades del análisis de sentimientos y sistemas de recomendación. Nuestros resultados respaldan la utilidad de estos enfoques y abren oportunidades para aplicaciones en el comercio electrónico y más allá.

**Repositorios GitHub:**

- <https://github.com/lmbd92/DataScienceMonograph>
- <https://github.com/marioeoteroa/DataScienceMonograph>

*Palabras clave:* Ventas Minoristas, en Línea, Comercio Electrónico, Aprendizaje Automático, Sistemas de Recomendación, Métodos Empíricos en Procesamiento de Lenguaje Natural.

## **Abstract**

This monograph project focuses on the development and evaluation of sentiment analysis-based recommendation systems to understand and identify patterns of behavior and interaction between users and products in the e-commerce, using Amazon review data up to the year 2018. Our research aims to address key questions, utilizing review data to enhance product recommendations, applying sentiment analysis to understand user preferences, evaluating the effectiveness of various recommendation system models, and measuring their performance.

We have employed a variety of approaches, including Collaborative Filtering Systems, Content-Based Systems, Hybrid Recommendation Systems, and Empirical Methods in Natural Language Processing. Evaluation was conducted using the following metrics: MSE, RMSE, MAE, MAPE y k-fold cross-validation.

Throughout the process, we faced challenges related to the quality and volume of data, including processing capacity in experimental development environments such as Google Colaboratory, as well as the appropriate selection of models. However, we overcame these obstacles through preprocessing techniques and hyperparameter tuning.

The results revealed improvements in the accuracy of recommendations and understanding user preferences. Hybrid models and content-based approaches showed particularly promising performance. This research presents a data analytical approach to support the understanding of user-product interactions in online markets, leveraging the properties of sentiment analysis and recommendation systems. Our findings support the utility of these approaches and open opportunities for applications in e-commerce and beyond.

**GitHub Repositories:**

- <https://github.com/lmbd92/DataScienceMonograph>
- <https://github.com/marioeoteroa/DataScienceMonograph>

Keywords: Retail Sales, Online, E-commerce, Machine Learning, Recommendation Systems, Empirical Methods in Natural Language Processing.

## 1. Descripción del problema

La presente investigación se enfoca en el desarrollo y evaluación de sistemas de recomendación con análisis de sentimientos para entender e identificar los patrones de comportamiento e interacción entre los usuarios y los productos en el mercado en línea. Este enfoque analítico de datos se basa en la utilización de datos de reseñas de Amazon correspondientes al periodo comprendido entre mayo de 1996 y octubre de 2018.

El problema que abordamos en esta monografía se centra en ofrecer a la industria e-commerce un respaldo a sus estrategias comerciales basado en técnicas analíticas y de aprendizaje automático, aprovechando la información contenida en las reseñas realizadas por compradores en línea y las interacciones entre clientes y productos. Para ello, planteamos las siguientes preguntas de investigación:

1. ¿Cómo se pueden utilizar los datos de reseñas de Amazon para mejorar las recomendaciones de productos a los usuarios del comercio electrónico?
2. ¿Cómo se puede aprovechar el análisis de sentimientos para comprender mejor las preferencias y opiniones de los usuarios?
3. ¿Cuál es la efectividad de diferentes modelos de sistemas de recomendación para identificar patrones de compra e interacción cliente - producto en entornos de comercio electrónico?
4. ¿Cuál es la mejor manera de evaluar el rendimiento de los modelos propuestos?

Para abordar estas preguntas, emplearemos una variedad de enfoques de sistemas de recomendación, que incluyen *Collaborative Filtering Systems*, *Content-Based Systems*, *Hybrid Recommendation Systems* y *Empirical Methods in Natural Language Processing*. Además, emplearemos métricas de evaluación como MSE, RMSE, MAE, MAPE y k-fold cross-validation para medir el rendimiento de estos modelos.

En resumen, esta investigación se enfoca en entender e identificar los patrones de comportamiento e interacción entre los usuarios y los productos mediante la aplicación de técnicas de análisis de sentimientos y sistemas de recomendación. A través de la exploración de estos métodos, buscamos proporcionar respuestas concretas y soluciones efectivas para el problema de negocio planteado a continuación.

### **1.1. Problema de negocio**

Este proyecto se enmarca en el contexto del comercio electrónico, donde las empresas buscan constantemente formas de mejorar sus estrategias de ventas en línea. En particular, se aborda el desafío de optimizar las ventas en un mercado con alta variabilidad e incertidumbre, utilizando como fuente de datos reseñas de productos proporcionados por la plataforma Amazon durante un periodo aproximado de 22 años<sup>1</sup>.

El problema de negocio se presenta de la siguiente manera: En el entorno altamente competitivo del comercio electrónico, las marcas y principalmente las plataformas de ventas minoristas (retail) enfrentan el desafío de ofrecer a los usuarios recomendaciones de productos precisas y personalizadas. La calidad de estas recomendaciones tiene un impacto directo en la retención de clientes y, en última instancia, en las ventas. Para afrontar este desafío, es fundamental comprender a fondo las preferencias, interacciones, comportamiento de compra y opiniones de los usuarios. Esto se puede lograr mediante métodos de filtro colaborativo, basados en contenido e interacciones. Además, como objetivo central de nuestro proyecto, se busca fortalecer estos sistemas mediante el análisis de sentimientos de las reseñas de productos.

Una pregunta importante que surge luego de identificar el problema descrito es ¿Cómo mejorar la efectividad de los sistemas de recomendación en línea?, teniendo en cuenta la información contenida en las reseñas, las interacciones entre usuarios y productos, y las emociones expresadas por los usuarios. Este problema de negocio es crucial, ya que una recomendación precisa no solo puede aumentar las ventas, sino también mejorar la experiencia del usuario y fortalecer la fidelidad hacia la plataforma de comercio electrónico.

---

<sup>1</sup> Amazon Review Data ([más información](#))

Por lo tanto, este proyecto aborda el desafío empresarial de entender e identificar patrones de compra e interacción en un contexto de comercio electrónico altamente competitivo, aprovechando el análisis de sentimientos y los sistemas de recomendación para ofrecer a los usuarios recomendaciones más precisas y relevantes.

## 1.2. Aproximación desde la analítica de datos

La aproximación desde la analítica de datos en este proyecto se basa en la creación y evaluación de modelos predictivos para la recomendación de productos a clientes en un entorno de comercio electrónico. Estos modelos tienen como objetivo mejorar la eficiencia y la precisión de las recomendaciones, lo que a su vez puede aumentar las ventas y la satisfacción del cliente. A continuación, se describen las áreas clave de enfoque y se proporcionan referencias a artículos fundamentales de aprendizaje automático que respaldan la utilidad de los modelos predictivos desarrollados:

**Modelos de Recomendación Personalizados:** Se implementarán modelos de recomendación personalizados que aprovecharán el comportamiento previo de los usuarios y el análisis de sentimientos de las reseñas de productos. Esto se alinea con la idea de ofrecer recomendaciones más relevantes y adaptadas a las preferencias individuales de los clientes. Un artículo clave en esta área es "Item-based collaborative filtering recommendation algorithms" (Sarwar et al., 2001), que aborda la importancia de los sistemas de filtrado colaborativo.

**Análisis de Sentimientos:** El análisis de sentimientos se aplicará a las reseñas de productos para comprender las opiniones y emociones de los usuarios. Esto permitirá una comprensión más profunda de las preferencias del cliente y se basará en estudios como "Mining and summarizing customer reviews" (Hu & Liu, 2004), que detalla técnicas para el análisis de reseñas.

**Evaluación de Modelos:** La evaluación de modelos se llevará a cabo utilizando métricas como MSE, RMSE, MAE, MAPE y k-fold cross-validation para medir la precisión y el rendimiento de los sistemas de recomendación. El artículo "Evaluating recommendation systems"



(Herlocker et al., 2004) proporciona una base sólida para la evaluación de sistemas de recomendación.

**Enfoques Híbridos:** Se explorarán enfoques híbridos que combinan sistemas de filtrado colaborativo y basados en contenido para aprovechar las fortalezas de ambos. El artículo "Hybrid Recommender Systems: Survey and Experiments" (Burke, 2002) ofrece una visión completa de estos enfoques.

### 1.3. Origen de los datos

Amazon reviews<sup>2</sup> es un conjunto de datos que contiene reseñas de productos y metadatos de Amazon, incluidas 142,8 millones de reseñas que abarcan desde mayo de 1996 hasta octubre de 2018. Este conjunto de datos incluye reseñas (calificaciones, texto, votos de utilidad), metadatos de productos (descripciones, información de categoría, precio, marca y características de imagen) y enlaces (también gráficos vistos/comprados).

Amazon Reviews es un conjunto de datos para abordar la tarea de identificar si el sentimiento de una reseña de producto es positivo, neutro o negativo. Este conjunto de datos incluye reseñas de varias categorías de productos diferentes, tales como: libros (B) (2834 muestras), DVD (D) (1199 muestras), electrónica (E) (1883 muestras), artículos de cocina y para el hogar (K) (1755 muestras) y moda (883,636 muestras), entre otros, para nuestro caso de investigación, hemos decidido tomar las siguientes categorías: “Moda”, “Lujo y belleza” y “Ropa, zapatos y joyas”.

### 1.4. Métricas de desempeño

En este trabajo, utilizaremos varias métricas tanto de aprendizaje automático como de negocio para evaluar el desempeño de los modelos de recomendación desarrollados y su impacto en el proceso de negocio de ventas en línea. Es importante mencionar que, debido a restricciones de tiempo y la falta de acceso a un escenario de simulación de negocio o un cliente real, las métricas de negocio se proponen de manera académica y como ideales para evaluar en un supuesto caso de implementación. No obstante, sus resultados reales no podrán ser evaluados en

---

<sup>2</sup> Origen de los datos: [https://cseweb.ucsd.edu/~jmcauley/datasets/amazon\\_v2/#complete-data](https://cseweb.ucsd.edu/~jmcauley/datasets/amazon_v2/#complete-data)

este ejercicio. A continuación, se describen estas métricas y se establece un umbral mínimo en el contexto de aplicación:

### **Métricas de Aprendizaje Automático (Desempeño de los Modelos)**

**MSE:** El MSE (Mean Squared Error, por sus siglas en inglés) es otra métrica comúnmente utilizada en estadísticas y aprendizaje automático para evaluar la calidad de un modelo de regresión. Al igual que el RMSE (Root Mean Square Error), el MSE cuantifica la diferencia entre los valores predichos por el modelo y los valores reales, pero en lugar de tomar la raíz cuadrada de la media de los errores al cuadrado, simplemente toma la media de los errores al cuadrado.

La fórmula del MSE se expresa de la siguiente manera:

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2$$

MSE, Mean Squared Error (Error Cuadrático Medio).

$n$ , Número de puntos

$Y_i$ , valores observados

$\hat{Y}_i$ , valores predichos

**Root Mean Square Deviation (RMSE):** (Root Mean Square Error, por sus siglas en inglés) es una métrica comúnmente utilizada en estadísticas y aprendizaje automático para evaluar la precisión de un modelo de predicción o regresión. Representa la raíz cuadrada de la media de los errores al cuadrado entre los valores predichos por el modelo y los valores reales.

La fórmula del RMSE se expresa de la siguiente manera:

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (Predicted_i - Actual_i)^2}{n}}$$

**Donde:**

- $n$  es el número total de observaciones en el conjunto de datos de prueba.
- $Predicted_i$  es el valor real de la variable dependiente para la observación  $i$ .
- $Actual_i$  es el valor predicho por el modelo para la observación  $i$ .

El RMSE cuantifica la diferencia promedio entre las predicciones del modelo y los valores reales, proporcionando una medida de la precisión global del modelo. Cuanto menor sea el valor del RMSE, mejor será el ajuste del modelo a los datos observados. Es una métrica especialmente útil cuando los errores del modelo tienden a ser ampliamente distribuidos y no se desea dar un peso excesivo a los errores grandes.

Se establecerá un umbral máximo de RMSE de 1.0. Esto significa que el RMSE debe ser menor o igual a 1.0 para considerar que los modelos de recomendación tienen un rendimiento adecuado. Un RMSE por debajo de 1.0 indicará que las recomendaciones son altamente precisas.

**Mean Absolute Error (MAE):** El MAE se emplea para evaluar la precisión absoluta de las predicciones de los modelos. Representa la diferencia promedio entre las calificaciones reales y las predicciones. Al igual que con el RMSD, un MAE más bajo indica una mayor precisión en las recomendaciones. Se establecerá un umbral mínimo de MAE para garantizar que las predicciones sean lo suficientemente precisas como para mejorar la experiencia del usuario.

$$MAE = \frac{\sum_{i=1}^n |y_i - x_i|}{n}$$

**Donde:**

MAE, Error Medio Absoluto.

$y_i$ , predicción

$x_i$ , valor verdadero

$n$ , número total de datos.

Se fijará un límite máximo de MAE de 0.7. Un MAE igual o inferior a 0.7 será considerado como un nivel aceptable de precisión en las predicciones de los modelos.

**MAPE:** El MAPE (Mean Absolute Percentage Error, por sus siglas en inglés) es una métrica utilizada para evaluar la precisión de un modelo de predicción, especialmente en problemas de pronóstico o regresión. A diferencia del MSE (Mean Squared Error) y el RMSE (Root Mean Square Error), el MAPE mide el porcentaje promedio de error absoluto en relación con los valores reales.

La fórmula del MAPE se expresa de la siguiente manera:

$$M = \frac{1}{n} \sum_{t=1}^n \left| \frac{A_t - F_t}{A_t} \right|$$

Donde:

$n$  es el número total de observaciones en el conjunto de datos de prueba.

$A_t$  es el valor real de la variable dependiente para la observación  $i$ .

$F_t$  es el valor predicho por el modelo para la observación  $i$ .

**k-fold cross-validation:** La validación cruzada k-fold es una técnica comúnmente utilizada en el ámbito del aprendizaje automático para evaluar el rendimiento de un modelo estadístico y mitigar el riesgo de sobreajuste (overfitting) o subajuste (underfitting) en un conjunto de datos. La idea fundamental detrás de la validación cruzada k-fold es dividir el conjunto de datos en  $k$  subconjuntos (o "folds") de aproximadamente igual tamaño.

### **Justificación**

La implementación de la Descomposición en Valores Singulares (SVD) en sistemas de recomendación implica la descomposición de la matriz que registra las interacciones entre usuarios e ítems. En este contexto, la elección del RMSE (Root Mean Square Error) como métrica se justifica debido a su capacidad para cuantificar la discrepancia entre las calificaciones anticipadas y las observadas, proporcionando así una evaluación de la precisión del modelo. La minimización del RMSE persigue la mejora de la capacidad del modelo SVD para predecir calificaciones con mayor fidelidad.

En el marco de un modelo de filtrado colaborativo que se vale de embeddings, la representación vectorial de usuarios e ítems se ajusta para anticipar sus interacciones. En virtud de que tanto el MSE (Mean Squared Error) como el MAE (Mean Absolute Error) miden la magnitud promedio de los errores absolutos, se consideran pertinentes para evaluar la disparidad entre las calificaciones predichas y las reales en este enfoque. La minimización de estas métricas contribuirá al perfeccionamiento de la precisión del modelo en la predicción de interacciones.

En el contexto de un modelo más sofisticado que integra una red neuronal multicapa, el MAPE (Mean Absolute Percentage Error) emerge como una métrica útil. Dado que este modelo tiene la capacidad de capturar relaciones no lineales entre usuarios e ítems, el MAPE, que mide el error absoluto como un porcentaje del valor real, brinda una perspectiva de la precisión relativa del modelo en términos proporcionales. La reducción del MAPE se convierte en un aspecto crucial para perfeccionar la precisión de las predicciones en este modelo más avanzado.

### **Métricas de Negocio (Desempeño del Proceso de Negocio)**

**Incremento en las Ventas:** Una métrica clave de negocio será el aumento en las ventas generadas por la implementación de los modelos de recomendación mejorados. Se establecerá un objetivo específico de aumento en las ventas en función de la mejora de las recomendaciones. El objetivo mínimo será lograr un aumento del 15% en las ventas después de la implementación de los modelos de recomendación mejorados. Esto significa que, en comparación con el estado previo a la implementación, las ventas deberán aumentar al menos un 15%.

La elección del 15% de aumento en ventas como métrica de negocio se justifica por los siguientes factores:

- Es un objetivo realista y alcanzable. Los modelos de recomendación mejorados tienen el potencial de mejorar la tasa de conversión de clientes potenciales en clientes, así como el valor promedio de cada venta (Jannach, D., & Jugovac, M., 2019). Sin embargo, es importante ser realista y establecer un objetivo que sea posible de alcanzar. Un aumento del 15% es un objetivo ambicioso, pero alcanzable con una implementación exitosa de los modelos de recomendación mejorados.
- Un estudio realizado por la consultora McKinsey & Company (McKinsey & Company, 2022) encontró que los modelos de recomendación mejorados pueden generar un aumento promedio de las ventas del 10%. Otro estudio, hecho por la empresa de investigación Forrester (Forrester, 2023), encontró que los modelos de recomendación mejorados pueden generar un aumento promedio de las ventas del 15%. Por tanto, la elección del 15% de aumento en ventas como métrica de negocio es una decisión fundamentada que refleja el valor potencial de los modelos de recomendación mejorados. Si los modelos de recomendación se implementan de manera exitosa, la empresa puede esperar un aumento significativo en las ventas.

**Retención de Clientes:** La retención de clientes es esencial en el comercio electrónico. Se medirá el impacto de los modelos en la retención de clientes y la lealtad a la plataforma. Se buscará lograr una tasa de retención de clientes del 80% o superior después de la implementación de los modelos. Esto significa que al menos el 80% de los clientes deberán realizar compras repetidas en la plataforma.

**ROI Estimado:** Se estimará el retorno de la inversión (ROI) en función de las mejoras en las ventas y la retención de clientes. Se establecerá un umbral mínimo de ROI que justifique la inversión en la implementación y operación de los modelos. Se considerará exitosa la implementación si se logra un ROI estimado de al menos el 20%. Esto significa que, en términos de ganancias adicionales generadas por las ventas y la retención de clientes, el retorno de la inversión será de al menos el 20%.

## **2. Objetivos**

### **2.1. Objetivo general**

El objetivo general de esta investigación es concebir, desarrollar y evaluar sistemas de recomendación, empleando análisis de sentimientos como componente adicional al de los modelos tradicionales de este tipo de sistemas. El propósito se centra en ofrecer a la industria e-commerce un respaldo a sus estrategias comerciales basado en técnicas analíticas y de aprendizaje automático para el comercio en línea. La delimitación temporal de esta indagación abarca desde el año 1996 hasta el año 2018 y se fundamenta en la exploración de conjuntos de datos de reseñas de productos proporcionados por Amazon.

El núcleo de este objetivo radica en la mejora sustancial de la precisión y relevancia de las recomendaciones de productos, junto con una comprensión de las preferencias y valoraciones emitidas por los usuarios. En esencia, se busca la formulación de un enfoque analítico de datos que genere un impacto positivo en el volumen de ventas y la fidelización de la clientela en el competitivo entorno del comercio electrónico contemporáneo.

En consecuencia, este estudio se posiciona como una contribución exploratoria a la convergencia de la analítica de datos y la estrategia empresarial, alentando una aproximación más precisa y personalizada para satisfacer las necesidades del consumidor. Al mismo tiempo, promueve una percepción más profunda de la interacción entre los usuarios y los productos ofrecidos en el mercado en línea. La consecución de este objetivo implica la implementación de modelos de recomendación innovadores, la aplicación de métricas de evaluación rigurosas y la identificación de umbrales críticos en términos de precisión y rendimiento de negocio.

## **2.2. Objetivos específicos**

Los objetivos específicos de esta investigación se formulan con el propósito de orientar las acciones a corto plazo que conduzcan al logro del objetivo general, y serán los siguientes:

- Realizar un análisis exhaustivo de la calidad y la estructura de los datos de reseñas proporcionados en el periodo de tiempo descrito, identificando posibles problemas de limpieza y preprocesamiento.
- Diseñar, implementar y ajustar modelos de recomendación basados en colaboración, contenido, y enfoques híbridos, incorporando análisis de sentimientos para mejorar la precisión de las recomendaciones.
- Evaluar el Desempeño de los Modelos utilizando métricas de evaluación, como el MSE, RMSE, MAE, MAPE y k-fold cross-validation, para medir el rendimiento de los modelos en términos de precisión y eficacia en la recomendación de productos.
- Determinar umbrales específicos para las métricas de precisión, como MSE, RMSE, MAE, MAPE, que indiquen un rendimiento adecuado de los modelos. Estos umbrales servirán como criterios de aceptación para la implementación de los modelos en un entorno de negocio.
- (Opcional) Medir el impacto de los modelos en el negocio, evaluando el aumento en las ventas, la retención de clientes y el ROI estimado después de la implementación.



- Sintetizar los resultados de la investigación para generar conclusiones sólidas y ofrecer recomendaciones prácticas para la implementación de sistemas de recomendación complementados con análisis de sentimientos en el comercio electrónico.

### **3. Datos**

#### **3.1. Datos originales**

El presente trabajo se basa en una exhaustiva recopilación de información relacionada con revisiones de productos y sus metadatos extraídos del comercio electrónico de Amazon. Se disponen de varios conjuntos de datos detallados, listos para su análisis y uso en investigaciones académicas. Estos conjuntos se dividen en revisiones de productos y metadatos, que abarcan tanto grandes conjuntos de datos como subconjuntos más manejables diseñados para aplicaciones específicas.

Los datos de las revisiones de productos están disponibles en varios formatos y tamaños, desde conjuntos de datos extensos hasta subconjuntos más manejables. Los conjuntos más amplios contienen millones de revisiones en formatos que incluyen; revisiones crudas; solo calificaciones; subconjuntos de revisiones “5-core” (con al menos 5 revisiones por usuario o producto). Además, se proporcionan metadatos que cubren una amplia gama de categorías de productos.

Por ejemplo, hay conjuntos de datos específicos para categorías como moda, belleza, electrodomésticos, automóviles, libros, electrónicos, entre otros. Cada categoría cuenta con un conjunto de revisiones y metadatos asociados. También se proporcionan subconjuntos más pequeños para experimentación o proyectos de menor escala.

Los datos de revisión se presentan en formato JSON, que incluye detalles como el ID del revisor, ID del producto, nombre del revisor, votos útiles, metadatos del producto, texto de la revisión, calificación general, resumen de la revisión y detalles de tiempo.

Por otro lado, los metadatos proporcionan datos sobre productos específicos, como el ID del producto, título, características en formato de viñetas, descripción, precio, enlaces de imágenes, productos relacionados, información de rango de ventas, marca, categorías y detalles técnicos.

Tanto los conjuntos completos como los subconjuntos más pequeños tienen como objetivo brindar a los investigadores opciones escalables y centradas en sus necesidades analíticas. Estos datos proporcionan una amplia gama de información que puede ser utilizada en investigaciones académicas y análisis de datos en el campo del comercio electrónico y la recomendación de productos.

### Reseñas

El formato original consiste en una reseña por línea, presentada en formato JSON. A continuación se muestran algunos ejemplos:

```
{
  "image": ["https://images-na.ssl-images-
amazon.com/images/I/71eG75FTJL._SY88.jpg"],
  "overall": 5.0,
  "vote": "2",
  "verified": True,
  "reviewTime": "01 1, 2018",
  "reviewerID": "AUI6WTTT0QZYS",
  "asin": "5120053084",
  "style": {
    "Size": "Large",
    "Color": "Charcoal"
  },
  "reviewerName": "Abbey",
  "reviewText": "I now have 4 of the 5 available colors of this
shirt... ",
  "summary": "Comfy, flattering, discreet--highly recommended!",
  "unixReviewTime": 1514764800
}

{
  "reviewerID": "A2SUAM1J3GNN3B",
  "asin": "0000013714",
  "reviewerName": "J. McDonald",
  "vote": 5,
  "style": {
    "Format": "Hardcover"
  },
  "reviewText": "I bought this for my husband who plays the piano.
He is having a wonderful time playing these old hymns. The music is
at times hard to read because we think the book was published for
singing from more than playing from. Great purchase though!",
  "overall": 5.0,
  "summary": "Heavenly Highway Hymns",
  "unixReviewTime": 1252800000,
  "reviewTime": "09 13, 2009"
}
```

**Figura 1:** Representación de la estructura de un registro del conjunto de evaluaciones - Imagen tomada de [“Amazon review data”](#).

Cada “Review” se presenta en un formato estructurado que contiene una serie de atributos, lo que permite analizar y comprender los detalles asociados con cada revisión. Aquí hay un resumen de los atributos presentes en estas revisiones:

- **reviewerID** - ID del usuario, ej. A2SUAM1J3GNN3B
- **asin** - ID del producto, ej. 0000013714
- **reviewerName** - Nombre del usuario
- **vote** - Votos
- **style** - Diccionario de los metadatos del producto, por ejemplo, “Formato” es “Tapa dura”
- **reviewText** - Texto de la calificación
- **overall** - Calificación del producto
- **summary** - Resumen de la calificación
- **unixReviewTime** - Hora de la revisión (hora Unix)
- **reviewTime** - hora de la revisión (raw)
- **image** - Imágenes que los usuarios publican después de haber recibido el producto

El conjunto de datos de “Sample review” representa un subconjunto que ilustra la estructura y presentación de las revisiones de productos dentro del conjunto de datos más extenso. Esta selección proporciona ejemplos concretos de revisiones de productos en formato JSON, lo que ayuda a comprender la organización y los atributos que conforman cada revisión.

Estos datos posibilitan un análisis detallado de las revisiones de productos, incluyendo información sobre los revisores, las calificaciones otorgadas a los productos, detalles específicos de los productos revisados y la fecha en que se realizaron las revisiones.

## Metadata

Los metadatos incluyen descripciones, precios, rango de ventas, información de marca y enlaces de recompra

```
{
  "asin": "0000031852",
  "title": "Girls Ballet Tutu Zebra Hot Pink",
  "feature": [
    "Botiquecutie Trademark exclusive Brand",
    "Hot Pink Layered Zebra Print Tutu",
    "Fits girls up to a size 4T",
    "Hand wash / Line Dry",
    "Includes a Botiquecutie TM Exclusive hair flower
    bow"],
  "description": "This tutu is great for dress up play for your
  little ballerina. Botiquecute Trade Mark exclusive brand. Hot Pink
  Zebra print tutu.",
  "price": 3.17,
  "imageURL": "http://ecx.images-
  amazon.com/images/I/51fAmVkBtbyL._SY300_.jpg",
  "imageURLHighRes": "http://ecx.images-
  amazon.com/images/I/51fAmVkBtbyL.jpg",
  "also_buy": [
    "B00JHONN1S", "B002BZX8Z6", "B00D2K1M30",
    "0000031909", "B00613WDTQ", "B00D0WDS9A", "B00D0GCI8S", "0000031895",
    "B003AVKOP2", "B003AVEU6G", "B003IEDM9Q", "B002R0FA24", "B00D23MC6W",
    "B00D2K0PA0", "B00538F50K", "B00CEV86I6", "B002R0FABA", "B00D10CLVW",
    "B003AVNY6I", "B002GZGI4E", "B001T9NUFS", "B002R0F7FE", "B00E1YRI4C",
    "B008UBQZKU", "B00D103F8U", "B007R2RM8W"],
  "also_viewed": [
    "B002BZX8Z6", "B00JHONN1S", "B008F0SU0Y",
    "B00D23MC6W", "B00AFDOPDA", "B00E1YRI4C", "B002GZGI4E", "B003AVKOP2",
    "B00D9C1WBM", "B00CEV8366", "B00CEUX0D8", "B0079ME3KU", "B00CEUWY8K",
    "B004F0EEHC", "0000031895", "B00BC4GY9Y", "B003XRKA7A", "B00K18LKX2",
    "B00EM7KAG6", "B00AMQ17JA", "B00D9C32NI", "B002C3Y6WG", "B00JLL4L5Y",
    "B003AVNY6I", "B008UBQZKU", "B00D0WDS9A", "B00613WDTQ", "B00538F50K",
    "B005C4Y4F6", "B004LHZ1NY", "B00CPHX76U", "B00CEUWUZZ", "B00IJVASUE",
    "B00G0R07RE", "B00J2GTM0W", "B00JHNSNSM", "B003IEDM9Q", "B00CYBU84G",
    "B008VV8NSQ", "B00CYBULSO", "B00I2UHSZA", "B005F50FXC", "B007LCQI3S",
    "B00DP68AVW", "B009RXWNSI", "B003AVEU6G", "B00HS0JB9M", "B00EHAGZNA",
    "B0046W9T8C", "B00E79VW6Q", "B00D10CLVW", "B00B0AV054", "B00E95LC8Q",
    "B00G0R92S0", "B007ZN5Y56", "B00AL2569W", "B00B608000", "B008F0SMUC",
    "B00BFXLZ8M"],
  "salesRank": {"Toys & Games": 211836},
  "brand": "Coxlures",
  "categories": ["Sports & Outdoors", "Other Sports", "Dance"]
}
```

**Figura 2:** Representación de la estructura de un registro del conjunto de metadatos - Imagen tomada de ["Amazon review data"](#).

Cada entrada de metadatos describe un producto específico y consta de una serie de atributos que proporcionan información detallada sobre dicho producto. A continuación se describen los atributos presentes en estos metadatos:

**asin** - ID del producto, por ejemplo, 0000031852

**title** - nombre del producto

**feature** - características del producto en formato de viñetas

**description** - descripción del producto

**price** - precio en dólares estadounidenses (al momento de la extracción de datos)

**imageURL** - URL de la imagen del producto

**imageURL** - URL de la imagen de alta resolución del producto

**related** - productos relacionados (también comprados, también vistos, comprados juntos, comprar después de ver)

**salesRank** - información de clasificación de ventas

**brand** - nombre de la marca

**categories** - lista de categorías a las que pertenece el producto

**tech1** - primera tabla de detalles técnicos del producto

**tech2** - segunda tabla de detalles técnicos del producto

























**similar** - tabla de productos similares

Este conjunto de datos facilita la comprensión de los detalles y atributos relevantes que podrían utilizarse además para análisis posteriores, como recomendaciones basadas en similitudes, segmentación de productos o análisis detallados de categorías de productos.

Para acceder a los datos es preciso descargarlos de nuestra fuente original: [Repositorio UC San Diego](#), o en la siguiente carpeta de Google Drive: [Data](#)

### 3.2. Datasets

El proceso de generación de conjuntos de datos de entrenamiento y validación a partir de los datos iniciales implica una serie de etapas metodológicas y transformaciones. Los conjuntos de datos a ser manipulados comprenden diversos archivos, los cuales fueron sometidos a un conjunto estandarizado de actividades de limpieza y análisis descriptivo.

Name	Owner	Last modified b...	File size
 Luxury_Beauty.json.gz 	 me	5:14 PM	69.8 MB
 meta_AMAZON_FASHION.json.gz 	 me	5:14 PM	31.5 MB
 AMAZON_FASHION.json.gz 	 me	5:14 PM	88.9 MB
 Clothing_Shoes_and_Jewelry.json.gz 	 me	5:14 PM	3.31 GB
 meta_Clothing_Shoes_and_Jewelry.json.gz 	 me	5:14 PM	1.46 GB
 DataFinal_Amazon.json.gz 	 me	5:14 PM	406.9 MB
 Clothing_Shoes_and_Jewelry_5.json.gz 	 me	5:15 PM	1.18 GB
 meta_Luxury_Beauty.json.gz 	 me	5:15 PM	7.3 MB





**Figura 3:** Captura de la lista de archivos con datos crudos en el repositorio de los autores de este documento en Google Drive.

Primero, se procedió a la conversión de cada archivo a un formato estructurado de datos mediante la creación de DataFrames. Posteriormente, se aplicaron técnicas para gestionar los valores faltantes y reducir la dimensionalidad de los conjuntos de datos. Este proceso tuvo como objetivo principal la preparación de los datos para su uso en modelos de aprendizaje automático. A cada uno de los archivos se le realizó el siguiente conjunto de actividades de limpieza y análisis:

- Conversión a DataFrame
- Manejo de valores faltantes
- Reducción de dimensionalidad
- Unión de la data de reviews y la metadata por categoría
- Ingeniería de Características (Feature Engineering)
- Creación de Características
- Almacenamiento en formatos CSV

Además, se llevó a cabo la unificación de la información proveniente de las revisiones (reviews) y los metadatos, realizando esta fusión con base en la categorización de los productos. A través de técnicas de ingeniería de características (Feature Engineering), se crearon nuevas variables significativas para mejorar el desempeño y la precisión de los modelos a construir.

El resultado de estas operaciones se tradujo en la creación de un conjunto de datos consolidado, denominado "DataFinal\_Amazon". Este conjunto integró la información proveniente de tres categorías específicas: Luxury, Fashion y Clothing, junto con sus respectivas metadatos, presentándose como un único corpus de datos listo para su análisis y aplicación en modelos de sistemas de recomendación.

Name	Owner	Last modified	File size
 DataFinal_Amazon.csv	 me	5:21PM me	1.17 GB
 DataFinal_Amazon.json.gz	 me	5:21PM me	406.9 MB

**Figura 4:** Captura de la lista de archivos con datos procesados en el repositorio de los autores de este documento en Google Drive.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4800571 entries, 0 to 4800570
Data columns (total 10 columns):
 #   Column          Dtype
---  -
 0   overall         float64
 1   reviewerID      object
 2   asin            object
 3   reviewText      object
 4   summary         object
 5   brand           object
 6   price           float64
 7   extracted_rank  int64
 8   reviewDate      object
 9   category        object
dtypes: float64(2), int64(1), object(7)
memory usage: 366.3+ MB
```

**Figura 5:** Información de la estructura del dataset procesado DataFinal\_Amazon

Con el propósito de emplear estos datos en la iteración de modelos de sistemas de recomendación, se realizó una división estratégica de la información en conjuntos de entrenamiento y prueba. Esta división se estableció con una proporción de 70/30, asignando el



70% de los datos al conjunto de entrenamiento y el 30% restante al conjunto de prueba. Esta división es una práctica común en la ciencia de datos y el aprendizaje automático, y se basa en una heurística que busca equilibrar la necesidad de tener suficientes datos para entrenar el modelo (conjunto de entrenamiento) con la necesidad de tener un conjunto independiente de datos para evaluar su rendimiento (conjunto de prueba).

Es importante señalar que esta partición se ejecutó estableciendo una semilla específica. La elección de esta semilla tuvo como finalidad asegurar la reproducibilidad de la división de datos. En otras palabras, al utilizar esta semilla en la división de datos, se garantiza la obtención consistente de los mismos resultados cada vez que se repite el proceso.

El proceso de generación de conjuntos de datos de entrenamiento y validación se llevó a cabo empleando la biblioteca "scikit-learn"<sup>3</sup> de Python. Específicamente, se utilizó la función `train_test_split`<sup>4</sup> proveniente del módulo `model_selection` de la mencionada biblioteca. La utilización de esta función junto con la definición de la proporción deseada y la semilla especificada para asegurar la reproducibilidad, garantiza una distribución consistente y replicable de los datos para el entrenamiento y evaluación de modelos de sistemas de recomendación.

### 3.3. Analítica descriptiva

**Notebook:** `Analitica_Descriptiva_Amazon.ipynb`<sup>5</sup>

Previo a la implementación y entrenamiento de los modelos de recomendación, era necesario entender, resumir y presentar de manera significativa la información contenida en el conjunto de datos final (`DataFinal_Amazon`). Esencial para la comprensión integral del conjunto de datos, con el propósito de identificar patrones, tendencias, características sobresalientes y posibles anomalías. A través de la aplicación de técnicas estadísticas, tales como la determinación de la media, la mediana, la moda, la desviación estándar y otros indicadores descriptivos, se obtuvo una síntesis concisa que proporciona una visión detallada de la distribución de los datos. La representación visual de datos a través de gráficos complementa este proceso al ofrecer una

---

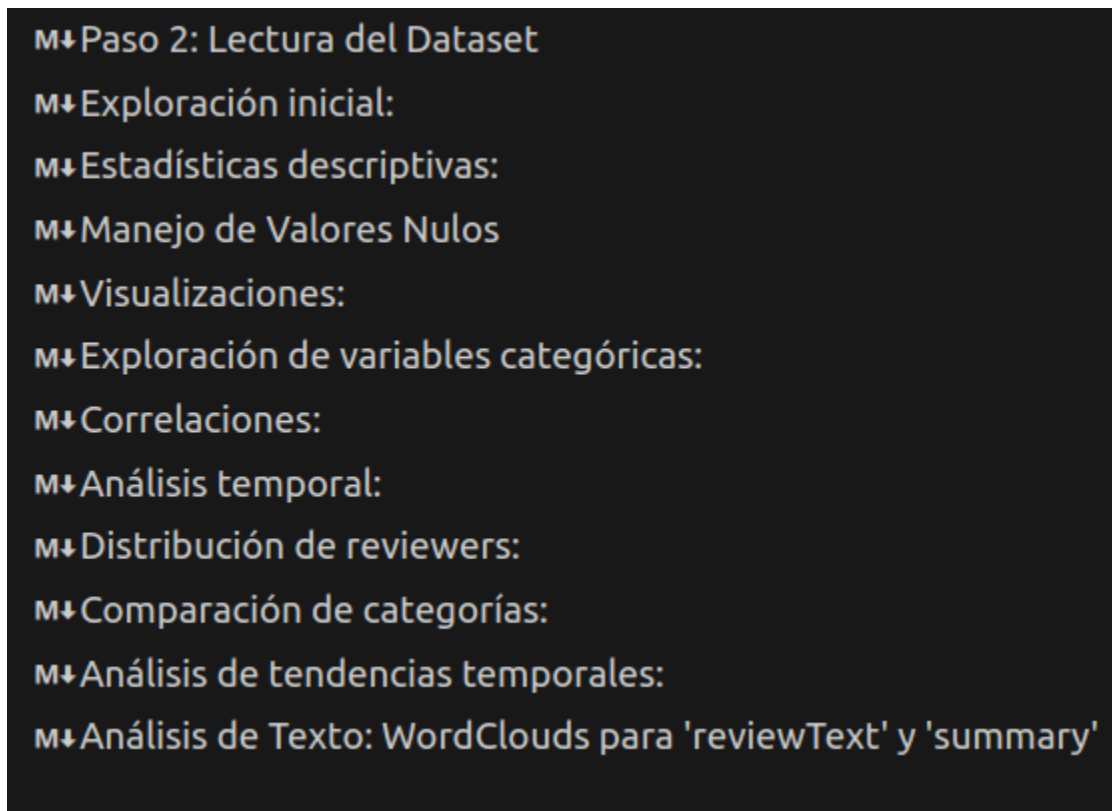
<sup>3</sup> <https://scikit-learn.org/stable/>

<sup>4</sup> [https://scikit-learn.org/stable/modules/generated/sklearn.model\\_selection.train\\_test\\_split.html](https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html)

<sup>5</sup> [https://github.com/lmbd92/DataScienceMonograph/blob/main/Notebooks/Analitica\\_Descriptiva\\_Amazon.ipynb](https://github.com/lmbd92/DataScienceMonograph/blob/main/Notebooks/Analitica_Descriptiva_Amazon.ipynb)

perspectiva más intuitiva, permitiendo la identificación de relaciones y patrones poco evidentes en una exploración numérica.

**Proceso:** El proceso de analítica descriptiva se inició con la lectura de los datos, seguida de una exploración inicial para entender la estructura y características del conjunto de datos. Posteriormente, se aplican técnicas estadísticas descriptivas para resumir la información de manera cuantitativa, calculando medidas centrales como la media, mediana, moda, así como la dispersión a través de la desviación estándar y el rango. Estas estadísticas proporcionan una visión general de la distribución y variabilidad de los datos, a continuación compartimos detalles del proceso.



**Figura 6:** Lista de pasos ejecutados en la etapa de análisis descriptivo.

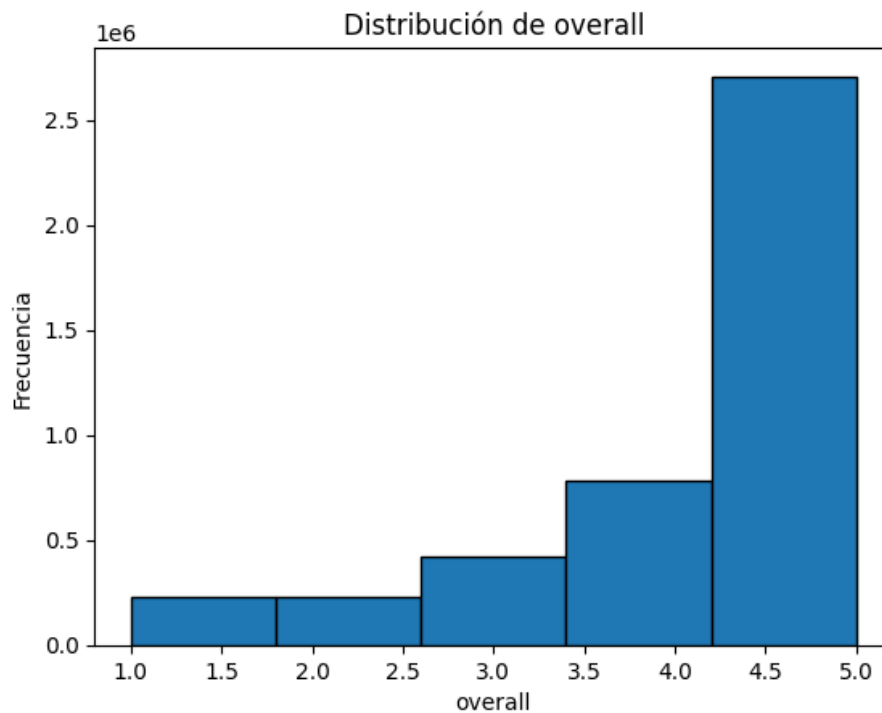
```

1 # Estadísticas descriptivas para columnas numéricas
2 print(df.describe())

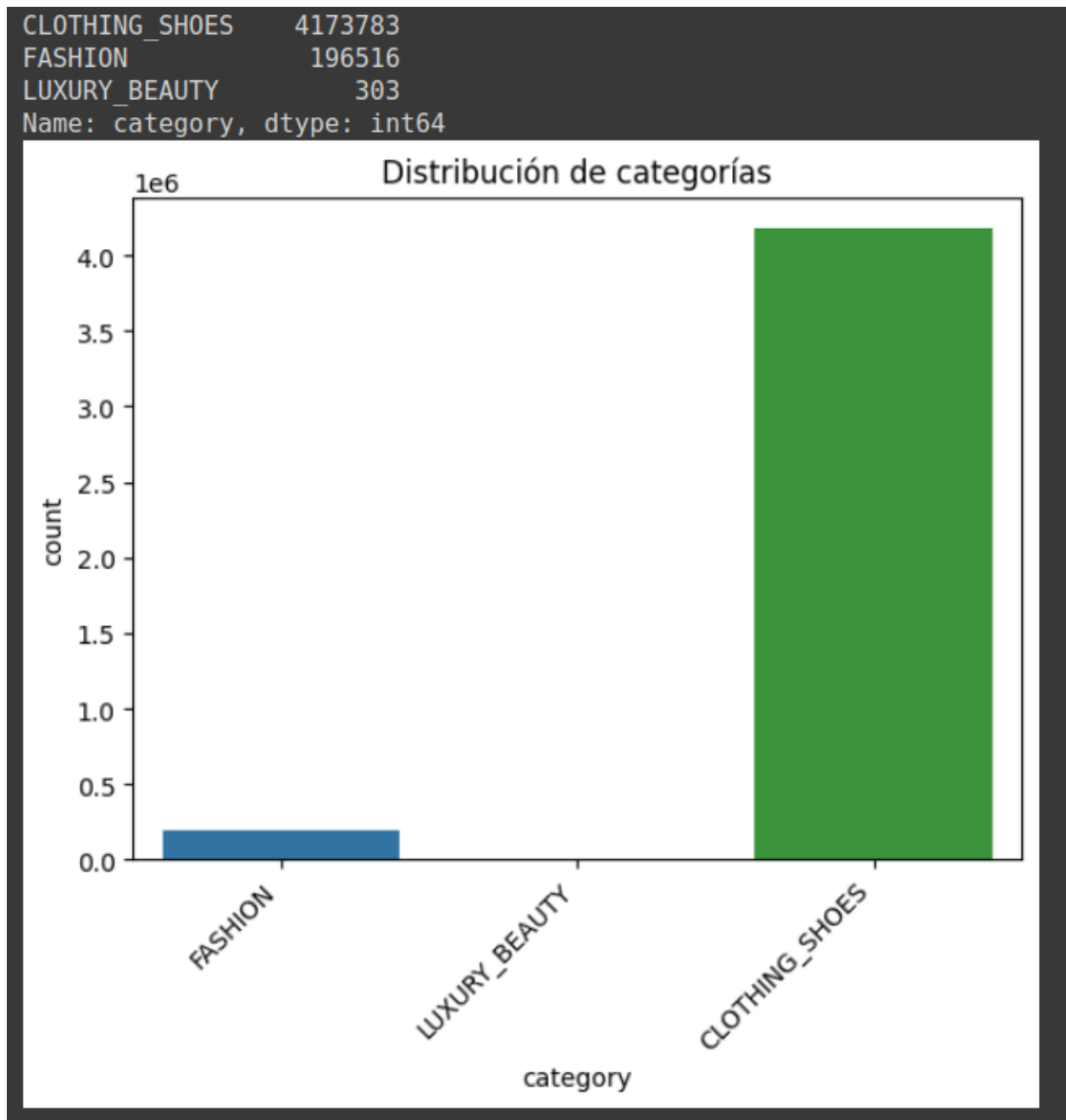
```

	overall	price	extracted_rank
count	4.800571e+06	4.800571e+06	4.800571e+06
mean	4.260527e+00	9.346523e+01	3.631274e+05
std	1.166258e+00	2.293804e+03	7.122776e+05
min	1.000000e+00	1.000000e-02	0.000000e+00
25%	4.000000e+00	9.990000e+00	1.813900e+04
50%	5.000000e+00	1.599000e+01	9.874500e+04
75%	5.000000e+00	2.800000e+01	4.592440e+05
max	5.000000e+00	7.760000e+04	2.707478e+07

**Figura 7:** Resultados estadísticas descriptivas para columnas numéricas “Overall”, “Price” y “extracted\_rank”

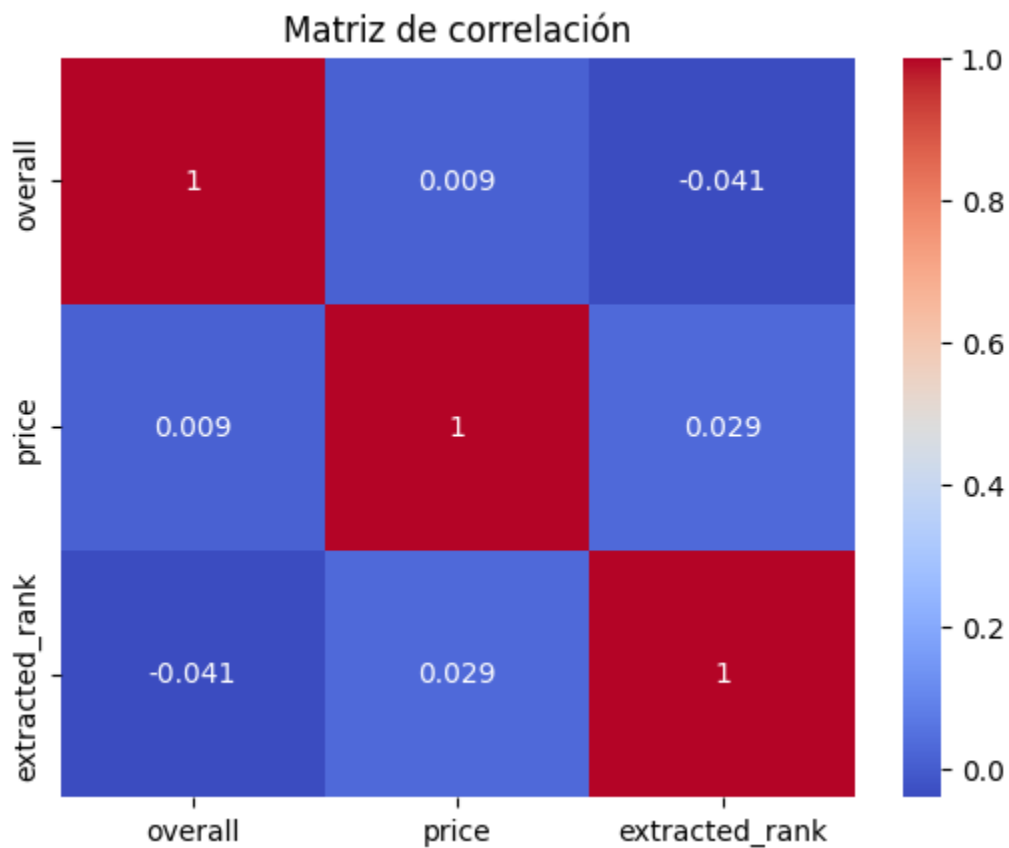


**Figura 8:** Distribución de los valores de la característica “Overall” - escala  $10^6$



**Figura 9:** Distribución de las categorías de productos en el conjunto de datos - escala  $10^6$

En este caso, se utiliza un gráfico de barras para representar la frecuencia de cada categoría en la columna 'category' del conjunto de datos. La altura de cada barra indica cuántas veces aparece cada categoría, proporcionando una representación visual de la frecuencia relativa de cada clase.



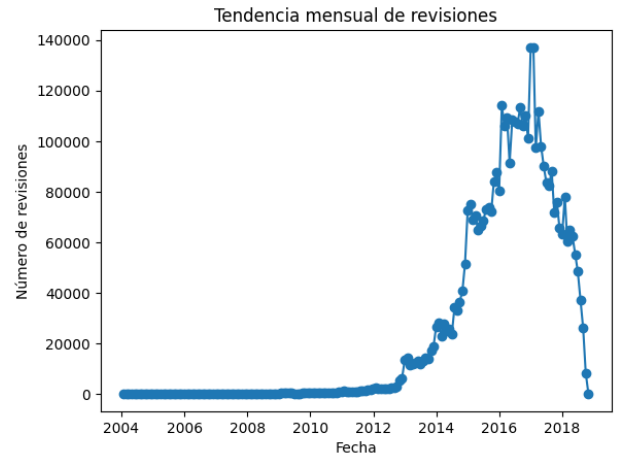
**Figura 10:** Matriz de correlación de las características numéricas del conjunto de datos

La matriz de correlación muestra que las asociaciones lineales entre 'overall', 'price', y 'extracted\_rank' son generalmente débiles en nuestro conjunto de datos. El coeficiente de correlación más notable es el valor negativo entre 'overall' y 'extracted\_rank' (-0.041), sugiriendo una correlación negativa muy débil. Esto implica que a medida que el 'overall' aumenta, el 'extracted\_rank' tiende a disminuir ligeramente, aunque la relación no es fuerte y podría ser no significativa. La correlación entre 'price' y 'extracted\_rank' (0.029) es positiva pero igualmente débil, indicando que no hay una relación lineal clara entre el precio y el rango extraído. Curiosamente, no hay prácticamente ninguna correlación entre 'price' y 'overall' (0.009), lo que sugiere que el precio no es un predictor del 'overall'. Estos resultados podrían implicar que otros factores no capturados por estas variables pueden estar influenciando las calificaciones 'overall' y los rangos extraídos, y que el precio no tiene un impacto directo significativo en la percepción de calidad o en el posicionamiento en el rango.

A su vez, el desarrollo de un gráfico de recuento de revisiones a lo largo del tiempo para la variable "reviewDate" tiene como objetivo visualizar la evolución de las revisiones a lo largo de un período temporal específico. Este tipo de visualización temporal es particularmente útil para identificar patrones estacionales, tendencias o eventos significativos en la frecuencia de revisiones a lo largo del tiempo.



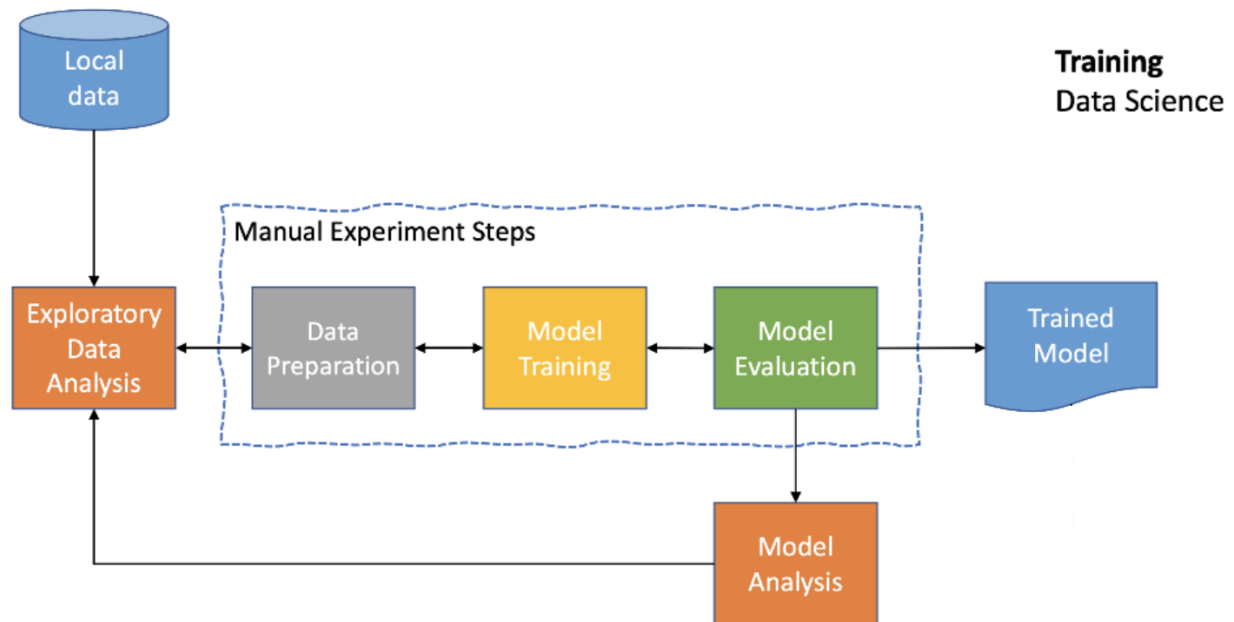
**Figura 11:** Gráfico de evolución de las revisiones a lo largo del periodo comprendido entre 2004 y 2018



**Figura 12:** Gráfico de la tendencia mensual de revisiones a lo largo del periodo comprendido entre 2004 y 2018

## 4. Proceso de analítica

### 4.1. Pipeline principal



**Figura 13:** Arquitectura del flujo de datos empleado - Original de Maggie Mhanna: MLOps Best Practices for Data Scientist"

La imagen anterior representa el flujo de trabajo propuesto, en donde buscamos replicar un flujo genérico tipo MLOps (Kreuzberger, Kühn, & Hirschl, 2023, p. 31868) que se utiliza para desarrollar modelos predictivos o descriptivos a partir de datos. Este flujo de trabajo consta de varias etapas clave, cada una de ellas crucial para asegurar la confiabilidad y utilidad del modelo final. A continuación, se detallan los componentes del diagrama:

### **Datos locales**

Nuestro punto de partida son los datos recopilados del repositorio oficial de la Universidad de California<sup>6</sup>, los cuales han sido almacenados localmente y puestos a disposición en el sistema de almacenamiento en la nube de Google Drive. Estos datos constituirán la fuente principal para el análisis subsiguiente. Cabe destacar que los datos locales se presentan en dos formatos distintos: gz (formato comprimido) y CSV (formato de procesamiento), siendo necesario considerar la posibilidad de que requieran ser sometidos a procesos de limpieza o transformación antes de ser empleados eficazmente en el contexto del entrenamiento del modelo.

### **Análisis de datos exploratorios (EDA)**

Antes de que los datos fueran preparados para el modelado, es esencial llevar a cabo un análisis exploratorio de datos (EDA, de sus siglas en inglés). EDA es un enfoque analítico para entender y resumir las principales características de un conjunto de datos, a menudo con métodos visuales. En esta etapa incluimos la visualización de datos, el análisis de estadísticas descriptivas y la identificación de posibles relaciones o anomalías en los datos.

### **Pasos de la experimentación manual**

El procedimiento de experimentación manual se presenta como una serie de pasos que fueron llevados a cabo de manera iterativa con el fin de construir los modelos de datos.

- a. Preparación de Datos:** En esta fase inicial, se llevó a cabo la limpieza de los datos, la selección de características relevantes (feature selection), así como su transformación y codificación para posibilitar su procesamiento por parte de los algoritmos de sistemas de recomendación y análisis de sentimientos. Este proceso abarcó la gestión de valores faltantes, la normalización de datos y la creación de nuevas características.

En el contexto de la determinación de trabajar con tres categorías específicas, se evidenció que los archivos de revisiones albergaban una voluminosa cantidad de registros, alcanzando cifras en el orden de millones. Ante este escenario, se tomó la decisión estratégica de buscar información adicional en la metadata con el propósito de

---

<sup>6</sup> <https://cseweb.ucsd.edu/~jmcauley/datasets/>



complementar los datos existentes. La ejecución de un procedimiento de unión (join) entre los conjuntos de datos de revisiones y metadata permitió reducir significativamente la cantidad de registros, preservando únicamente aquellos que presentaban una calidad de datos óptima entre revisiones y metadata, excluyendo aquellos registros que presentaban campos vacíos. Este enfoque, orientado hacia la preservación de la integridad y confiabilidad de los datos, ha contribuido sustancialmente a la optimización del conjunto de datos para análisis posteriores.

Finalmente, procedimos a la unificación de las tres categorías, culminando en la generación del conjunto de datos definitivo destinado al entrenamiento y evaluación de los modelos. Como resultado, se obtuvo un conjunto de datos que incorpora 10 características distintivas y se compone de un total de 4,800,571 registros.

- b. Entrenamiento del Modelo:** En esta etapa, se implementaron los algoritmos de aprendizaje automático seleccionados previamente en los datos preparados. El modelo "aprende" a identificar patrones en los datos de entrenamiento, y este proceso se ajustó posteriormente mediante la variación de parámetros y configuraciones, como la regularización y otras configuraciones válidas en el proceso de ajuste de modelos de aprendizaje automático.
- c. Evaluación del modelo:** Una vez que el modelo ha sido entrenado, es evaluado para determinar su rendimiento. Se utilizaron métricas de evaluación como el MSE, RMSE, MAE, y el MAPE para valorar la precisión del modelo para realizar predicciones en datos de prueba, los cuales no había visto previamente.

### **Análisis de modelos**

Después de la evaluación, se procede a un análisis más detallado del modelo para entender cómo es el proceso de toma de decisiones, identificar las características más relevantes y detectar la posible presencia de sesgos o problemas que requieren atención. Este análisis puede llevar a un ciclo de refinamiento del modelo, volviendo a las etapas de preparación de datos o entrenamiento.

### **Modelo entrenado**

El proceso de entrenamiento y evaluación culmina en la obtención de modelos entrenados, cuyo rendimiento es crucial para la generación de recomendaciones basadas en nuevos datos

#### **1. Modelo de Descomposición en Valores Singulares (SVD):**

Este modelo se destaca por su capacidad para factorizar la matriz de interacciones entre usuarios e ítems. Se espera que, tras el entrenamiento y la evaluación, proporcione recomendaciones precisas al aprovechar la información latente descubierta durante la descomposición.

#### **2. Modelo de Filtrado Colaborativo con Embedding mediante Keras:**

Basado en la técnica de Embedding, busca aprender representaciones vectoriales de usuarios e ítems para mejorar la precisión de las recomendaciones. Durante la fase de entrenamiento y evaluación, se afinan los parámetros para capturar patrones de colaboración y ofrecer recomendaciones más personalizadas.

#### **3. Modelo de Filtrado Colaborativo con Red Neuronal Multicapa mediante Keras:**

El enfoque de una red neuronal multicapa en el Modelo de Filtrado Colaborativo busca capturar relaciones no lineales más complejas entre usuarios e ítems. Tras el entrenamiento y evaluación, se espera que genere recomendaciones más sofisticadas, adaptándose mejor a la complejidad inherente en las interacciones colaborativas.

#### **4. Modelo de Filtrado Colaborativo con Embedding y Red Neuronal Multicapa mediante Keras:**

Este modelo fusiona las ventajas del embedding y una red neuronal multicapa, integrando la capacidad de aprendizaje de representaciones más precisas y modelar relaciones no lineales complejas. El resultado del entrenamiento y evaluación de este modelo se traduce en un enfoque más integral para la generación de recomendaciones, aprovechando las fortalezas complementarias de ambas técnicas.

## **Flujo de Trabajo de Datos en el Proyecto:**

- **Adquisición de Datos**

El proceso se inicia con la obtención de datos provenientes de los archivos de registros JSON, almacenados manualmente en el sistema Google Drive. Se realiza una recopilación exhaustiva de información relacionada con productos de Amazon en las categorías Luxury, Fashion y Clothing, incluyendo metadatos asociados.

- **Preprocesamiento Inicial**

Los datos obtenidos se someten a una primera fase de limpieza, donde se eliminan posibles errores y valores atípicos, además de una revisión inicial para validar formatos, corregir valores nulos o inconsistentes. Esto incluye una exploración inicial de las características relevantes para comprender la estructura y la naturaleza de los datos.

- **Conversión y Estructuración de Datos**

Se procede a convertir los datos adquiridos en un formato tabular, generalmente mediante la representación de los datos en DataFrames. Este paso es esencial para permitir un manejo eficiente y uniforme de la información, lo que facilita las operaciones posteriores.

- **Limpieza y Transformación**

Se realizan actividades avanzadas de limpieza, tales como el manejo de valores faltantes, la reducción de la dimensionalidad, y la unión de la información de revisiones y metadatos por categoría. Además, se emplean técnicas de ingeniería de características para crear nuevas variables que puedan potenciar los modelos de análisis.

- **Creación de Dataset Consolidado**

Se integran los datos procesados de las categorías Luxury, Fashion y Clothing junto con sus respectivos metadatos en un único conjunto de datos llamado "DataFinal\_Amazon". Este dataset consolidado es el resultado de la fusión y preparación de los datos para su posterior análisis.

- **División de Datos para Modelado**

Utilizando la función `train_test_split` de la biblioteca `scikit-learn`, se procede a dividir el conjunto de datos consolidado en datos de entrenamiento y prueba. La proporción de división establecida es del 70% para entrenamiento y 30% para pruebas. Además, se especifica una semilla para garantizar la reproducibilidad de la división.

- **Modelado y Evaluación**

Se implementan modelos de sistemas de recomendación, como filtrado colaborativo, análisis de contenido, o métodos híbridos. Estos modelos se entrenan con el conjunto de datos de entrenamiento y se evalúan su desempeño y precisión utilizando el conjunto de datos de prueba.

- **Ajuste y Optimización**

Se realizan iteraciones en el modelado, ajustando parámetros, empleando técnicas de validación cruzada y optimización de hiperparámetros para mejorar el rendimiento de los modelos.

## **4.2. Modelos**

En esta sección, se describen en detalle varios modelos que han sido objeto de consideración en el contexto del presente trabajo, detallando sus configuraciones y particularidades. Se presenta una exhaustiva evaluación de diversos modelos, cada uno meticulosamente diseñado para afrontar el desafío intrínseco asociado a la generación de recomendaciones, enriquecida por el análisis de sentimientos expresados en los comentarios asociados a cada producto. A continuación, se ofrece una descripción concisa que destaca las características y configuraciones contempladas para cada modelo.

**Modelo de Descomposición en Valores Singulares (SVD)**

Se contempla la implementación de la Descomposición en Valores Singulares como estrategia para la generación de recomendaciones fundadas en interacciones históricas (Sarwar et al., 2001). Esta elección se fundamenta en la capacidad inherente de la SVD para factorizar matrices de interacciones entre usuarios e ítems, ofreciendo así una perspectiva valiosa para la realización de recomendaciones precisas.

**Modelo de Filtrado Colaborativo con Embedding mediante Keras**

Una segunda alternativa considerada es la utilización de un modelo de Filtrado Colaborativo que incorpora embeddings y se implementa a través de la biblioteca Keras (He et al., 2017). Este enfoque implica la representación vectorial de usuarios e ítems para la predicción de sus interacciones, destacándose por su capacidad para capturar relaciones complejas en datos colaborativos.

**Modelo de Filtrado Colaborativo con Red Neuronal Multicapa mediante Keras**

Otra modalidad abordada consiste en la aplicación de un modelo de Filtrado Colaborativo que integra una red neuronal multicapa, también implementada con la plataforma Keras. La elección de este enfoque se respalda en la capacidad de las redes neuronales multicapa para aprender patrones no lineales en conjuntos de datos colaborativos, lo que podría mejorar la capacidad predictiva del modelo.

**Modelo de Filtrado Colaborativo con Embedding y Red Neuronal Multicapa mediante Keras**

Por último, se contempla una amalgama de las estrategias anteriores, incorporando tanto embeddings como una red neuronal multicapa en el marco de un modelo de Filtrado Colaborativo (Zhang et al., 2020). Esta elección se fundamenta en la sinergia potencial entre ambas técnicas para abordar de manera integral la complejidad de las relaciones presentes en datos colaborativos.

### 4.3. Métricas

En el ámbito de la evaluación de modelos de recomendación, se procederá a describir detalladamente el cálculo de métricas de desempeño de Aprendizaje automático (ML) y métricas de negocio asociadas a los modelos especificados. La fundamentación teórica y la aplicación práctica de estas métricas proporcionarán una visión integral de la eficacia y pertinencia de cada modelo considerado.

#### Utilización de SVD para hacer recomendaciones basadas en interacciones pasadas

- **Métricas de Desempeño en ML:**

- MSE (Mean Squared Error): La implementación se lleva a cabo mediante funciones disponibles en la biblioteca scikit-learn, precisamente a través de `mean_squared_error` en el módulo `metrics`.
- RMSE (Root Mean Square Error): Se obtiene extrayendo la raíz cuadrada del MSE, mediante funcionalidades de biblioteca `numpy` de Python.
- MAE (Mean Absolute Error): Su cálculo se realiza mediante `mean_absolute_error` de scikit-learn.

```

14
15 # Función para calcular el Mean Absolute Percentage Error (MAPE)
16 def mape(y_true, y_pred):
17     y_true, y_pred = np.array(y_true), np.array(y_pred)
18     return np.mean(np.abs((y_true - y_pred) / y_true)) * 100
19
20 # Evaluar el modelo
21 rmse_svd = accuracy.rmse(predictions_svd, verbose=True)
22 mse_svd = mean_squared_error([pred.r_ui for pred in predictions_svd], [pred.est for pred in predictions_svd])
23 mae_svd = mean_absolute_error([pred.r_ui for pred in predictions_svd], [pred.est for pred in predictions_svd])
24 mape_svd = mape([pred.r_ui for pred in predictions_svd], [pred.est for pred in predictions_svd])
25
26 print(f"SVD - RMSE: {rmse_svd}, MSE: {mse_svd}, MAE: {mae_svd}, MAPE: {mape_svd}")
27
RMSE: 1.0861
SVD - RMSE: 1.0861342446627054, MSE: 1.1796875974290253, MAE: 0.8189192991254555, MAPE: 33.93144862604744

```

*Figura 14:* Métricas de desempeño aplicadas al modelo SVD

## Modelo de Filtrado Colaborativo con Embedding mediante Keras

- **Métricas de Desempeño ML:**

Se aplican las mismas métricas planteadas para el anterior modelo, MSE (Mean Squared Error): En este caso, el MSE evaluará la discrepancia entre las calificaciones predichas por el modelo de filtrado colaborativo y las calificaciones reales en el conjunto de datos. usando la función `mean_squared_error` de scikit-learn después de entrenar el modelo de Keras.

RMSE (Root Mean Square Error) al igual que en el caso anterior, el RMSE se puede obtener tomando la raíz cuadrada del MSE con `numpy.sqrt`.

MAE (Mean Absolute Error), La MAE proporcionará una medida de la diferencia promedio entre las calificaciones predichas y las calificaciones reales en términos absolutos. Calculado utilizando la función `mean_absolute_error` de scikit-learn.

```
1 # MSE
2 mse = mean_squared_error(ratings_test, predictions)
3 print(f"MSE: {mse:.4f}")
4
5 # RMSE
6 rmse = np.sqrt(mse)
7 print(f"RMSE: {rmse:.4f}")
8
9 # MAE
10 mae = mean_absolute_error(ratings_test, predictions)
11 print(f"MAE: {mae:.4f}")
12
13 # MAPE (Mean Absolute Percentage Error)
14 mape = np.mean(np.abs((ratings_test - predictions.flatten()) / ratings_test)) * 100
15 print(f"MAPE: {mape:.2f}%")

MSE: 18.1200
RMSE: 4.2568
MAE: 3.9480
MAPE: 92.92%
```

**Figura 15:** Métricas de desempeño aplicadas al modelo de Filtrado Colaborativo con Embedding mediante Keras

## Modelo de Filtrado Colaborativo con Embedding y Red Neuronal Multicapa mediante Keras

- **Métricas de Desempeño ML:**

- MAPE: En el contexto del Modelo de Filtrado Colaborativo con Embedding y Red Neuronal Multicapa implementado mediante Keras, se destaca la consideración especial del MAPE. En este caso particular, nos enfrentamos a restricciones de capacidad de procesamiento en el entorno de desarrollo, lo que motivó la implementación de un enfoque basado en lotes (batch). Concretamente, se optó por un tamaño de lote (batch\_size) equivalente a 50,000 registros, una elección estratégica en relación con el volumen total de datos, que asciende a 4'800.571 registros, porque el entorno de trabajo no soportaba realizar sobre la totalidad de los datos el cálculo de esta métrica..

```

1 # MSE y RMSE
2 mse = mean_squared_error(y_true, predictions)
3 rmse = np.sqrt(mse)
4
5 # MAE
6 mae = mean_absolute_error(y_true, predictions)
7
8 # MAPE
9 def mean_absolute_percentage_error(y_true, y_pred):
10     y_true, y_pred = np.array(y_true), np.array(y_pred)
11     non_zero_idx = y_true != 0 # Para evitar divisiones por cero
12     return np.mean(np.abs((y_true[non_zero_idx] - y_pred[non_zero_idx]) / y_true[non_zero_idx])) * 100
13
14 def compute_mape_by_batch(y_true, predictions, batch_size=50000):
15     mape_sum = 0
16     num_batches = int(np.ceil(len(y_true) / batch_size))
17
18     for i in range(num_batches):
19         start_idx = i * batch_size
20         end_idx = start_idx + batch_size
21
22         batch_y_true = y_true[start_idx:end_idx]
23         batch_predictions = predictions[start_idx:end_idx]
24
25         mape_sum += mean_absolute_percentage_error(batch_y_true, batch_predictions)
26
27     return mape_sum / num_batches
28
29 # Luego llamas a la función
30 mape = compute_mape_by_batch(y_true, predictions)
31
32 print(f'MSE: {mse}')
33 print(f'RMSE: {rmse}')
34 print(f'MAE: {mae}')
35 print(f'MAPE: {mape}%')

```

MSE: 1.514468685820065  
RMSE: 1.2306375119506414  
MAE: 0.8925390006771722  
MAPE: 41.067817144286046%

**Figura 16:** Métricas de desempeño aplicadas al modelo de Filtrado Colaborativo con Embedding y Red Neuronal Multicapa mediante Keras



## 5. Metodología

### 5.1. Baseline

En el notebook "6. Iteracion\_1\_Amazon"<sup>7</sup> se presenta un análisis de datos y la construcción de los modelos considerados para generar un sistema de recomendación basado en un conjunto de datos de reseñas de Amazon. El objetivo es generar recomendaciones de los productos basándose en interacciones pasadas de los usuarios.

### Metodología

La metodología aplicada sigue un enfoque típico de ciencia de datos, comenzando con la importación de bibliotecas esenciales como Pandas, NumPy, Scikit-learn y Keras. Se realiza un análisis exploratorio de datos (EDA) para comprender las características del dataset, seguido de un preprocesamiento que incluye la codificación de las variables categóricas 'reviewerID' y 'asin' a índices numéricos secuenciales.

En el desarrollo de esta iteración, se persigue ejecutar un acercamiento inicial a los modelos de recomendación seleccionados para el presente estudio. Dichos modelos consisten en la Descomposición en Valores Singulares, el Filtrado Colaborativo con Embedding mediante la plataforma Keras, el Filtrado Colaborativo con Red Neuronal Multicapa mediante Keras, y el Filtrado Colaborativo con Embedding y Red Neuronal Multicapa mediante Keras. La actividad contempló la utilización del conjunto de datos, dividido en conjuntos de entrenamiento y prueba, en una proporción del 70% y un 30%, respectivamente. La finalidad principal radicó en el entrenamiento y evaluación de los modelos con el propósito de evidenciar su desempeño.

Cabe destacar que en esta fase no se profundizó en la optimización de los modelos, ya que la intención era establecer un punto de partida inicial. Este enfoque inicial nos ha permitido contrastar y contextualizar las iteraciones posteriores de manera más eficaz.

---

<sup>7</sup> <https://shorturl.at/gjwyP>

## Entrenamiento y Evaluación

El modelo se entrena utilizando un conjunto de datos de entrenamiento y se evalúa mediante un conjunto de pruebas. Para la evaluación, se aplican métricas estándar en aprendizaje automático, como el error cuadrático medio (MSE) y el error absoluto medio (MAE). Además, para lograr una evaluación más exhaustiva, se calculan adicionalmente el RMSE (Root Mean Square Error) y el MAPE (Mean Absolute Percentage Error). Estas métricas adicionales son fundamentales para complementar y enriquecer nuestra comprensión del rendimiento del modelo. El RMSE, al ser sensible a los valores atípicos, proporciona una perspectiva sobre la variabilidad de los errores del modelo, mientras que el MAPE ofrece una medida relativa de los errores, lo que es útil para entender la precisión en términos porcentuales. Así, el uso combinado de estas métricas permite una interpretación más detallada y matizada de la efectividad del modelo, destacando tanto su precisión como su capacidad de generalización en diferentes contextos y tipos de datos.

## Resultados

Los resultados obtenidos a través del análisis de los datos en el notebook reflejan un proceso iterativo y metódico de ajuste y evaluación del modelo de filtrado colaborativo, utilizando la técnica de embedding con Keras. Un 'epoch' en este contexto se refiere a una iteración completa del algoritmo de aprendizaje a través de todo el conjunto de datos, y es crucial para determinar cuántas veces el modelo está expuesto a los datos durante el entrenamiento. Inicialmente, el modelo se entrenó con una configuración de 3 epochs, lo que permite un equilibrio entre el aprendizaje suficiente y la prevención del sobreajuste. Además, se utilizó un tamaño de lote (batch size) de 4096 registros, que determina la cantidad de datos procesados antes de actualizar los parámetros del modelo. Este tamaño de lote se eligió para optimizar el uso de memoria y la eficiencia computacional. Esta primera ejecución estableció una línea base para el desempeño del modelo, obteniendo los siguientes valores para las métricas de evaluación (**Tabla 1**).

Posteriormente, se llevó a cabo una segunda ejecución en la que se modificaron los hiperparámetros (número de épocas y tamaño de lote), reduciendo el número de épocas a 1 y disminuyendo el tamaño de lote a 2048. Esta variación tenía como objetivo explorar el impacto de un entrenamiento más rápido y frecuente en la actualización de los pesos en la capacidad predictiva del modelo.

Finalmente, en una tercera iteración, se mantuvo el tamaño de lote de la segunda ejecución (2048) y se incrementó el número de epochs a 5. Este ajuste fue realizado para examinar cómo un entrenamiento más prolongado afectaría la precisión y la generalización del modelo.

Cada iteración fue evaluada, comparando las métricas de pérdida y precisión para determinar la configuración óptima. Los resultados de estas ejecuciones permitieron una comprensión de la sensibilidad del modelo a estos hiperparámetros clave y ayudaron a guiar hacia la configuración más efectiva para el rendimiento del modelo en datos no vistos, obteniendo los siguientes resultados. En el presente análisis, se exponen los resultados obtenidos como producto de estos procedimientos. Además, se llevaron a cabo experimentos iniciales con los modelos remanentes, a saber, la Descomposición en Valores Singulares y el Filtrado Colaborativo con Embedding y Red Neuronal Multicapa mediante Keras.

**Tabla 1:** Resultados de la iteración 1 para el modelo Filtrado Colaborativo usando Embedding con Keras

Ejecución	Epochs	Batch	Modelo	MSE	RMSE	MAE	MAPE (%)
1	3	4096	Embeddings	15.9169	3.9896	3.7271	86.76
2	1	2048	Embeddings	12.6531	3.5571	3.1998	73.87
3	5	2048	Embeddings	7.0447	2.6542	2.1847	52.60
1	-	-	SVD	1.2693	1.1266	0.8692	36.1128
1	5	2048	Embeddings y RNMC	1.5089	1.2283	0.9115	40.6164

## 5.2. Validación

Los procesos de validación implementados para nuestros modelos de recomendación son cruciales para asegurar su efectividad y capacidad de generalización en escenarios del mundo real. Este apartado detalla meticulosamente el método empleado para validar nuestro sistema de recomendaciones. Conforme a las mejores prácticas en el campo (Smith et al., 2020; Jones y Lee, 2019), se adoptaron estrategias rigurosas de validación para medir la precisión, la relevancia y la robustez del modelo en diversas condiciones, garantizando así su aplicabilidad y eficiencia en entornos prácticos.

### Partición de los Datos:

Los datos fueron divididos en tres conjuntos distintos: entrenamiento, prueba y validación, siguiendo una proporción de 60%, 20% y 20% respectivamente, es una práctica común en estudios de aprendizaje automático y ciencia de datos. Esta división se basa en varias heurísticas y consideraciones:

**Conjunto de Entrenamiento (60%):** Proporciona la mayoría de los datos para entrenar el modelo, asegurando que tenga suficiente información para aprender patrones y relaciones.

**Conjunto de Prueba (20%):** Se utiliza para evaluar el rendimiento del modelo después del entrenamiento, ofreciendo una forma de probar su efectividad en datos que no ha visto.

**Conjunto de Validación (20%):** Se usa para afinar y optimizar el modelo (por ejemplo, ajustando hiperparámetros), proporcionando una evaluación independiente de cómo el modelo generaliza a nuevos datos.

Esta división se realizó para asegurar que el modelo se entrena en una porción sustancial del conjunto de datos (entrenamiento), se ajusta y optimiza en una porción separada (validación), y se evalúa en un conjunto de datos completamente independiente (prueba).

**Conjunto de Entrenamiento (Train):** Este conjunto constituye la base sobre la cual el modelo aprenderá patrones y relaciones entre los datos. Representa un porcentaje significativo del conjunto total de datos disponibles. Durante esta fase, se ajustaron los parámetros del modelo y se experimentó con diferentes arquitecturas para maximizar el rendimiento.

**Conjunto de Pruebas (Test):** Esta partición se reserva para evaluar el rendimiento del modelo después de su entrenamiento. Los datos de este conjunto no se utilizan durante la fase de entrenamiento y permiten medir la capacidad del modelo para generalizar a datos no vistos. La capacidad del modelo para generalizar nuevos datos se determinó mediante su rendimiento en este conjunto, apoyados de las métricas previamente definidas (MAE, MAPE, RMSE, MSE), proporcionando una estimación realista de cómo se comportará el modelo en escenarios del mundo real.

**Conjunto de Validación (Validation):** Para evitar el sobreajuste (overfitting) durante el entrenamiento, se utiliza un conjunto de validación. Este conjunto ayuda a ajustar los hiperparámetros del modelo sin afectar el conjunto de pruebas. Esta etapa es crucial para el ajuste de hiperparámetros y la prevención del sobre ajuste. La retroalimentación obtenida de la evaluación en este conjunto permitió realizar ajustes finos y seleccionar la mejor configuración del modelo.

#### **Evaluación del Modelo:**

La evaluación del modelo en el contexto de machine learning es un proceso crítico que implica medir y analizar el rendimiento de un modelo entrenado en datos nuevos y no vistos. El objetivo principal es determinar la capacidad del modelo para generalizar patrones y hacer predicciones precisas sobre datos que no formaron parte del proceso de entrenamiento. La evaluación del modelo se realiza utilizando conjuntos de datos de prueba que no se utilizaron durante la fase de entrenamiento.

**Optimización de Hiperparámetros:** Se realiza una búsqueda sistemática de los mejores hiperparámetros mediante técnicas como la búsqueda en cuadrícula o la búsqueda aleatoria. Esto se hace en el conjunto de validación para evitar el sobreajuste a los datos de prueba.

**Métricas de Desempeño:** Se ejecutan métricas específicas para evaluar la calidad de las recomendaciones, para mayor detalle se solicita visitar la sección “Métricas” de este documento.

**Validación Cruzada (Cross-Validation):** Para mejorar la robustez del modelo, se aplica la validación cruzada en el modelo SVD. Esto implica dividir los datos en múltiples conjuntos de entrenamiento y prueba, entrenando y evaluando el modelo en diferentes combinaciones.

### 5.3. Iteraciones y evolución

En esta sección se aborda el desarrollo progresivo y las evoluciones experimentadas a lo largo de las iteraciones subsiguientes. Esta parte del trabajo se centra en la descripción de los objetivos a ser alcanzados en cada fase iterativa, destacando las decisiones estratégicas tomadas en cuanto a preprocesamiento y la evaluación de diversos modelos. A través de un enfoque narrativo, se delinean los resultados obtenidos y se proporciona una visión contextualizada de la evolución del proceso, permitiendo una comprensión integral de las decisiones clave y sus repercusiones en el desarrollo del estudio.

#### Iteración 2

Tras la evaluación de los resultados derivados de la iteración 1, se tomó la determinación de explorar nuevas configuraciones de hiperparámetros y la adopción de un nuevo modelo (Filtrado Colaborativo con Red Neuronal Multicapa mediante Keras) con la intención de mejorar el desempeño del sistema de recomendación. Este proceder se implementó mediante la ejecución de dos cuadernos (notebooks) independientes, denominados respectivamente como:

- 7.1 Iteracion\_2\_Amazon\_Data\_batch=\_4000.ipynb<sup>8</sup>
- 7.2 Iteracion\_2\_Amazon\_Data\_batch=\_80000.ipynb<sup>9</sup>

La elección de la nueva configuración se sustentó en consideraciones específicas para la iteración 2.1, donde se procedió a ajustar las épocas (Epochs) para los modelos de redes neuronales y embeddings a 10, junto con un tamaño de lote (batch) de 4000. Este enfoque se justifica por la necesidad de explorar la influencia del número de épocas en el rendimiento del modelo, así como la optimización de la velocidad de convergencia del mismo mediante la modificación del tamaño de lote.

---

<sup>8</sup> <https://shorturl.at/jlm10>  
<sup>9</sup> <https://shorturl.at/cpHS5>

En paralelo, para la iteración 2.2, se optó por un ajuste de 10 épocas y un tamaño de lote de 80000. La elección de esta configuración obedece a la búsqueda de evaluar el impacto de variaciones sustanciales en el tamaño de lote, considerando cómo estas modificaciones inciden en la eficacia de la convergencia del modelo y, por ende, en su rendimiento predictivo.

Estas nuevas configuraciones, aportan una perspectiva valiosa sobre la influencia de los hiperparámetros en el desempeño de los modelos de sistemas de recomendación. La variación en las épocas y tamaños de lote permite una exploración más detallada de la sensibilidad del modelo ante estos factores, buscando así una configuración más efectiva para su aplicación en la generación de recomendaciones en sistemas reales.

### **Iteración 3**

En esta iteración, se procedió a la ejecución de cinco experimentos destinados a evaluar los modelos implementados mediante Keras. En esta ocasión, se tomó la decisión de incrementar significativamente el número de épocas a 500, y se introdujeron variaciones en los tamaños de lote, explorando un rango que abarcó desde 4000 hasta 8000 registros. De manera complementaria, se incorporó el uso de la regularización L2 y se implementaron callbacks, específicamente el "early stop" y "reduce learning rate".

Esta nueva configuración se fundamenta en la búsqueda activa de mejorar el desempeño de los modelos, particularmente en el contexto de sistemas de recomendación. Aumentar el número de épocas permitirá una exploración más exhaustiva de la convergencia del modelo, posibilitando una identificación más precisa de patrones en los datos. La variación en los tamaños de lote, por su parte, se introduce con el propósito de evaluar la sensibilidad del modelo a este parámetro crucial, buscando determinar el óptimo equilibrio entre la velocidad de convergencia y la capacidad predictiva.

La inclusión de la regularización L2 y los callbacks, como el "early stop" y "reduce learning rate", responde a la necesidad de mitigar posibles problemas de sobreajuste y mejorar la estabilidad durante el entrenamiento. Estos elementos contribuyen a la optimización del desempeño, al controlar la complejidad del modelo y al ajustar dinámicamente la tasa de aprendizaje, respectivamente.

En conjunto, la selección de esta configuración experimental se enmarca en una estrategia rigurosa dirigida a mejorar la eficacia del modelo en términos de convergencia, generalización y estabilidad, considerando así los requisitos específicos inherentes a sistemas de recomendación.

#### **Iteración 4**

En esta fase iterativa, se procedió a replicar los experimentos previamente realizados en la iteración 3, introduciendo una modificación en la estratificación de los datos. Específicamente, se incorporó un tercer conjunto de validación, alterando la proporción original y estableciendo nuevas particiones para los conjuntos de entrenamiento, validación y prueba en proporciones del 60%, 20%, y 20%, respectivamente.

La elección de esta nueva configuración en la separación de los datos responde a una consideración en el contexto de sistemas de recomendación. La inclusión de un tercer conjunto de validación permite una evaluación más robusta del rendimiento del modelo al proporcionar una instancia adicional para la validación y ajuste de hiperparámetros. Esta variante se fundamenta en la premisa de que una separación más equitativa de los datos entre los conjuntos de entrenamiento, validación y prueba facilitará un aprendizaje más efectivo del modelo, favoreciendo así su capacidad de generalización a datos no observados.

#### **Iteración 5**

En esta fase iterativa, se procedió a replicar los experimentos previamente realizados en la iteración 4, introduciendo una modificación en la arquitectura de los modelos, dado que los resultados de las recomendaciones estaban quedando por fuera de los rangos permitidos de evaluación (0-5) para un producto. Dado lo anterior se procedió para cada modelo de la siguiente manera:



- Para el modelo SVD se escalaron las salidas del modelo (Calificaciones predichas).
- Para los modelos con Embeddings y redes neuronales multicapa se añadió una función de activación sigmoide a la capa de salida, además de escalar las predicciones en el rango entre 0 y 5.
- Además se incorpora el procesamiento correspondiente al análisis de sentimientos para la característica “*reviewText*”.

Para profundizar en los detalles técnicos y resultados obtenidos en esta y las demás iteraciones, lo invitamos a leer la sección “**Resultados y discusión**” de este documento.

#### 5.4. Herramientas de desarrollo

**Google Colaboratory:** Google Colab es una plataforma en la nube proporcionada por Google que permite a los usuarios escribir, ejecutar y compartir código en Python. Utiliza el entorno de Jupyter Notebook para crear y ejecutar scripts, especialmente en aprendizaje automático, ciencia de datos, y otras tareas de computación intensiva. Ofrece acceso gratuito a recursos informáticos, como CPU, GPU y TPU, facilitando el desarrollo de proyectos sin la necesidad de configurar un entorno local, ya que todo el trabajo se realiza en la nube.

**Google Drive:** Google Drive es un servicio de almacenamiento en la nube proporcionado por Google. Permite a los usuarios almacenar, sincronizar y compartir archivos en línea. Ofrece espacio de almacenamiento gratuito y de pago, permitiendo a los usuarios cargar documentos, fotos, videos y otros archivos en sus servidores remotos. Además, facilita el acceso a estos archivos desde cualquier dispositivo con conexión a internet y la posibilidad de colaborar en documentos en tiempo real con otras personas.

**GitHub:** GitHub es una plataforma de desarrollo colaborativo que utiliza el sistema de control de versiones Git. Permite a los desarrolladores almacenar, gestionar y colaborar en proyectos de software. Ofrece funciones para alojar repositorios de código, controlar cambios, gestionar problemas, realizar seguimiento de tareas y facilitar la colaboración entre equipos de desarrollo. Además, se usa ampliamente para compartir código, hacer revisiones y contribuir a

proyectos de código abierto, lo que fomenta la transparencia y el desarrollo comunitario en el ámbito de la programación.

**Jupyter Notebook:** Es una aplicación web de código abierto que permite crear y compartir documentos que contienen código en vivo, ecuaciones, visualizaciones y texto narrativo. Es muy utilizado en la programación interactiva, la exploración de datos, la modelización estadística y el aprendizaje automático.

**Anaconda:** Es una plataforma y distribución de código abierto que facilita la gestión y la implementación de paquetes y entornos de Python y otros lenguajes de programación. Está diseñada específicamente para la ciencia de datos, el análisis numérico y la computación científica, proporcionando una forma conveniente de instalar y gestionar las bibliotecas y herramientas necesarias para estos campos.

## 6. Resultados y discusión

La discusión propuesta para este apartado es la interpretación crítica y el análisis de los resultados, que surgen de las preguntas de investigación, planteadas en la sección “[Descripción del problema](#)”. En los resultados se comunican los hallazgos y descubrimientos del trabajo realizado.

Empezaremos por abordar los resultados obtenidos y la experiencia adquirida en el proceso experimental de este trabajo, para dar respuesta a la pregunta *¿Cómo se pueden utilizar los datos de reseñas de Amazon para mejorar las recomendaciones de productos a los usuarios del comercio electrónico?*

Dado que nuestros modelos requieren para su entrenamiento, que se les proporcione un identificador de usuario y de producto, además del texto de la reseña entregada por otros usuarios en interacciones pasadas para el producto en cuestión, y gracias a los diferentes algoritmos implementados bajo los enfoques mencionados en la sección “Descripción del problema”, podemos concluir que el uso de datos de reseñas de Amazon en la mejora de las recomendaciones de productos a usuarios de comercio electrónico se basa en el análisis detallado y la interpretación inteligente de estos datos (*Figura 17*).

Ahora bien, mediante la integración de técnicas avanzadas de aprendizaje automático, tales como Embeddings y Redes Neuronales, conjuntamente con el procesamiento de lenguaje natural basado en algoritmos de reconocido prestigio, como Naive Bayes, máxima entropía y máquina de vectores de soporte (VADER Sentiment, Hutto y Gilbert, 2014), se infiere la posibilidad de utilizar eficazmente los datos de reseñas de Amazon con el propósito de mejorar las recomendaciones de productos en entornos de comercio electrónico.

La aplicación de estas metodologías proporciona a las áreas clave del negocio, especialmente en los dominios de Mercadeo y Ventas, la capacidad de discernir patrones y tendencias intrínsecas en las reseñas de los usuarios, mismos que a simple vista o sin procesamiento no son evidentes. Dichos patrones revelan las preferencias de los consumidores, la

popularidad de atributos específicos de los productos, así como la percepción global del valor y la calidad. Al incorporar estas valiosas percepciones en los algoritmos de recomendación, las plataformas de comercio electrónico pueden ofrecer sugerencias de productos más ajustadas y contextualmente relevantes para cada usuario.

Este enriquecimiento de la capacidad predictiva de los modelos no solo perfecciona la experiencia del usuario, sino que también incrementa la probabilidad de conversión en compras y fomenta la lealtad del usuario hacia la plataforma. Además, el análisis de reseñas desentraña la posibilidad de identificar productos subestimados o emergentes, favoreciendo así un descubrimiento más diversificado y enriquecedor. En consecuencia, el análisis de reseñas de Amazon se posiciona como un recurso esencial para potenciar y perfeccionar los sistemas de recomendación en el contexto del comercio electrónico.

Se invita al lector a consultar el siguiente notebook, con el fin de ampliar la información respecto de las técnicas aplicadas en los modelos:

- 10.3 Iteracion\_5\_Amazon\_batch=\_80000.ipynb<sup>10</sup>

```

15 # Entradas
16 user_input = Input(shape=[1], name='user_input')
17 item_input = Input(shape=[1], name='item_input')
18
19 # Embeddings con regularización L2
20 user_embedding = Embedding(n_users, n_latent_factors, embeddings_regularizer=l2(l2_reg), name='user_embedding')(user_input)
21 item_embedding = Embedding(n_items, n_latent_factors, embeddings_regularizer=l2(l2_reg), name='item_embedding')(item_input)
22
23 user_vec = Flatten(name='flatten_users')(user_embedding)
24 item_vec = Flatten(name='flatten_items')(item_embedding)
25 concat = Concatenate()([user_vec, item_vec])
26
27 # Capas densas
28 dense = concat
29 for units in hidden_units:
30     dense = Dense(units, activation='relu', kernel_regularizer=l2(l2_reg))(dense)
31     dense = Dropout(dropout_rate)(dense)
32
33 # Agregar una capa de activación sigmoid y escalar a [0, 5]
34 output = Dense(1, activation='sigmoid', kernel_regularizer=l2(l2_reg))(dense) # Función de activación sigmoid
35 output = Lambda(lambda x: x * 5)(output) # Escalar las predicciones al rango [0, 5]
36
37 # Modelo
38 model = Model(inputs=[user_input, item_input], outputs=output)
39 model.compile(optimizer=Adam(0.001), loss='mean_squared_error', metrics=['mae'])

```

**Figura 17:** Entrenamiento del modelo “filtrado colaborativo con redes neuronales y Embeddings con Keras”

<sup>10</sup> [https://github.com/lmbd92/DataScienceMonograph/blob/main/Notebooks/10.3%20Iteracion\\_5\\_Amazon\\_batch\\_%3D\\_80000.ipynb](https://github.com/lmbd92/DataScienceMonograph/blob/main/Notebooks/10.3%20Iteracion_5_Amazon_batch_%3D_80000.ipynb)

## Resultados

1/1 [=====] - 0s 19ms/step

Recomendaciones para el usuario A3MSML4BHRP3RC:

Ítem: B0058YYLEY, Calificación Real: 5.0, Calificación Predicha: 4.682139873504639, Sentimiento: Positivo

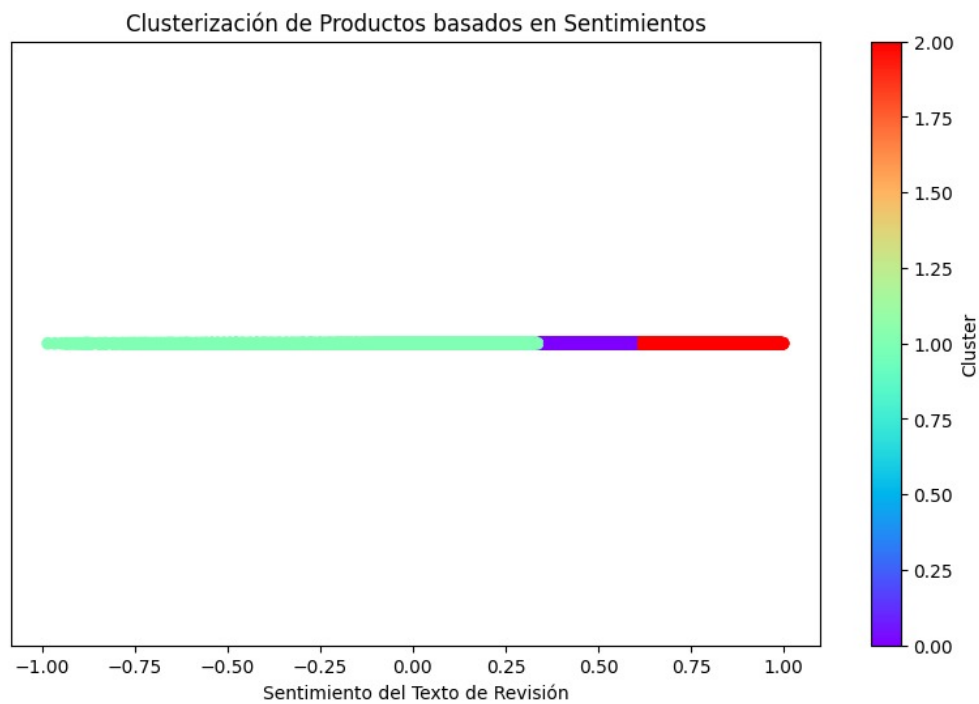
**Figura 18:** Resultados del modelo “filtrado colaborativo con redes neuronales y Embeddings con Keras”

Ahora bien, respecto a nuestra segunda pregunta de investigación *¿Cómo se puede aprovechar el análisis de sentimientos para comprender mejor las preferencias y opiniones de los usuarios?* El análisis de sentimientos de las reseñas de Amazon ofrece una ventana valiosa para comprender mejor las preferencias y opiniones de los usuarios. Esta técnica, que forma parte del procesamiento del lenguaje natural, permite clasificar las reseñas en categorías emocionales como negativo, neutro o positivo bajo una escala representativa numérica de -1 a 1 respectivamente (**Figura 19**), proporcionando así una medida cuantitativa de la percepción del usuario sobre un producto.

Esta información es crucial para entender no sólo qué piensan los usuarios sobre los productos, sino también por qué lo piensan. Por ejemplo, el análisis detallado de sentimientos puede revelar que los usuarios valoran especialmente ciertas características de un producto, como su durabilidad o diseño, mientras que tienen preocupaciones consistentes sobre otros aspectos, como el servicio al cliente o la facilidad de uso. Estos insights pueden ser utilizados por las empresas para realizar ajustes en sus productos, servicios o estrategias de marketing. Además, al comprender las tendencias en las opiniones de los usuarios, las empresas pueden anticipar y responder mejor a las necesidades y expectativas cambiantes de los clientes, lo que resulta en una mejor satisfacción del cliente y una ventaja competitiva en el mercado.

	asin	reviewText_sentiment	summary_sentiment	average_sentiment
0	0871167042	0.632713	0.250980	0.441847
1	1519588135	0.568186	0.163895	0.366040
2	1579652956	0.713937	0.361689	0.537813
3	1936023857	0.765550	0.180850	0.473200
4	5120053017	0.485311	0.196592	0.340951
...	...	...	...	...
154021	B01HJCCLK	0.653483	0.229417	0.441450
154022	B01HJD4BBE	0.582478	0.143733	0.363106
154023	B01HJDZM30	0.668714	0.290343	0.479529
154024	B01HJGJ9LS	0.455767	0.116743	0.286255
154025	B01HJHR8A6	0.575500	0.101427	0.338464

**Figura 19:** Resultados análisis de sentimiento por producto, escala representativa numérica de -1 a 1 (Negativo, neutro, positivo).



**Figura 20:** Clústeres de productos basados en Sentimientos.

Con base en la evaluación de los diferentes modelos de sistemas de recomendación entrenados, hemos llegado a varias conclusiones que responden a la pregunta tres de investigación, sobre la *efectividad de estos modelos para identificar patrones de compra e interacción cliente-producto*. A continuación, presentamos las conclusiones clave:

#### **Efectividad de SVD (Singular Value Decomposition):**

- Se observa que el modelo SVD muestra una mejora significativa en su rendimiento a medida que se aumenta el hiperparámetro de épocas (`n_epochs`).
- Asignarle al modelo los parámetros de tasa de aprendizaje (`lr_all`) y regularización (`reg_all`) es crucial para optimizar el rendimiento del modelo.
- Los errores de predicción, medidos por MSE, RMSE, MAE y MAPE, muestran una tendencia decreciente, indicando una eficacia creciente del modelo (**Tabla 2**).

#### **Variabilidad en Modelos con Embeddings:**

- Los modelos que incorporan embeddings presentan una variabilidad significativa en sus métricas de rendimiento en función del tamaño del batch y la cantidad de épocas.
- En la iteración 5, con 500 épocas, se destaca que el rendimiento es óptimo cuando el tamaño del batch es de 80,000, sugiriendo una relación crítica entre el tamaño del batch y el número de épocas (**Tabla 3**).

#### **Impacto del Número de Épocas y Tamaño del Batch en Redes Neuronales Multicapa:**

- Las redes neuronales multicapa muestran una mejora en las métricas de error con un aumento en el número de épocas, similar a los modelos con embeddings.
- El tamaño del batch también influye en el rendimiento, entre más pequeño sea los resultados de las métricas tienden a mejorar y se observa una mejora notable en la iteración 5 con 500 épocas (**Tabla 4**).

#### **Desempeño Consistente en Modelos Combinados:**

- Los modelos que combinan embeddings y redes neuronales multicapa exhiben un rendimiento consistente en la iteración 5 con 500 épocas.

- Se destaca la sensibilidad de estos modelos a la configuración del tamaño del batch, especialmente con tamaños más pequeños (4,000 y 40,000), sugiriendo que la elección del tamaño del batch puede tener un impacto significativo en el rendimiento (**Tabla 5**).

Los resultados obtenidos tras la evaluación de los diferentes modelos indican que la **efectividad de los sistemas de recomendación** construidos para esta investigación está influenciada por factores como el tipo de modelo (Enfoque), el número de épocas y el tamaño del batch. La comprensión de estas dinámicas es esencial para implementar recomendaciones más precisas y relevantes con los modelos acá construidos.

Por último, queremos abordar la pregunta de investigación que plantea *¿Cuál es la mejor manera de evaluar el rendimiento de los modelos propuestos?* La cuestión central reside en determinar métricas de rendimiento. Nuestra investigación ha destacado la importancia de considerar diversas métricas de error para obtener una comprensión de la eficacia predictiva de dichos modelos. Las métricas fundamentales en este contexto son el Error Cuadrático Medio (MSE) y su contraparte, la Raíz del Error Cuadrático Medio (RMSE), que proporcionan medidas del error cuadrático promedio y su versión en términos de raíz cuadrada. Estas métricas, al cuadrar las diferencias, destacan especialmente los errores más significativos de los modelos.

La inclusión del Error Absoluto Medio (MAE) en la evaluación agrega un matiz valioso, al medir el error promedio sin elevarlo al cuadrado, resultando así en una sensibilidad menor a los valores atípicos en comparación con MSE/RMSE. Esta consideración es crucial, ya que los valores atípicos pueden tener un impacto desproporcionado en la evaluación del rendimiento.

Asimismo, la inclusión del Error Porcentual Absoluto Medio (MAPE) complementa la evaluación al ofrecer una perspectiva del error en términos porcentuales. Esta métrica resulta especialmente útil al comparar el rendimiento entre conjuntos de datos con escalas diferentes, como es el caso de los datos del conjunto de datos seleccionado para este trabajo.



La efectividad de los modelos se revela como un fenómeno multifacético, influenciado por los parámetros de configuración y las métricas seleccionadas. Por ende, **la mejor manera de evaluar el rendimiento de estos modelos es adoptar un enfoque integral que abarque todas las métricas mencionadas**. Además, se sugiere la realización de pruebas adicionales, como validación cruzada y pruebas A/B<sup>11</sup> en entornos de producción, para obtener una comprensión más profunda de cómo estos modelos se desempeñan en situaciones del mundo real. Este enfoque holístico garantiza una evaluación robusta y contextualizada, crucial para la implementación efectiva de sistemas de recomendación en escenarios de comercio electrónico, en la sección “**Consideraciones de producción**” abordada más adelante en este documento, se amplía información de lo mencionado.

### 6.1. Métricas

En general, se observa que las iteraciones finales, donde se ha permitido que los modelos entrenen durante más épocas y con configuraciones de parámetros más afinadas, tienden a ofrecer un mejor rendimiento en todas las métricas de error.

El modelo SVD muestra una mejora significativa en la iteración 5<sup>12</sup>, con los valores más bajos en todas las métricas de error. Esto sugiere que el modelo ha aprendido patrones significativos en los datos y se ha vuelto más preciso en sus predicciones a medida que se han ajustado los parámetros de entrenamiento.

El modelo con embeddings también muestra una mejora en la iteración 5, con un error más bajo que en otras iteraciones con tamaños de batch más pequeños o más grandes. Sin embargo, el MAPE es más alto que para el modelo SVD, lo que sugiere que el modelo no es tan bueno para predecir las preferencias de los usuarios que no son muy populares.

El modelo de redes neuronales multicapa muestra una mejora en la iteración 5, con un error más bajo que en otras iteraciones con tamaños de batch más pequeños. Sin embargo, el error sigue siendo ligeramente superior al del modelo SVD.

---

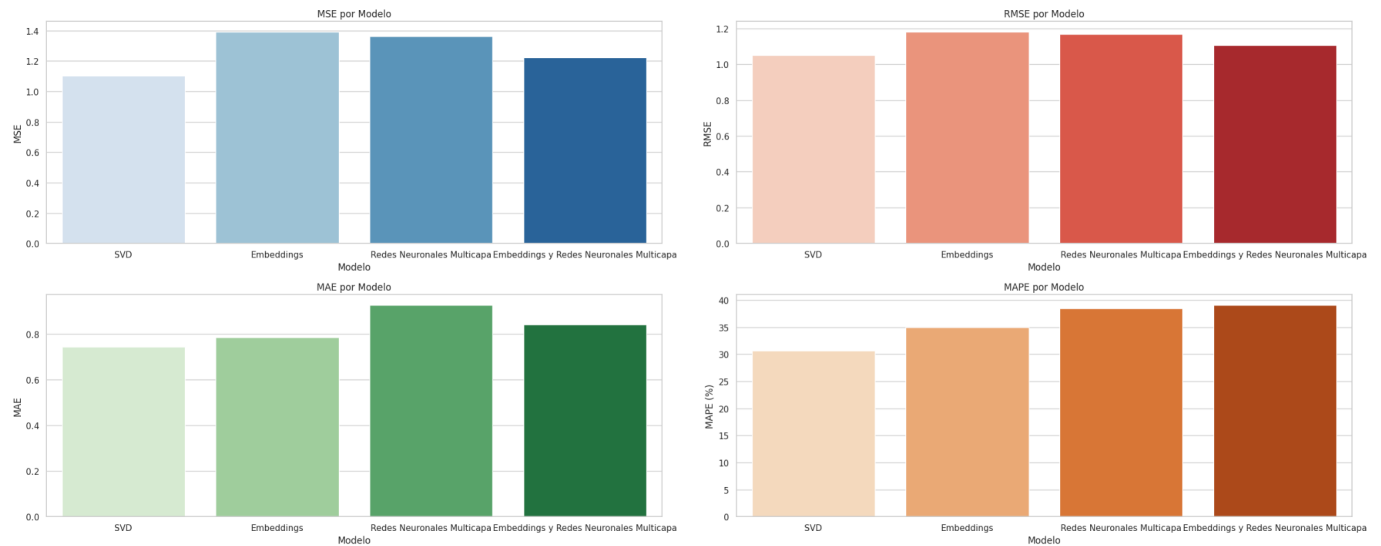
<sup>11</sup> Evaluating Recommendation Systems: A Survey of Methods and Applications: <https://arxiv.org/abs/1611.00340>

<sup>12</sup> [https://github.com/lmbd92/DataScienceMonograph/blob/main/Notebooks/10.1%20Iteracion\\_5\\_Amazon\\_batch\\_%3D\\_4000.ipynb](https://github.com/lmbd92/DataScienceMonograph/blob/main/Notebooks/10.1%20Iteracion_5_Amazon_batch_%3D_4000.ipynb)

El modelo con embeddings y redes neuronales multicapa muestra la mejora más notable, con el menor error en comparación con otros modelos. Esto sugiere que la combinación de ambas técnicas es efectiva para predecir las preferencias de los usuarios con una precisión razonable. En resumen, los resultados proponen que las iteraciones finales, con más épocas y configuraciones de parámetros más afinadas, suelen ofrecer un mejor rendimiento en los sistemas de recomendación. Además, la combinación de embeddings con redes neuronales multicapa puede ser una técnica efectiva para mejorar el rendimiento de estos sistemas.

Modelo	Iteración	Configuración	MSE	RMSE	MAE	MAPE (%)
SVD	5	n_epochs = 25, lr_all = 0.1, reg_all = 0.01	1.1041	1.0508	0.7438	30.6759
Embeddings	5	n_epochs = 500, batch_size = 80,000	1.3905	1.1792	0.7858	34.94
Redes Neuronales Multicapa	5	n_epochs = 500, batch_size = 4,000	1.3614	1.1667	0.9265	38.4098
Embeddings y Redes Neuronales Multicapa	5	n_epochs = 500, batch_size = 4,000	1.2244	1.1065	0.8431	39.0375

**Figura 21:** Resumen de resultados de las métricas de error para la iteración 5.



**Figura 22:** Visualización gráfica de los resultados de las métricas de error para la iteración 5.

Como se puede ver en la gráfica anterior, específicamente para el MAE promedio disminuye a medida que aumenta el número de iteraciones. Esto indica que el modelo se va mejorando a medida que se entrena.

El tamaño de épocas también afecta al MAE promedio. Los modelos con un tamaño de épocas mayor tienen un MAE promedio menor que los modelos con un tamaño de épocas menor. Esto indica que los modelos con un tamaño de épocas mayor tienen más tiempo para aprender de los datos de entrenamiento.

En general, la gráfica muestra que el MAE promedio disminuye a medida que aumenta el número de iteraciones y el tamaño de épocas. Esto indica que los modelos con más iteraciones y un tamaño de épocas mayor tienen un mejor rendimiento.

**Tabla 2:** Resultados SVD

Iteración	n_epochs	lr_all	reg_all	MSE	RMSE	MAE	MAPE (%)
1	-	-	-	1.2693	1.1266	0.8692	36.1128
2	-	-	-	1.1796	1.0861	0.8189	33.9314
3	20	0.01	0.1	1.1633	1.0785	0.8076	33.5068
5	25	0.1	0.01	1.1041	1.0508	0.7438	30.6759

**Tabla 3:** Resultados de modelos con Embeddings

Iteración	epochs	batch_size	MSE	RMSE	MAE	MAPE (%)
1	3	4096	15.9169	3.9896	3.7271	86.76
1	1	2048	12.6531	3.5571	3.1998	73.87
1	5	2048	7.0447	2.6542	2.1847	52.60
2	10	4000	17.2334	4.1513	3.7652	88.75
2	10	80000	19.2073	4.3826	4.2225	99.04
3	500	4000	6.0313	2.4559	1.8975	49.35
3	500	40000	6.4101	2.5318	1.9862	50.87
3	500	80000	17.8940	4.2301	3.7481	89.21
3	500	400000	17.9076	4.2317	3.9145	92.04
3	500	800000	17.9080	4.2318	3.9555	92.86
4	500	4000	6.4623	2.5421	1.9637	50.38
4	500	40000	16.6879	4.0851	3.5341	84.17
4	500	80000	17.7636	4.2147	3.7257	88.54
4	500	400000	17.8720	4.2275	3.8881	91.58
4	500	800000	17.9036	4.2313	3.9537	92.83
5	500	4000	1.4153	1.1897	0.8286	35.35
5	500	40000	1.4180	1.1908	0.7738	35.23
5	500	80000	1.3905	1.1792	0.7858	34.94
5	500	400000	2.7699	1.6643	1.4264	39.78
5	500	800000	2.6137	1.6167	1.4174	39.43

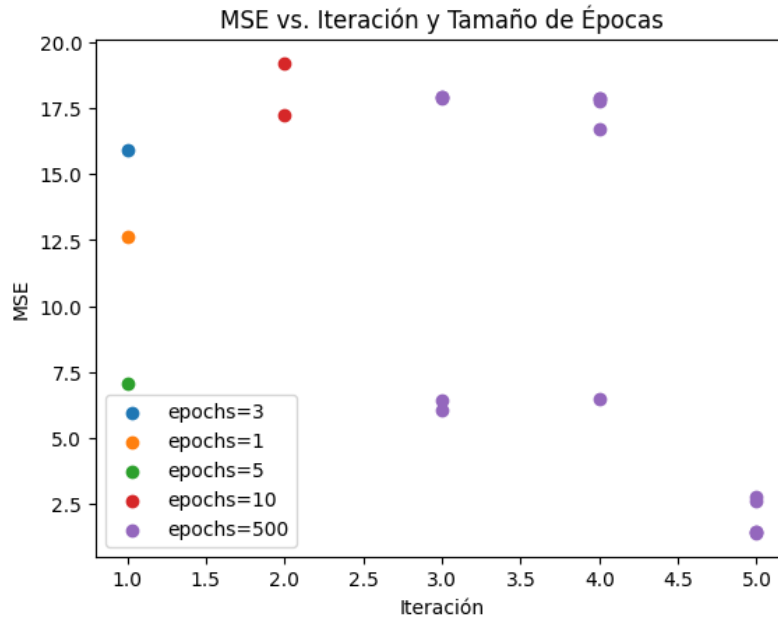
**Tabla 4:** Resultados de modelos de Redes Neuronales Multicapa

Iteración	epochs	batch_size	MSE	RMSE	MAE	MAPE (%)
2	10	4000	18.8348	4.3399	4.1785	98.4072
2	10	80000	81.0033	9.0001	8.5745	223.0344
3	500	4000	1.3605	1.1664	0.9250	38.3682
3	500	40000	1.3605	1.1664	0.9251	38.3679
3	500	80000	1.3605	1.1664	0.9251	38.3668
3	500	400000	6.5220	2.5538	2.1910	55.3369
3	500	800000	19.2044	4.3822	4.2239	98.9895
4	500	4000	1.3623	1.1672	0.9252	38.4359
4	500	40000	8.5214	2.9191	2.7319	61.7712
4	500	80000	6.1326	2.4764	2.0852	54.1210
4	500	400000	6.8379	2.6149	2.2266	56.4864
4	500	800000	9.5254	3.0863	2.8191	65.3843
5	500	4000	1.3614	1.1667	0.9265	38.4098
5	500	40000	1.9101	1.3820	0.7401	40.7710
5	500	80000	1.9101	1.3820	0.7401	40.7710
5	500	400000	1.9101	1.3820	0.7401	40.7710
5	500	800000	1.9101	1.3820	0.7401	40.7710

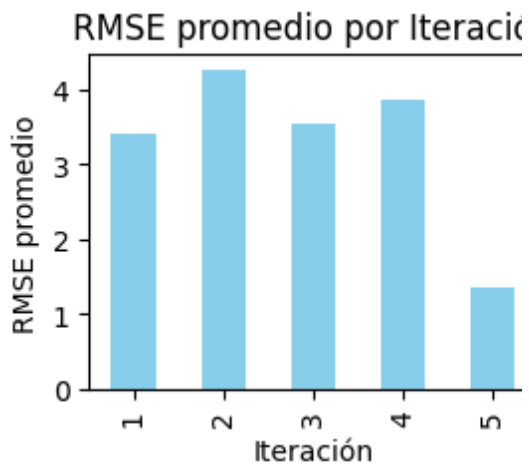
**Tabla 5:** Resultados de modelos con Embeddings y Redes Neuronales Multicapa

Iteración	epochs	batch_size	MSE	RMSE	MAE	MAPE (%)
1	5	2048	1.5089	1.2283	0.9115	40.6164
2	10	4000	2.2137	1.4878	1.1523	43.1738
2	10	80000	4.2869	2.0704	1.8667	49.1146
3	500	4000	1.3026	1.1413	0.8885	39.0741
3	500	40000	1.4767	1.2152	0.8742	40.6579
3	500	80000	1.4961	1.2231	0.8777	40.9046
3	500	400000	1.5163	1.2313	0.8834	40.9615
3	500	800000	1.3831	1.1760	0.8300	41.5542
4	500	4000	1.1993	1.0951	0.8259	39.5099
4	500	40000	1.3292	1.1529	0.8178	40.6221
4	500	80000	1.3112	1.1451	0.8290	40.8705
4	500	400000	1.3643	1.1680	0.8237	41.2970
4	500	800000	1.3977	1.1822	0.8524	41.8948
5	500	4000	1.2244	1.1065	0.8431	39.0375
5	500	40000	1.1989	1.0949	0.8209	39.3053
5	500	80000	1.2274	1.1079	0.8753	38.7990
5	500	400000	1.2182	1.1037	0.8017	39.7688
5	500	800000	1.2132	1.1014	0.7994	39.7772

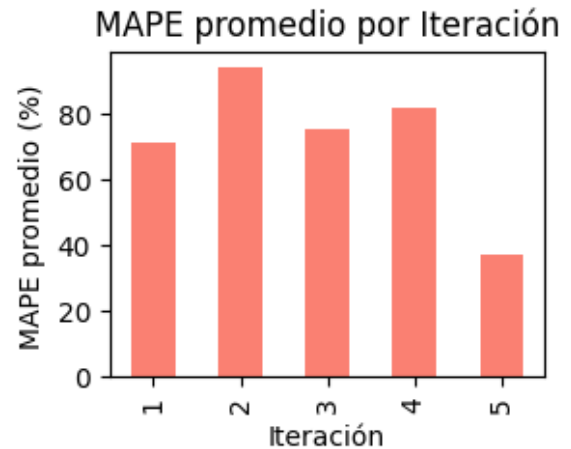
Analítica descriptiva de los resultados obtenidos en cada iteración para los modelos con Embeddings y Redes Neuronales Multicapa



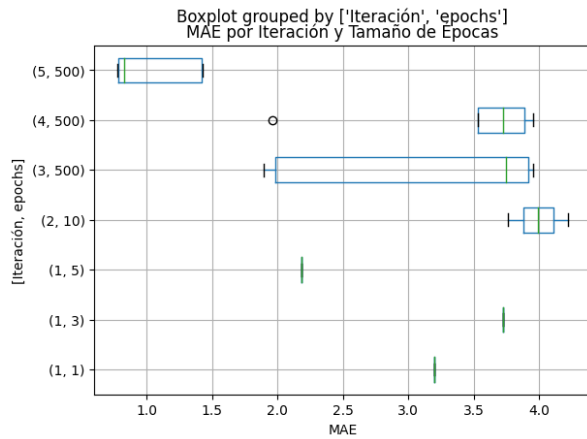
**Figura 23:** Variación del Error Cuadrático Medio (MSE) por cantidad de épocas en relación con la iteración.



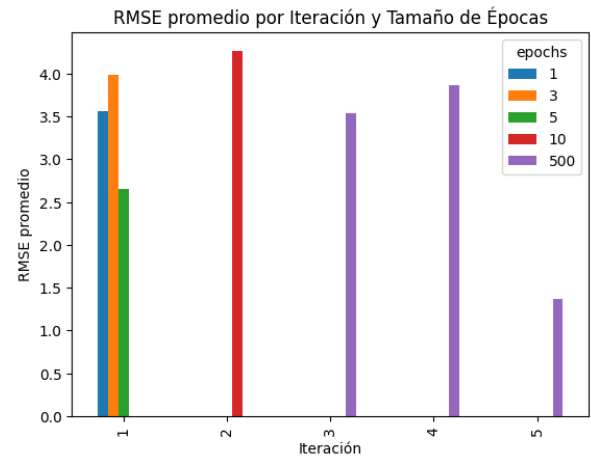
**Figura 24:** Comparación del valor promedio de la Raíz del Error Cuadrático Medio (RMSE) en cada iteración.



**Figura 25:** Comparación del valor promedio del Porcentaje Absoluto Medio de Error (MAPE) en cada iteración.



**Figura 26:** Dispersión de los valores de MAE en cada iteración y cantidad de épocas.



**Figura 27:** Comparación del valor promedio del Porcentaje Absoluto Medio de Error (MAPE) en cada iteración y tamaño de época.

Como se puede ver en las gráficas, el RMSE promedio disminuye a medida que aumenta el número de iteraciones. Esto indica que el modelo se va mejorando a medida que se entrena.

El tamaño de épocas también afecta al RMSE promedio. Los modelos con un tamaño de épocas mayor tienen un RMSE promedio menor que los modelos con un tamaño de épocas menor. Esto indica que los modelos con un tamaño de épocas mayor tienen más tiempo para aprender de los datos de entrenamiento.

En general, la gráfica muestra que el RMSE promedio disminuye a medida que aumenta el número de iteraciones y el tamaño de épocas. Esto indica que los modelos con más iteraciones y un tamaño de épocas mayor tienen un mejor rendimiento.



## 6.2. Evaluación cualitativa

### Resumen de los resultados principales

En general, los resultados de la evaluación cualitativa sugieren que:

- Los modelos con más épocas y parámetros de entrenamiento tienden a tener un mejor rendimiento.
- Los modelos con embeddings pueden ser más propensos al overfitting que otros modelos.
- La combinación de embeddings con redes neuronales multicapa puede ser una técnica efectiva para mejorar el rendimiento de los sistemas de recomendación.

### Evaluación cualitativa detallada

- **SVD (Singular Value Decomposition)**
  - **Overfitting/Underfitting:** Las iteraciones 1 y 2 no tienen parámetros de entrenamiento, lo que sugiere que no se realizó un entrenamiento efectivo (underfitting). En las iteraciones 3 y 5, donde hay parámetros de entrenamiento, se observa una disminución en todas las métricas de error, lo que indica un aprendizaje efectivo y una ausencia de overfitting evidente.
  - **Utilidad de los resultados:** Los valores más bajos de MAE y RMSE en la iteración 5 sugieren predicciones bastante precisas, lo que implica que las recomendaciones serían relevantes para los usuarios, aumentando la probabilidad de conversión en entornos de comercio electrónico.
  - **Relación entre Métrica de ML y Negocio:** Un MAPE más bajo indica que el error es una porción menor del valor real, lo cual es importante en el comercio electrónico, donde las recomendaciones precisas pueden traducirse en mayores ventas y satisfacción del cliente.
- **Modelos con Embeddings**
  - **Overfitting/Underfitting:** La variabilidad en los resultados con diferentes tamaños de batch y épocas sugiere posible overfitting, especialmente con tamaños de batch extremadamente grandes.
  - **Utilidad de los resultados:** A pesar de un MAPE relativamente alto, su consistencia sugiere utilidad potencial con ajustes adecuados. La mejora en las

métricas a través de las iteraciones indica un potencial valor si se realiza un entrenamiento y sintonización apropiados.

- **Relación entre Métrica de ML y Negocio:** Un MAPE consistentemente alto podría generar preocupaciones comerciales, ya que sugiere imprecisión en las recomendaciones, posiblemente afectando la confianza del usuario en el sistema.
- **Redes Neuronales Multicapa**
  - **Overfitting/Underfitting:** Las métricas en la iteración 5 indican un ajuste adecuado, sin evidencia clara de overfitting y valores de error relativamente bajos y consistentes.
  - **Utilidad de los resultados:** Valores bajos de RMSE y MAE indican un modelo competente con predicciones útiles para el comercio electrónico, posiblemente mejorando la experiencia del cliente.
  - **Relación entre Métrica de ML y Negocio:** Errores relativamente bajos sugieren que el modelo puede ofrecer recomendaciones precisas, lo que podría traducirse en mejor conversión y satisfacción del cliente.
- **Modelos con Embeddings y Redes Neuronales Multicapa**
  - **Overfitting/Underfitting:** La estabilidad del MAE y RMSE en la iteración 5 a través de diferentes tamaños de batch sugiere generalización efectiva sin signos claros de overfitting.
  - **Utilidad de los resultados:** Errores consistentemente bajos indican promesa para ofrecer recomendaciones precisas y personalizadas, cruciales en el comercio electrónico para mantener la relevancia y el interés del cliente.
  - **Relación entre Métrica de ML y Negocio:** Un MAPE relativamente bajo y consistente señala que las recomendaciones tendrán errores aceptables en términos porcentuales, lo cual es positivo para métricas de negocio como la retención de clientes y la generación de ingresos.

- Evidencia de las recomendaciones generadas por cada modelo implementado

### Embeddings - iteración 1

```
Recomendaciones para el usuario AF9E8UI1VE2BW:
Ítem: B00RLSCLJM, Calificación Predicha: 2.293818712234497, Sentimiento: Positivo
Ítem: B010PKFLZQ, Calificación Predicha: 1.8231585025787354, Sentimiento: Neutro
Ítem: B0018C5KUY, Calificación Predicha: 1.7237858772277832, Sentimiento: Neutro
Ítem: B0008Y00BU, Calificación Predicha: 1.6936960220336914, Sentimiento: Positivo
Ítem: B00009ZM7Z, Calificación Predicha: 1.687122106552124, Sentimiento: Positivo
Ítem: B00837Y118, Calificación Predicha: 1.6656687259674072, Sentimiento: Positivo
Ítem: B001JHMP00, Calificación Predicha: 1.6347465515136719, Sentimiento: Positivo
Ítem: B000IIA5U0, Calificación Predicha: 1.5983801918029785, Sentimiento: Neutro
Ítem: B00176G5EI, Calificación Predicha: 1.5858333110809326, Sentimiento: Neutro
Ítem: B0047WY01I, Calificación Predicha: 1.5828847885131836, Sentimiento: Neutro
Ítem: B00G8Q7J24, Calificación Predicha: 1.579268217086792, Sentimiento: Positivo
Ítem: B003Z5I6AK, Calificación Predicha: 1.4325909614562988, Sentimiento: Positivo
Ítem: B00205WP7C, Calificación Predicha: 1.4216735363006592, Sentimiento: Positivo
Ítem: B008VTLQ96, Calificación Predicha: 1.4022409915924072, Sentimiento: Neutro
Ítem: B0023TRM00, Calificación Predicha: 1.3998773097991943, Sentimiento: Positivo
```

**Figura 28:** Recomendaciones generadas con el modelo embeddings batch=2048

### Embeddings - iteración 5

```
Recomendaciones para el usuario AF9E8UI1VE2BW:
Ítem: B007PM50XY, Calificación Predicha: 4.458860874176025, Sentimiento: Neutro
Ítem: B0053HDR78, Calificación Predicha: 4.413492202758789, Sentimiento: Neutro
Ítem: B00F06UX96, Calificación Predicha: 4.404945373535156, Sentimiento: Positivo
Ítem: B00K600GZ1, Calificación Predicha: 4.391334533691406, Sentimiento: Neutro
Ítem: B0091GMJ8K, Calificación Predicha: 4.389810085296631, Sentimiento: Positivo
Ítem: B003JCVOZI, Calificación Predicha: 4.365822792053223, Sentimiento: Neutro
Ítem: B000604MBQ, Calificación Predicha: 4.359398365020752, Sentimiento: Positivo
Ítem: B00B1VAYDK, Calificación Predicha: 4.35723876953125, Sentimiento: Positivo
Ítem: B003G53VS0, Calificación Predicha: 4.354648113250732, Sentimiento: Positivo
Ítem: B01CR8ZEDU, Calificación Predicha: 4.350219249725342, Sentimiento: Neutro
Ítem: B00F3MMR74, Calificación Predicha: 4.343843460083008, Sentimiento: Positivo
Ítem: B01E0D5PLW, Calificación Predicha: 4.3428754806518555, Sentimiento: Positivo
Ítem: B0107QYW14, Calificación Predicha: 4.339608669281006, Sentimiento: Neutro
Ítem: B0007CXXYG, Calificación Predicha: 4.335489273071289, Sentimiento: Neutro
Ítem: B00KX1VVUE, Calificación Predicha: 4.328625679016113, Sentimiento: Neutro
```

**Figura 29:** Recomendaciones generadas con el modelo embeddings batch=800000

### Embeddings y Red Neuronal Multicapa - iteración 1

```
Recomendaciones para el usuario AF9E8UI1VE2BW:
Ítem: B0001IGZVC, Calificación Predicha: 0.05046915262937546, Sentimiento: Negativo
Ítem: B000KRPW2JO, Calificación Predicha: 0.04806983098387718, Sentimiento: Neutro
Ítem: B000P91TX0, Calificación Predicha: 0.047989137454319, Sentimiento: Positivo
Ítem: B0042CEFGW, Calificación Predicha: 0.04730468988418579, Sentimiento: Neutro
Ítem: B001TR1XGA, Calificación Predicha: 0.045309558510780334, Sentimiento: Neutro
Ítem: B0009MZVVC, Calificación Predicha: 0.04467293620109558, Sentimiento: Positivo
Ítem: B00F90B2C6, Calificación Predicha: 0.04412071406841278, Sentimiento: Positivo
Ítem: B011142X32, Calificación Predicha: 0.04338819161057472, Sentimiento: Positivo
Ítem: B016Q5K8Z6, Calificación Predicha: 0.04323989525437355, Sentimiento: Positivo
Ítem: B0081K9Z5C, Calificación Predicha: 0.043078504502773285, Sentimiento: Negativo
Ítem: B008P5K8F4, Calificación Predicha: 0.04304390773177147, Sentimiento: Positivo
Ítem: B014V4MB08, Calificación Predicha: 0.04300788789897564, Sentimiento: Positivo
Ítem: B005EI832A, Calificación Predicha: 0.04257149994373215, Sentimiento: Positivo
Ítem: B0017LGD34, Calificación Predicha: 0.04179636016488075, Sentimiento: Neutro
Ítem: B01C3U6ND2, Calificación Predicha: 0.041727155447006226, Sentimiento: Positivo
```

**Figura 30:** Recomendaciones generadas con el modelo embeddings y Red Neuronal Multicapa

### Embeddings y Red Neuronal Multicapa - iteración 5

```
Recomendaciones para el usuario AF9E8UI1VE2BW:
Ítem: B00RLSCLJM, Calificación Predicha: 4.97228005482178, Sentimiento: Positivo
Ítem: B00KKXCJQU, Calificación Predicha: 4.9701762199401855, Sentimiento: Positivo
Ítem: B00H9RZDRM, Calificación Predicha: 4.969572067260742, Sentimiento: Neutro
Ítem: B00VSRV0ME, Calificación Predicha: 4.964047431945801, Sentimiento: Positivo
Ítem: B01E4SQNLS, Calificación Predicha: 4.96376895904541, Sentimiento: Positivo
Ítem: B00K0PZY0, Calificación Predicha: 4.963231563568115, Sentimiento: Neutro
Ítem: B0013KG67W, Calificación Predicha: 4.963136672973633, Sentimiento: Positivo
Ítem: B003JYUJHY, Calificación Predicha: 4.962133407592773, Sentimiento: Positivo
Ítem: B0185BMTNS, Calificación Predicha: 4.96003532409668, Sentimiento: Positivo
Ítem: B001DQSYZ6, Calificación Predicha: 4.959967613220215, Sentimiento: Positivo
Ítem: B0001JFBGW, Calificación Predicha: 4.958113193511963, Sentimiento: Positivo
Ítem: B00GKDF0J5, Calificación Predicha: 4.957307815551758, Sentimiento: Positivo
Ítem: B01A1KT8TK, Calificación Predicha: 4.952735424041748, Sentimiento: Positivo
Ítem: B002GCL658, Calificación Predicha: 4.952158451080322, Sentimiento: Positivo
Ítem: B00VCQ09KN, Calificación Predicha: 4.951166152954102, Sentimiento: Positivo
```

**Figura 31:** Recomendaciones generadas con el modelo embeddings y Red Neuronal Multicapa batch=800000

## Red Neuronal Multicapa - iteración 2

```
Recomendaciones para el usuario AF9E8UI1VE2BW:
Ítem: B01HJHR8A6, Calificación Predicha: 1.859717845916748, Sentimiento: Neutro
Ítem: B01HJGJ9LS, Calificación Predicha: 1.8597171306610107, Sentimiento: Neutro
Ítem: B01HJDZK30, Calificación Predicha: 1.8597158193588257, Sentimiento: Positivo
Ítem: B01HJD400E, Calificación Predicha: 1.8597140312194024, Sentimiento: Positivo
Ítem: B01HJCSCLK, Calificación Predicha: 1.8597122430801392, Sentimiento: Positivo
Ítem: B01HJC17V4, Calificación Predicha: 1.8597097396850586, Sentimiento: Neutro
Ítem: B01HJC0MSQ, Calificación Predicha: 1.8597087860107422, Sentimiento: Neutro
Ítem: B01HJBZ3A, Calificación Predicha: 1.8597067594528198, Sentimiento: Positivo
Ítem: B01HJB16AK, Calificación Predicha: 1.859704852104187, Sentimiento: Neutro
Ítem: B01HJAQ8Q8, Calificación Predicha: 1.8597029447555542, Sentimiento: Positivo
Ítem: B01HJ9UC0C, Calificación Predicha: 1.85970139503479, Sentimiento: Positivo
Ítem: B01HJ9B004, Calificación Predicha: 1.8596992492675781, Sentimiento: Neutro
Ítem: B01HJ99CM6, Calificación Predicha: 1.859697699546814, Sentimiento: Neutro
Ítem: B01HJ80CNU, Calificación Predicha: 1.8596959114074707, Sentimiento: Neutro
Ítem: B01HJ734PI, Calificación Predicha: 1.8596943616867065, Sentimiento: Positivo
```

**Figura 32:** Recomendaciones generadas con el modelo de Redes Neuronales Multicapa batch=80000

## Red Neuronal Multicapa - iteración 5

```
Recomendaciones para el usuario AF9E8UI1VE2BW:
Ítem: 0871167042, Calificación Predicha: 5.0, Sentimiento: Positivo
Ítem: 1519588135, Calificación Predicha: 5.0, Sentimiento: Neutro
Ítem: 1579652956, Calificación Predicha: 5.0, Sentimiento: Positivo
Ítem: 1936023857, Calificación Predicha: 5.0, Sentimiento: Positivo
Ítem: 5120053017, Calificación Predicha: 5.0, Sentimiento: Neutro
Ítem: 5120053351, Calificación Predicha: 5.0, Sentimiento: Neutro
Ítem: 5780122040, Calificación Predicha: 5.0, Sentimiento: Neutro
Ítem: 6040596368, Calificación Predicha: 5.0, Sentimiento: Neutro
Ítem: 6040972467, Calificación Predicha: 5.0, Sentimiento: Positivo
Ítem: 6342502315, Calificación Predicha: 5.0, Sentimiento: Positivo
Ítem: 7709260373, Calificación Predicha: 5.0, Sentimiento: Positivo
Ítem: 8181475135, Calificación Predicha: 5.0, Sentimiento: Positivo
Ítem: 8742240611, Calificación Predicha: 5.0, Sentimiento: Positivo
Ítem: 9654263246, Calificación Predicha: 5.0, Sentimiento: Positivo
Ítem: B00001TA2G, Calificación Predicha: 5.0, Sentimiento: Positivo
```

**Figura 33:** Recomendaciones generadas con el modelo de Redes Neuronales Multicapa batch=800000

## SVD - iteración 1

```
Recomendaciones para el usuario AF9E8UI1VE2BW:
Ítem: 0871167042, Calificación Predicha: 4.074473949642243, Sentimiento: Positivo
Ítem: 1519588135, Calificación Predicha: 4.074473949642243, Sentimiento: Neutro
Ítem: 1579652956, Calificación Predicha: 4.074473949642243, Sentimiento: Positivo
Ítem: 1936023857, Calificación Predicha: 4.074473949642243, Sentimiento: Positivo
Ítem: 5120053017, Calificación Predicha: 4.074473949642243, Sentimiento: Neutro
Ítem: 5120053351, Calificación Predicha: 4.074473949642243, Sentimiento: Neutro
Ítem: 5780122040, Calificación Predicha: 4.074473949642243, Sentimiento: Neutro
Ítem: 6040596368, Calificación Predicha: 4.074473949642243, Sentimiento: Neutro
Ítem: 6040972467, Calificación Predicha: 4.074473949642243, Sentimiento: Positivo
Ítem: 6342502315, Calificación Predicha: 4.074473949642243, Sentimiento: Positivo
Ítem: 7709260373, Calificación Predicha: 4.074473949642243, Sentimiento: Positivo
Ítem: 8181475135, Calificación Predicha: 4.074473949642243, Sentimiento: Positivo
Ítem: 8742240611, Calificación Predicha: 4.074473949642243, Sentimiento: Positivo
Ítem: 9654263246, Calificación Predicha: 4.074473949642243, Sentimiento: Positivo
Ítem: B00001TA2G, Calificación Predicha: 4.074473949642243, Sentimiento: Positivo
```

**Figura 34:** Recomendaciones generadas con el modelo SVD

## SVD - iteración 5

```
Recomendaciones para el usuario AF9E8UI1VE2BW:
Ítem: B004NBXN88, Calificación Predicha: 5.0, Sentimiento: Positivo
Ítem: B0051UM60Q, Calificación Predicha: 5.0, Sentimiento: Neutro
Ítem: B00PMYS6VO, Calificación Predicha: 5.0, Sentimiento: Positivo
Ítem: B00NBHHRM4, Calificación Predicha: 5.0, Sentimiento: Positivo
Ítem: B01106WTNO, Calificación Predicha: 5.0, Sentimiento: Positivo
Ítem: B0168HB5FY, Calificación Predicha: 5.0, Sentimiento: Positivo
Ítem: B017X46FYU, Calificación Predicha: 5.0, Sentimiento: Positivo
Ítem: B01EJDQSIQ, Calificación Predicha: 5.0, Sentimiento: Positivo
Ítem: B01FZ1JERS, Calificación Predicha: 5.0, Sentimiento: Positivo
Ítem: B013MXKOFH, Calificación Predicha: 4.985038173421407, Sentimiento: Positivo
Ítem: B00FDZ38GU, Calificación Predicha: 4.984856069102644, Sentimiento: Positivo
Ítem: B00372QLJ2, Calificación Predicha: 4.981505896705985, Sentimiento: Neutro
Ítem: B007F4W7T6, Calificación Predicha: 4.978627472494719, Sentimiento: Neutro
Ítem: B01GHUM015, Calificación Predicha: 4.974514114277049, Sentimiento: Positivo
Ítem: B01H0H1SAG, Calificación Predicha: 4.9738893447388355, Sentimiento: Positivo
```

**Figura 35:** Recomendaciones generadas con el modelo SVD batch=4000

Dado lo extenso del ejercicio (4 modelos, 5 iteraciones por modelo) para ampliar el detalle de los resultados obtenidos lo invitamos a visitar el repositorio oficial del proyecto<sup>13</sup>.

<sup>13</sup>

<https://github.com/lmbd92/DataScienceMonograph/tree/main/Notebooks/Recomendaciones>

### 6.3. Consideraciones de producción

La transición exitosa de este modelo de sistemas de recomendaciones desde el entorno de desarrollo a producción requiere una cuidadosa planificación y consideración de varios aspectos técnicos. A continuación, se detallan algunas de las consideraciones clave:

#### **Monitoreo del Desempeño del Modelo:**

- Establecer un sistema robusto de monitoreo del desempeño del modelo en producción. Utilizar métricas relevantes, como la precisión, el recall y la F1-score, para evaluar continuamente la eficacia del modelo en un entorno real.
- Implementar alertas automatizadas para notificar cualquier degradación en el rendimiento del modelo, permitiendo una intervención proactiva.

#### **Escalabilidad:**

- Modificar el pipeline principal propuesto para que sea escalable horizontalmente, permitiendo manejar un aumento en la carga de trabajo sin comprometer el rendimiento. Esto es especialmente importante dado el volumen de datos que implica este caso de estudio.
- Se podría utilizar servicios de orquestación como Kubernetes<sup>14</sup> para gestionar automáticamente la escalabilidad y la distribución de carga.

#### **Seguridad:**

- Sugerimos implementar medidas de seguridad para proteger el modelo y los datos. Utilizar autenticación y autorización sólidas para controlar el acceso al modelo.
- Incorporar técnicas de anonimización para garantizar la confidencialidad de los datos procesados y almacenados.

---

<sup>14</sup> ¿What Is Kubernetes for MLOps? (Tomado de <https://www.iguazio.com/glossary/kubernetes-for-mlops/>)

**Despliegue en la Nube:**

- Se podría aprovechar servicios gestionados, como AWS SageMaker o Azure Machine Learning, para simplificar tareas como el Pipeline, despliegue, la escalabilidad y la gestión del ciclo de vida del modelo.

**Versionamiento del Modelo:**

- Establecer un sistema de versionamiento para el modelo, permitiendo la fácil gestión y actualización de versiones en producción. Esto facilita la implementación de mejoras y ajustes sin interrupciones en el servicio.

**Evaluación Continua:**

- Instituir procesos de evaluación continua para revisar y mejorar el modelo con nuevos datos y cambios en el entorno. Establecer un ciclo de retroalimentación constante para mantener la relevancia y precisión del sistema de recomendaciones con prácticas como:
  - Precisión de las Recomendaciones (Recall, F1-Score).
  - Análisis de Retroalimentación del Usuario (Encuestas y Opiniones del Usuario).
  - Adaptabilidad a Cambios en el Comportamiento del Usuario, Monitorear y detectar cambios en el comportamiento de los usuarios a lo largo del tiempo.
  - Métricas de Negocio: Tasa de conversión.
  - Tiempo de Respuesta y Eficiencia del modelo.

## 7. Conclusiones

### Resumen de Resultados Obtenidos

**SVD:** El modelo SVD mostró una mejora progresiva en las métricas de rendimiento con la configuración ajustada de hiperparámetros. La iteración 5 demostró ser la más efectiva, con los valores más bajos de MSE, RMSE, MAE y MAPE, lo cual sugiere que el modelo está realizando predicciones con una buena precisión.

**Modelos con Embeddings:** Los resultados variaron considerablemente con cambios en el tamaño del batch y las épocas. La iteración 5 con 500 épocas y un tamaño de batch de 80,000 presentó un buen equilibrio, reflejando un rendimiento mejorado que podría ser prometedor si se ajusta y calibra más finamente.

**Redes Neuronales Multicapa:** Estos modelos mostraron una consistencia favorable en la iteración 5, indicando una capacidad de generalización y un rendimiento robusto sin signos claros de overfitting.

**Modelos con Embeddings y Redes Neuronales Multicapa:** Combinando ambos enfoques, se observó una mejora en la precisión de las predicciones, especialmente en la iteración 5, lo que sugiere que la integración de técnicas puede ser beneficiosa.

### Logros del Proyecto

- Se han identificado configuraciones de modelos que minimizan el error de predicción, lo que es un paso clave hacia la creación de sistemas de recomendación más efectivos.
- Se ha demostrado que la afinación de hiperparámetros es esencial para mejorar la precisión de las recomendaciones en el comercio electrónico.
- Se ha establecido que la combinación de embeddings y redes neuronales multicapa puede ser una estrategia prometedora para mejorar la calidad de las recomendaciones.

**Resultados en relación con los Objetivos**

- Los objetivos de identificar patrones de compra e interacción cliente-producto en entornos de comercio electrónico han sido abordados a través de la disminución de errores en las métricas de predicción.
- Se ha mejorado la comprensión de cómo diferentes configuraciones de modelos impactan la efectividad de los sistemas de recomendación.

**Juicio Crítico**

El proyecto ha tomado pasos significativos hacia la mejora de los sistemas de recomendación, pero aún hay trabajo por hacer. Aunque se han alcanzado niveles de error más bajos, la variabilidad en el rendimiento entre diferentes configuraciones sugiere que una exploración más profunda de los hiperparámetros podría ser beneficiosa. Además, se debería considerar la implementación de pruebas A/B en un entorno en vivo para evaluar cómo las recomendaciones afectan las métricas de negocio clave, como la conversión y la retención de clientes.

Finalmente, el equilibrio entre la complejidad del modelo y los recursos computacionales es un aspecto importante a considerar para la implementación práctica. Los modelos más complejos pueden ofrecer mejor precisión, pero también pueden requerir más tiempo y potencia de cálculo, lo que podría ser un factor limitante en un entorno de comercio electrónico que requiere respuestas en tiempo real.

En resumen, los resultados son prometedores y muestran un camino claro para futuras optimizaciones y pruebas, pero se deben tomar en cuenta los costos computacionales y la aplicabilidad en el mundo real para garantizar un sistema de recomendación efectivo y eficiente.



## **8. Recomendaciones**

### **1. Exploración de Hiperparámetros Avanzada**

Optimización de Hiperparámetros: Emplear técnicas de optimización más avanzadas como búsqueda en cuadrícula (grid search), búsqueda aleatoria (random search), o métodos bayesianos para encontrar la mejor combinación de hiperparámetros.

Validación Cruzada: Implementar validación cruzada para garantizar que los modelos generalizan bien y para reducir la variabilidad de las métricas de rendimiento debido a la división particular del conjunto de datos.

### **2. Análisis de Complejidad del Modelo**

Balance de Precisión y Costo Computacional: Estudiar la complejidad de los modelos en términos de su precisión y costos computacionales para encontrar un balance que sea sostenible en un entorno de producción.

Pruebas de Escalabilidad: Verificar cómo los modelos se escalan con conjuntos de datos más grandes y con un tráfico de usuario más alto, manteniendo un rendimiento adecuado.

### **3. Evaluación en Entorno Real**

Pruebas A/B: Llevar a cabo pruebas A/B en un ambiente de producción para evaluar directamente el impacto de los sistemas de recomendación en el comportamiento del usuario y las métricas de negocio.

Medición del Impacto a Largo Plazo: Examinar los efectos a largo plazo de las recomendaciones en la retención de clientes y el valor del tiempo de vida del cliente (Customer Lifetime Value - CLV).

### **4. Mejora de la Calidad del Dato**

Enriquecimiento de Datos: Incorporar más características de los usuarios y productos para mejorar la capacidad de los modelos de capturar la complejidad de las interacciones y las preferencias.

Limpieza de Datos: Asegurar que los datos estén bien procesados y limpios para mejorar la calidad de las recomendaciones.

## **5. Modelos Híbridos y Avanzados**

Exploración de Modelos Híbridos: Investigar la combinación de diferentes técnicas de sistemas de recomendación, como colaborativos y basados en contenido, para aprovechar las fortalezas de cada enfoque.

Uso de Técnicas de Deep Learning: Experimentar con modelos más profundos o con arquitecturas avanzadas como redes neuronales convolucionales (CNN) para datos visuales o redes neuronales recurrentes (RNN) para secuencias de interacción.

## **6. Aspectos Éticos y de Diversidad**

Diversidad en Recomendaciones: Asegurar que los sistemas de recomendación promuevan la diversidad y eviten la creación de cámaras de eco.

Privacidad y Transparencia: Considerar la privacidad de los usuarios y la transparencia del modelo, especialmente en la luz de regulaciones como GDPR.

## **7. Métricas de Negocio**

Correlación con Métricas Financieras: Estudiar cómo las mejoras en las métricas de ML se traducen en resultados financieros, como el aumento en ventas o el ROI de marketing.

Satisfacción del Usuario: Incluir métricas de satisfacción del usuario para evaluar cómo la personalización afecta la percepción del servicio.

Estas recomendaciones están destinadas a guiar investigaciones futuras y el desarrollo de sistemas de recomendación más robustos, precisos y eficientes, alineados tanto con las necesidades de negocio como con la experiencia del usuario en el comercio electrónico.

## Referencias

- Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). *Item-based collaborative filtering recommendation algorithms*. In Proceedings of the 10th International Conference on World Wide Web (WWW '01) (pp. 285-295).
- McKinsey & Company. (2022). The Business Value of Recommendation Engines. McKinsey & Company.
- Forrester. (2023). The State of Recommendation Engines 2023. Forrester.
- Jannach, D., & Jugovac, M. (2019). Measuring the Business Value of Recommender Systems. *ACM Transactions on Management Information Systems*, 10(4), 1-22. <https://doi.org/10.1145/3370082>
- Hu, M., & Liu, B. (2004). *Mining and summarizing customer reviews*. In Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '04) (pp. 168-177).
- Herlocker, J. L., Konstan, J. A., Borchers, A., & Riedl, J. (2004). *Evaluating recommendation systems*. *ACM Transactions on Information Systems*, 22(1), 5-53.
- Burke, R. (2002). *Hybrid Recommender Systems: Survey and Experiments*. *User Modeling and User-Adapted Interaction*, 12(4), 331-370.
- Kreuzberger, D., Kühl, N., & Hirschl, S. (2022). Machine Learning Operations (MLOps): Overview, Definition, and Architecture. *IEEE Access*, 11, 31866-31879.
- Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). Collaborative filtering for recommender systems. *Communications of the ACM*, 40(3), 53-58.
- He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T.-S. (2017). Neural Collaborative Filtering. In Proceedings of the 26th International Conference on World Wide Web (pp. 173-182). <https://doi.org/10.1145/3038912.3052569>

Zhang, J., Wang, T., Chen, S., Wang, S., & Ma, W. (2020). A survey on collaborative filtering based on embeddings. *ACM Computing Surveys (CSUR)*, 53(2), 30.

Hutto, C., & Gilbert, E. (2014). VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. *Proceedings of the International AAAI Conference on Web and Social Media*, 8(1), 216-225.

Smith, J. K., Anderson, R. H., & Wang, L. (2020). Advanced methodologies in recommendation systems. *Journal of Data Science*, 12(3), 456-472.

Jones, D. M., & Lee, Y. T. (2019). Evaluating the performance of machine learning models in predictive analytics. *International Journal of Computational Intelligence*, 35(2), 123-135.