PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA

Implementación de un sistema de recomendación basado en el análisis de polaridad y caracterización de revisiones de usuarios de un *marketplace*

Tesis para obtener el título profesional de Ingeniero Informático

AUTOR:

Enrique André Pando Robles

ASESOR:

César Armando Beltrán Castañón

Lima, junio, 2021

Resumen

El crecimiento constante de Internet va de la mano con el rápido aumento del volumen de información, lo cual brinda una amplia gama de alternativas de compra al usuario, quien puede verse superado por la gran variedad de productos disponibles. A fin de ayudar en la elección de productos, se desarrollan los sistemas de recomendación, los cuales acotan los potenciales productos de agrado para el usuario. Con el fin de recabar mayor información, los sitios de comercio electrónico van añadiendo nuevas funcionalidades, tales como asignar una puntuación y comentarios sobre el producto adquirido. Este último expresa, en palabras del usuario, su sentimiento luego de realizar la compra, el cual podría ser un comentario positivo, negativo o neutro. Es así como surge la necesidad de poder analizar todos estos datos textuales, los cuales guardan una rica información sobre el parecer de los usuarios, pudiendo así ser utilizada para una potencial mejor toma de decisión a fin de mejorar el servicio de comercio.

Sin embargo, para poder otorgar una recomendación al usuario, es necesario analizar a los demás que ya adquirieron productos similares, siendo importante detectar a aquellos compradores que tengan un patrón similar (por ejemplo, en sus comentarios) siendo este un problema que demanda estrategias de filtrado de estas características similares.

Por lo tanto, el presente tema de tesis combina los sistemas de recomendación y el análisis de polaridad con el objetivo de brindar una recomendación de productos al usuario con base en las puntuaciones y comentarios (usando filtros colaborativos), de manera que el usuario pueda obtener una lista reducida de productos potenciales a adquirir con base en su histórico de compras. Teniendo como conclusión principal la comprobación estadística de que el sistema de recomendación propuesto es superior a solo usar las puntuaciones para recomendar.

Tabla de Contenido

Índice de Ilustraciones

Índice de Tablas

Capítulo 1. Generalidades

1.1 Problemática

Dado el acelerado crecimiento de información en Internet (Harrage, Alsalman, & Alqahtani, 2019), surge el reto sobre cómo afrontar el exponencial aumento de datos; es así que en el campo de ciencias de la computación se ha intentado hacer frente al reto del crecimiento en la cantidad de información textual a través de distintos métodos de manera que se pueda organizar esta información y transformarla en conocimiento que aporte valor (Harrage, Alsalman, & Alqahtani, 2019). Usualmente, las personas al querer comprar algo sesgan su decisión basándose en las características que buscan en ese producto, para finalmente quedarse con una lista reducida de potenciales candidatos, en la cual posteriormente se observan patrones más allá de las características del producto; es decir, las personas leen las revisiones textuales de otros usuarios sobre esos productos para determinar qué tan bueno es en calidad, garantía, entre otros (Harrage, Alsalman, & Alqahtani, 2019). Este comportamiento empezó incluso antes de Internet puesto que las personas consultaban con otras sobre qué tan bueno era un producto o servicio antes de comprarlo para medir, entre otras cosas, el nivel de satisfacción que se tenía sobre uno de los productos candidatos a comprar. Como parte de la información textual disponible en Internet se encuentran los comentarios de los usuarios o sus revisiones sobre algún producto. Dado esta gran cantidad de información, los investigadores se han planteado como reto analizarla y usarla de manera que pueda ayudar a los clientes en sus decisiones al comprar un producto (Harrage, Alsalman, & Alqahtani, 2019).

En años recientes, con el fin de poder abordar el problema del análisis de la información textual, los investigadores han ido adquiriendo un creciente interés en el análisis de sentimiento (Poggi & Augusto, 2016). El análisis de sentimiento o minería de opinión ha sido definido como el estudio computacional de las opiniones, sentimientos y emociones expresadas en texto (Leotta,

Beux, Mascardi, & Briola, 2015). Sus aplicaciones son extensas, yendo desde temas legales y psicológicos hasta comercio electrónico.

Así mismo, los sistemas de recomendación (SR) han ido atrayendo mucha atención tanto en el ámbito académico como industrial (Bobadilla, Ortega, Hernando, & Gutiérrez, 2013). Estos sistemas tienen como objetivo generar recomendaciones útiles para el usuario sobre productos que puedan ser de interés para él. También, estos ayudan a afrontar el problema de la sobrecarga de la información abundante que existe a través de una recomendación personalizada de artículos (libros, películas, entre otros) a usuarios basados en sus preferencias e intereses históricos (Ait Hammou & Ait Lahcen, 2017). Un ejemplo de esto puede ser que, al tener un producto con una mejor puntuación (donde 5 es mejor que 1), es más probable que este sea más atractivo para el usuario por sobre otros productos. En general, hay 2 tipos de enfoques en un sistema de recomendación: los basados en contenido y los basados en filtros colaborativos (Srifi, Hammou, Mouline, & Lahcen, 2018) los cuales serán explicados a continuación.

El sistema de recomendación basado en contenido se centra en analizar al producto como base de la predicción en lugar del usuario. Es decir, utiliza las características del producto como su marca, precio, tamaño, entre otras para poder recomendar productos similares con estas características (Nehete & Devane, 2018). El problema de este enfoque es que las recomendaciones están sesgadas a los productos que ha comprado el usuario anteriormente dado que, al ser predicciones basadas en las características del producto, no hay una amplia variedad de categorías o tipos de productos y no le da al usuario ese "factor sorpresa" de un producto totalmente nuevo que puede ser de su interés. Por ejemplo, si un usuario solo ha comprado memoria RAM, las recomendaciones que se le dará serán otros tipos de memoria RAM.

El segundo enfoque tiene como base al usuario; es decir, tiene como centro las características del usuario y es uno de los enfoques de recomendación más exitosos usado por varias compañías de venta electrónica como Amazon, eBay y Netflix (Ait Hammou & Ait Lahcen, 2017). Esto implica que, en este enfoque, se analiza el historial de compras del usuario, así como sus preferencias y calificaciones que le ha dado a otros productos y luego busca a usuarios que se parezcan a él (por ejemplo, habiendo comprado productos, calificado y opinado similar) para finalmente, dada una lista de productos que han sido exitosos en esos usuarios similares a él, recomendar esta lista de productos.

Este segundo enfoque funciona correctamente cuando hay suficiente información de las puntuaciones de usuarios a productos (Su & Khoshgoftaar, 2009). Sin embargo, su efectividad se ve reducida cuando ocurre el problema de la dispersión de datos (*sparsity problem*) (Liu, He, Wang, Song, & Du, 2013). Finalmente, después de una extensa revisión de la literatura y dado el origen de los datos a trabajar, se optó por utilizar el enfoque basado en filtros colaborativos dado que el primer enfoque está sesgado al producto en sí, mientras que el segundo enfatiza la variedad en la recomendación (Su & Khoshgoftaar, 2009). Por otro lado, para atacar el problema de la dispersión de los datos se tienen diversas técnicas como la descomposición por valores singulares (SVD por sus siglas en inglés) y la matriz de factorización no negativa, siendo la última la utilizada en el presente trabajo.

En la literatura se han trabajado distintos enfoques sobre cómo atacar el problema de recomendación, así como también se han desarrollado distintas formas de analizar el sentimiento en los textos; sin embargo, el problema radica en que estas recomendaciones se brindaban, en su mayoría, solo haciendo uso de las puntuaciones de productos (Nehete & Devane, 2018). No obstante, una puntuación puede no englobar todo lo que el usuario siente sobre el producto. Por ejemplo, dado un producto y dos usuarios, el primer usuario puede asignarle una puntuación de tres sobre cinco al producto mientras que el segundo le asigna una

puntuación de cinco sobre cinco, justificado en que el primer usuario puede encontrar una característica positiva del producto que no satisfaga su necesidad, mientras que, para el segundo usuario, esta característica encontrada sí lo satisface. Es por ello que, como enfoque básico, las puntuaciones cumplen con el propósito de mostrarnos la tendencia de las opiniones sobre un determinado producto, pero si se analizan los sentimientos que otros usuarios muestran en sus comentarios sobre dicho producto, se puede observar nuevas características subjetivas importantes a tomar en cuenta en la recomendación de productos.

A partir del uso de solo las puntuaciones, nace la inquietud de que si en el proceso final de selección de un producto por parte del usuario, luego de que este reduce su lista de candidatos a una cantidad manejable, se revisan los comentarios de otros usuarios sobre estos productos. De esta manera surge una pregunta a responder: ¿cómo se puede recomendar un producto a un usuario basado en los comentarios de los demás usuarios que compraron el producto en cuestión?

A raíz de ello, en el presente trabajo se desarrolló un modelo algorítmico que, tomando en cuenta el análisis de polaridad de los comentarios de los usuarios sobre un producto, se mejore el porcentaje de exactitud de la recomendación; es decir, la puntuación predicha sobre el producto sea más cercana a la puntuación real asignada por el usuario.

1.2 Objetivos

1.2.1 Objetivo general

Implementar un sistema de recomendación de productos para un *marketplace* basado en el análisis de polaridad y puntuaciones de los usuarios.

1.2.2 Objetivos específicos

O 1. Caracterizar perfiles de usuarios con base en sus comentarios y puntuaciones sobre los productos ofrecidos en un *marketplace.*

O 2. Predecir productos a recomendar usando solo las puntuaciones otorgadas por los usuarios del *marketplace* como modelo base.

O 3. Predecir la polaridad de los comentarios de los usuarios.

O 4. Predecir productos a recomendar usando las puntuaciones y comentarios de evaluaciones de los usuarios, comparándolo con el modelo base.

1.2.3 Resultados esperados

R 1. Para el objetivo 1.- Conjunto de datos (matriz) estructurados de puntuaciones.

R 2. Para el objetivo 1.- Conjunto de comentarios de usuarios de un *marketplace* sobre productos libres de términos poco relevantes para su posterior análisis de polaridad.

R 3. Para el objetivo 2.- Sistema de recomendación de productos basado solo en las puntuaciones de los productos.

R 4. Para el objetivo 2.- Reporte de análisis de resultados de productos recomendados con base en las puntuaciones de los productos.

R 5. Para el objetivo 3.- Algoritmo que evalúa un comentario asignándole un grado de polaridad (positivo o negativo).

R 6. Para el objetivo 4.- Sistema de recomendación de productos con base en el análisis de polaridad y las puntuaciones de los productos.

R 7. Para el objetivo 4.- Reporte de análisis de resultados de productos recomendados con base en el análisis de polaridad y las puntuaciones de los productos.

1.2.4 Mapeo de objetivos, resultados y verificación

Tabla 1. Mapeo de objetivos, resultados y verificación (elaboración propia)

Objetivo: Caracterizar perfiles de usuarios con base en sus comentarios y puntuaciones sobre los productos ofrecidos en un *marketplace*.		
Resultado	**Meta física**	**Medio de verificación**
Conjunto de datos (matriz) estructurados de puntuaciones.	*Dataframe* de puntuaciones de usuarios en productos.	- Visualización de los datos del *dataframe* a través de gráficos.
Objetivo: Caracterizar perfiles de usuarios con base en sus comentarios y puntuaciones sobre los productos ofrecidos en un *marketplace*.		
Resultado	**Meta física**	**Medio de verificación**
Conjunto de comentarios de usuarios de un *marketplace* sobre productos libres de términos poco relevantes para su posterior análisis de polaridad.	*Dataframe* de comentarios de los usuarios sobre los productos.	- Visualización de los datos del *dataframe* a través de gráficos.
Objetivo: Predecir productos a recomendar usando solo las puntuaciones otorgadas por los usuarios del *marketplace* como modelo base.		
Resultado	**Meta física**	**Medio de verificación**
Sistema de recomendación de productos basado solo en las puntuaciones de los productos.	Sistema de recomendación	- Error medio absoluto (MAE). - Error cuadrático medio (RMSE).
Objetivo: Predecir productos a recomendar usando solo las puntuaciones otorgadas por los usuarios del *marketplace* como modelo base.		
Resultado	**Meta física**	**Medio de verificación**
Reporte de análisis de resultados de productos recomendados con base en las puntuaciones de los productos.	Documento de análisis de resultados.	- Gráficos de pruebas de integridad de datos.
Objetivo: Predecir la polaridad de los comentarios de los usuarios.		
Resultado	**Meta física**	**Medio de verificación**
Algoritmo que evalúa un comentario asignándole	Algoritmo de predicción de la polaridad	- Métricas de precisión, *recall,* F1 y *accuracy.*

un grado de polaridad (positivo o negativo).		
Objetivo: Predecir productos a recomendar usando las puntuaciones y comentarios de evaluaciones de los usuarios, comparándolo con el modelo base.		
Resultado	**Meta física**	**Medio de verificación**
Sistema de recomendación de productos con base en el análisis de polaridad y las puntuaciones de los productos.	Sistema de recomendación	- Error medio absoluto (MAE). - Error cuadrático medio (RMSE).
Objetivo: Predecir productos a recomendar usando las puntuaciones y comentarios de evaluaciones de los usuarios, comparándolo con el modelo base.		
Resultado	**Meta física**	**Medio de verificación**
Reporte de análisis de resultados de productos recomendados con base en el análisis de polaridad y las puntuaciones de los productos.	Documento de análisis de resultados.	- Gráficos de pruebas de integridad de datos.

1.3 Herramientas y Metodología

1.3.1 Herramientas

- **Python**

Python es un lenguaje de programación interpretado, orientado a objetos y de alto nivel con semánticas dinámicas. Sus estructuras de datos integradas de alto nivel, combinado con la escritura dinámica y enlace dinámico, lo hace una alternativa bastante atractiva, así como también su capacidad de admitir el uso de módulos y paquetes fomentando de esta manera la modularidad del programa y la reutilización de código (Python Software Foundation, 2019). Por último, Python se distingue de otros lenguajes de programación por sus excelentes funcionalidades para procesar datos lingüísticos (Bird, Klein, & Loper, 2009), razón por la cual fue escogido para el desarrollo del presente proyecto.

- **Jupyter Notebook**

Jupyter Notebook es una aplicación web de código abierto que te permite crear y compartir documentos que contengan códigos, ecuaciones, visualizaciones y textos narrativos. Permite configurar y ordenar la interfaz de usuario para soportar una amplia cantidad de flujos de trabajo en campos como ciencia de datos, computación científica y aprendizaje de máquina. (Project Jupyter, 2019). Fue utilizado como *ide* del proyecto, teniendo como utilidades mostrar el código y los resultados de manera clara, ordenada y fácil de visualizar.

- **Natural Language Toolkit**

Natural Language Toolkit es una librería de Python que posee una amplia gama de funcionalidades destinadas al procesamiento de lenguaje natural (Bird, Klein, & Loper, 2009). Esta librería fue usada para realizar el preprocesamiento de la información textual.

- **Gensim**

Gensim es una librería gratuita de Python para analizar la estructura semántica de documentos de texto plano, recuperar la similitud semántica entre documentos y trabajar con semántica estadísticamente escalable (Rehurek,2010). Fue usado para representar de forma numérica la información textual a través de Word2Vec, algoritmo para descubrir automáticamente la estructura semántica de un documento examinando patrones estadísticos de coocurrencia dentro de un conjunto de entrenamiento de documentos.

- **Sklearn**

Sklearn es una eficiente herramienta y de fácil uso de Python para predecir y analizar datos (Pedregosa, 2011). Se usó para analizar y trabajar con la información textual, previamente transformada en información numérica.

1.3.2 Metodología de desarrollo

En la ilustración 1 se puede observar el esquema de trabajo del proceso principal, recomendación a través de puntuaciones y revisiones textuales hechas por los usuarios hacia los productos.

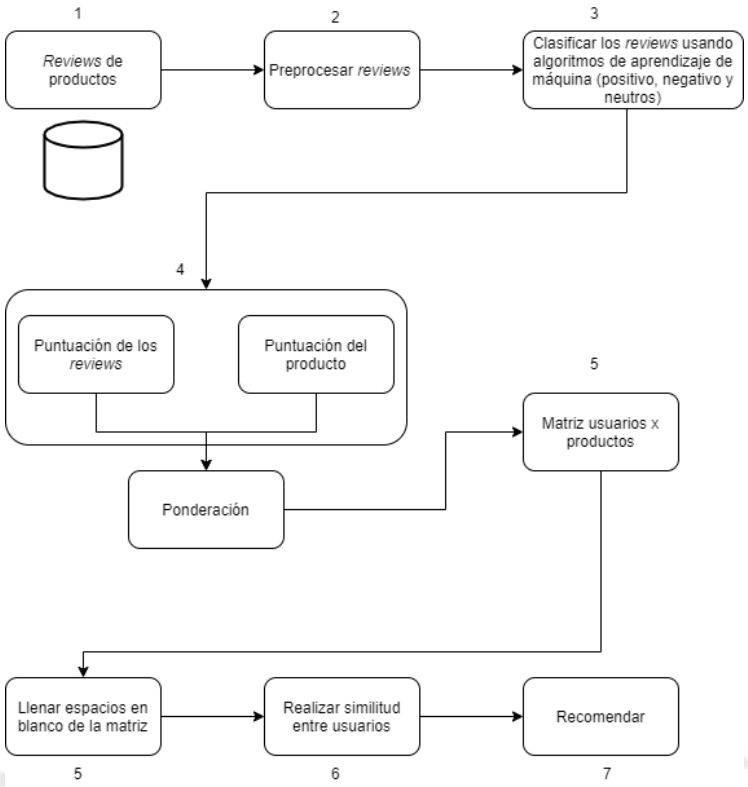


Ilustración 1.Esquema de trabajo del prototipo de sistema de recomendación usando puntuaciones y comentarios. (Elaboración propia).

En la ilustración 2 se muestra el flujo de trabajo del sistema de recomendación simple que recibe como entrada solo las puntuaciones de los productos y recomienda con base en ello.

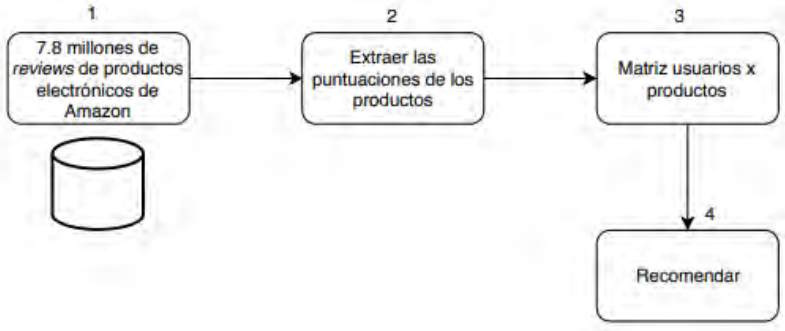


Ilustración 2. Esquema de trabajo del prototipo de sistema de recomendación simple usando solo puntuaciones (Elaboración propia).

Como se indicó en la problemática, se desea conocer si la información textual (comentarios) de los usuarios es relevante al momento de recomendar un producto, para ello se compararán ambos algoritmos de recomendación a través de las métricas de evaluación y análisis estadístico.

- **Preprocesamiento**

Este proceso es considerado uno de los más importantes puesto que ayuda a remover diferentes tipos de ruido (Harrage, Alsalman, & Alqahtani, 2019). Se procedió a remover puntuaciones, comillas y 2 o más espacios en blanco contiguos. Así mismo, se hizo una lematización de las palabras, las cuales fueron limitadas a aquellas que tuvieran una longitud de 2 caracteres o más.

Finalmente, se removieron los *stop words*, palabras que carecen de un valor intrínseco como vendrían a ser (en el idioma inglés, puesto que la información textual está en ese idioma) *in, on, the,* entre otros.

- **Similitud de Spearman**

Se usará la similitud de Spearman para hallar la similitud o correlación entre las puntuaciones hechas por los usuarios para así poder encontrar un conjunto de usuarios similares al actualmente evaluado. La similitud de Spearman consiste en encontrar un coeficiente de correlación no entre los valores de 2 variables, sino entre los puestos de estas variables. Se define de la siguiente ilustración 3:

$$r_{x,y} = \frac{\sum(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum(x_i - \overline{x})^2}\sqrt{\sum(y_i - \overline{y})^2}}$$

Ilustración 3. Ecuación de la similitud de Spearman (Van Dongen, S., & Enright, A. J., 2012).

- **Aprendizaje de máquina supervisado**

Ahora, una vez hecho el preprocesamiento, se puede proceder a realizar el análisis de sentimiento y usar uno de los algoritmos de aprendizaje de máquina para extraer la polaridad de cada revisión textual realizada por los usuarios y entrenar el modelo (Harrage, Alsalman, & Alqahtani, 2019).

En el enfoque de aprendizaje de máquina, un modelo será construido a partir de una base de datos de entrenamiento para poder aprender a clasificar elementos. Después de la construcción del modelo, este será usado en el proceso de clasificación de la base de datos de prueba (Harrage, Alsalman, & Alqahtani, 2019). En el presente trabajo, se ha usado el algoritmo de *Support Vector Machine* (SVM).

- **Máquina de vectores de soporte**

Es un clasificador lineal supervisado cuyo objetivo es encontrar un hiperplano tal que separe las clases a predecir con el mayor margen posible entre ellas (SVM por sus siglas en inglés). Luego de ubicar la información a un espacio dimensional superior (ya que de esta forma es más fácil clasificar con superficies lineales), se encuentra un hiperplano que clasificará y separará estos puntos (Harrage, Alsalman, & Alqahtani, 2019). Se usará en el presente proyecto para la clasificación de la polaridad del sentimiento, el cual es una clasificación binaria donde un comentario es catalogado de forma global con un sentimiento positivo o negativo. Puede extrapolarse a ser una clasificación multiclase (positivo, negativo y neutro).

- **Descomposición en Valores Singulares**

Es una técnica bastante conocida (SVD por sus siglas en inglés) que es usada para aproximar valores en una matriz de rangos dados (Sheng Zhang, Weihong Wang, Ford, Makedon, & Pearlman, 2005). Su utilidad radica en que permite predecir los valores faltantes en la matriz de puntuaciones de usuarios vs productos. Una vez obtenida la matriz, esta será considerada como la salida final del sistema de recomendación.

- ***Random forest***

Es un método de aprendizaje conjunto para clasificación que añade una capa adicional de aleatoriedad al *bagging*. Adicionalmente a construir cada árbol usando una diferente muestra de los datos, este método cambia cómo los árboles de regresión o clasificación son construidos

(Liaw, A., & Wiener, M., 2002). Es decir, es un método que se basa en generar una estructura de datos en forma de árbol encargada de clasificar un elemento en una determinada categoría y el resultado final será aquella categoría predicha por una mayor cantidad de árboles.

- **Matriz de factorización no negativa**

Es un método para poder brindar una representación adecuada de los datos usando la estructura de factores latentes, la cual se suele utilizar para reducir la dimensionalidad de los datos de manera que puedan ser usados posteriormente por otros métodos (Paatero and Tapper, 1994). Su utilidad radica en que, al tener bastante información de comentarios, mantener todos en memoria es muy costoso. He ahí donde radica la importancia de este método, ya que permite reducir la dimensionalidad de los datos, haciendo más factible y eficiente poder trabajar una gran cantidad de los mismos.

- **Gradiente descendente estocástica**

Es un método de optimización iterativo que toma de manera aleatoria una muestra durante cada iteración y calcula su gradiente. Luego, se toma este gradiente estocástico para actualizar los pesos del algoritmo dado una taza de aprendizaje (Robbins, H., & Monro, S., 1951). Este será usado para calcular y actualizar los pesos de la matriz de factorización no negativa.

- **Métricas de evaluación**

Las métricas de evaluación presentadas a continuación tienen como propósito describir el desempeño del nuevo algoritmo implementado en contraste con el definido como línea base.

- **Exactitud**

  La exactitud o *accuracy* se define como la fracción de predicciones que el modelo realizó correctamente

$$E = \frac{\text{Número de predicciones correctas}}{\text{Número total de predicciones}}$$

- **Precisión**

La precisión se define como el porcentaje de entidades identificadas y clasificadas correctamente con respecto al total del modelo identificado correctamente (Chinchor, N., & Sundheim, B., 1993).

$$P = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

- ***Recall***

  Conocido también como sensibilidad, se define como el porcentaje de entidades relacionadas extraídas con respecto al total de entidades relevantes (Chinchor, N., & Sundheim, B., 1993).

$$R = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

- **Medida F**

La medida F nos brinda una forma de combinar las métricas de precisión y exactitud, su fórmula es:

$$F = \frac{(2 \text{ x R x P})}{\text{R} + \text{P}}$$

Donde:

P es precisión y R es sensibilidad

- **Error medio absoluto**

Es la métrica de evaluación más usada en el mundo en sistemas de recomendación (MAE, por sus siglas en ingles). Estima la media de la diferencia absoluta entre los valores

estimados y las predicciones. Un sistema de recomendación basado en filtros colaborativos se considera que está funcionando correctamente cuando el MAE es pequeño. Para nuestro algoritmo, mientras más pequeño sea el MAE con respecto al modelo base, más eficiente será el análisis de las opiniones de las revisiones textuales. Su ecuación es Li, S., (Huang, C. R., Zhou, G., & Lee, S. Y. M., 2010:

$$MAE = \frac{1}{n} + \sum_{i=1}^{n} |Fi - Yi|$$

Donde:

n es el número de predicciones, Fi es la predicción del elemento i e Yi es el valor real del elemento i.

- **Error cuadrático medio**

El error cuadrático medio (RMSE por sus siglas en inglés) de un estimador mide el promedio de los errores al cuadrado, es decir, la diferencia entre el estimador y lo que se estima. El ECM es una función de riesgo, correspondiente al valor esperado de la pérdida del error al cuadrado o pérdida cuadrática. En una analogía con la desviación estándar, tomar la raíz cuadrada del ECM produce el error de la raíz cuadrada de la media o la desviación de la raíz cuadrada media (RMSE), que tiene las mismas unidades que la cantidad que se estima. Para un estimador insesgado, el RMSE es la raíz cuadrada de la varianza, conocida como la desviación estándar.

$$RMSE = \sqrt{\frac{1}{n} + \sum_{i=1}^{n} (Fi - Yi)^2}$$

Donde:

n es el número de predicciones, Fi es la predicción del elemento i e Yi es el valor real del elemento i.

1.4 Alcances y limitaciones

El presente proyecto titulación se realizó utilizando como conjunto de datos revisiones textuales (comentarios) de los usuarios de Amazon sobre productos electrónicos, así como sus respectivas puntuaciones otorgadas. El inicio del proyecto consistió en el preprocesamiento de las revisiones textuales de un aproximado de 7.8 millones de comentarios.

Luego, se procedió con la extracción de las características como vendrían a ser el número de oraciones, palabras positivas, palabras negativas, predicados, adjetivos y adverbios.

Posteriormente se aplicó un algoritmo de clasificación de aprendizaje de máquina para clasificar los comentarios (información textual) con base en su polaridad (positivo y negativo).

Seguido a ello, se procedió a utilizar la similitud de Spearman para calcular la similitud entre usuarios con el objetivo de encontrar para cada usuario el conjunto de usuarios similares a él (usuarios con potenciales productos de interés).

Finalmente, usando las polaridades de los comentarios y la matriz de similitud entre usuarios, se ponderaron y se generó una matriz de usuarios por productos, la cual fue completada por el método de matriz de factorización no negativa junto al gradiente descendente estocástico para predecir las puntuaciones, las cuales vendrían a ser la salida final del sistema de recomendación.

Se tuvo como limitaciones el problema del *cold start*, lo cual implica que las circunstancias no son todavía lo suficientemente óptimas para que el modelo algorítmico provea los mejores resultados posibles. Esto quiere decir que se necesitó que un usuario haya comprado al menos

un producto para poder realizar una recomendación óptima. Caso contrario, de tratarse de un usuario nuevo, el sistema recomendará los mejores productos de las categorías que tenga disponible como productos sugeridos.

1.5 Viabilidad

1.5.1 Viabilidad técnica

Los lenguajes de programación, las herramientas y métodos a utilizar durante el desarrollo del presente proyecto son de libre acceso. Adicionalmente, quien suscribe posee experiencia en su uso.

1.5.2 Viabilidad temporal

Se utilizó 6 meses para el desarrollo de todas las actividades derivadas de los objetivos específicos planteados. En el Anexo A se presentan las actividades realizadas y el tiempo de ejecución.

1.5.3 Viabilidad económica

Es importante resaltar el costo del tiempo invertido tanto del estudiando como del asesor; sin embargo, dado que las herramientas, los lenguajes de programación y los datos utilizados son de libre acceso, no existen limitaciones financieras ni de accesibilidad para el presente proyecto.

1.6 Riesgos

Tabla 2. Riesgos identificados en el proyecto (elaboración propia)

Descripción	Probabilidad	Impacto	Severidad	Mitigación	Contingencia
Denegación de uso de la información anónima de los usuarios por parte de Amazon	0.1	0.7	0.4	Se puede utilizar comentarios de otro *marketplace*	Análisis y obtención del conjunto de datos de comentarios de usuarios de otro *marketplace*

Capítulo 2. Marco Conceptual

Como se expresó en la problemática, la tesis consiste en la combinación de la extracción de la polaridad de los comentarios de las personas sobre los productos y la técnica de sistemas de recomendación basada en filtros colaborativos. A continuación, se presentarán los conceptos necesarios para entender los temas abordados en el presente documento:

2.1 Análisis de Sentimiento

El análisis de sentimiento o minería de opinión ha sido definido como el estudio computacional de las opiniones, sentimientos y emociones expresadas en texto (Leotta, Beux, Mascardi, & Briola, 2015). Kumar y Sebastian distinguen las siguientes tareas en el proceso del análisis de sentimiento (Poggi & Augusto, 2016, A. Kitchenham, 2007):

1. Clasificación de la subjetividad
2. Clasificación del sentimiento
3. Extracción del sujeto de la opinión
4. Extracción del objeto y la característica y/o aspecto

A estas tareas se podría añadir la sumarización de los sentimientos encontrados (Poggi & Augusto, 2016).

El análisis de sentimiento ha venido evolucionando y pasando por diversos niveles de granularidad (Poggi & Augusto, 2016, A. Kitchenham, 2007, Thelwall, Buckley, & Paltoglou, 2012):

- A nivel de documento: Consiste en determinar la polaridad o sentimiento general de un documento completo. Se considera el documento completo como una única opinión emitida por un sujeto sobre un único objeto. Usualmente depende de la determinación del sentimiento a nivel de palabra y a nivel de frase. (Poggi & Augusto, 2016)

- A nivel de oración. Nivel en el cual la polaridad del sentimiento expresada en cada oración es identificada (positiva o negativa). (Liu, 2015). Se clasifica primero la oración en objetiva o subjetiva. Cada oración subjetiva es considerada una única opinión emitida por un sujeto sobre un único objeto. También depende de la determinación del sentimiento a nivel de palabra y a nivel de frase. (Poggi & Augusto, 2016)
- A nivel de palabra: Consiste en utilizar la polaridad asociada a cada palabra para inferir la polaridad del texto que las contiene. (Ekman, 1994)
- A nivel de frase: Se diferencia de la granularidad a nivel de palabra en que se considera grupos de evaluación, en vez de palabras, como unidades mínimas de expresión (Wei & Gulla, 2010). Se considera el contexto de cada palabra para corregir su polaridad general.
- A nivel de aspecto y/o característica. Consiste en identificar el sentimiento de manera diferenciada por cada atributo de un producto (Picard, 2000). Los aspectos y/o características son extraídos y la opinión de cada uno de ellos es clasificada como positiva o negativa (Liu, 2015).
- A nivel de concepto: Consiste en utilizar como recurso a grandes bases de conocimientos para tomar como unidad mínima de expresión los conceptos expresados en el lenguaje natural de manera implícita o explícita (Poggi & Augusto, 2016, Cambria, Mazzocco, & Hussain, 2013).

2.2 Sistemas de Recomendación

Los sistemas de recomendación ayudan a afrontar el problema de la sobrecarga de información mediante una recomendación personalizada de artículos como libros, películas, entre otros para usuarios basados en sus preferencias e intereses pasados (Ait Hammou & Ait Lahcen, 2017).

En general, los sistemas de recomendación pueden ser divididos en 2 modelos (Srifi, Hammou, Mouline, & Lahcen, 2018):

- **Basados en filtros colaborativos**

  En particular, este método se basa en la similitud entre usuarios con un historial de puntuaciones para poder inferir sus preferencias con respecto a los productos (Herlocker, Konstan, Terveen, & Riedl, 2004). Está compuesto por 3 tipos, algoritmos basados en memoria, basados en modelos e híbridos. El primero es esencialmente tradicional ya que predice las puntuaciones basándose en una colección entera de productos previamente puntuados por los usuarios, para posteriormente brindar una similitud entre usuarios o entre productos a través de medidas como la similitud de Pearson, Spearman y Coseno (Bobadilla, Hernando, Ortega, & Bernal, 2011, Pazzani, 1999). El segundo nace por la necesidad de resolver los problemas del algoritmo basado en memoria (en el cual necesitas tener a todos los usuarios y/o productos en memoria para poder predecir correctamente, lo cual conlleva a un alto costo en recursos) y por ello se crean modelos que reciben entradas y del cual se obtienen salidas. Así mismo, cuentan con información de pruebas y entrenamiento. Algunos algoritmos son: Modelo Bayesiano (Ericson & Pallickara, 2013), clusterización (Ericson & Pallickara, 2011), entre otros.

  Finalmente, el último trata de combinar ambos aspectos de manera que se combinen las fortalezas y se disminuyan las desventajas individuales que poseen.

- **Basados en filtros de contenidos**

  Es un método constituido exclusivamente en los productos. Está basado netamente en recuperación de texto, por ejemplo, búsqueda semántica. Hay dos tipos, el primero es heurístico y el segundo basado en modelos. El primero lo contempla el algoritmo KNN y un modelo basado en vectores espaciales para los perfiles de

usuario que están basados en ontologías (Nehete & Devane, 2018). El segundo lo componen, por ejemplo, los algoritmos basados en modelos Bayesianos (Gorodetsky, Samoylov, & Serebryakov, 2010), algoritmos de árboles de decisión (Marović, Mihoković, Mikša, Pribil, & Tus, 2011) y algoritmos de clusterización (Vozalis & Margaritis, 2007).

En la ilustración 4 se presenta gráficamente los tipos de sistemas de recomendación, en ella se puede apreciar de forma jerárquica las distintas familias de métodos.

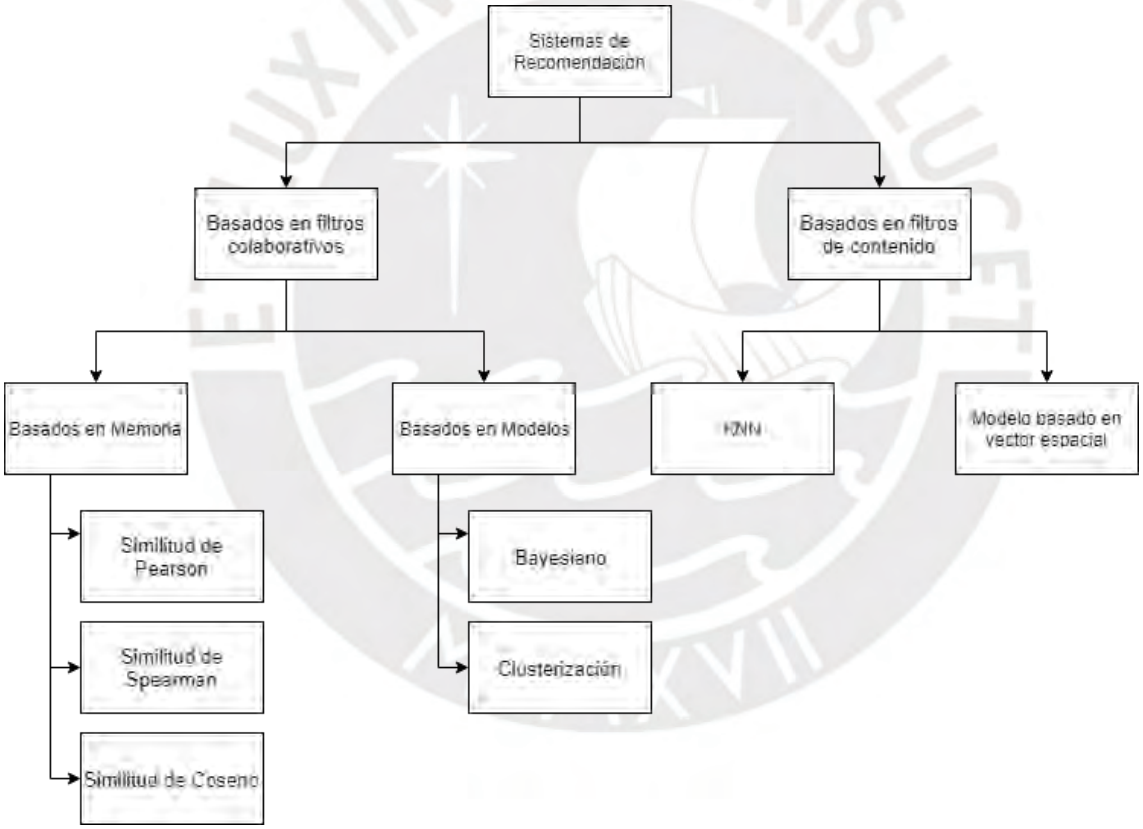


Ilustración 4. Representación en forma de árbol de algunas de las diversas técnicas para la elaboración de un sistema recomendador. Adaptado de (Nehete & Devane, 2018).

Capítulo 3. Estado del Arte

La presente revisión sistemática fue hecha con base en lo estipulado por Kitchenham and Charters (A. Kitchenham, 2007). En este caso, las actividades propuestas para la realización del proceso de revisión sistemática son: La elaboración de las preguntas de búsqueda, la definición de una estrategia de búsqueda, la selección de los estudios principales, la extracción de la información y la implementación de la síntesis de la estrategia (por ejemplo, los resultados del análisis).

3.1 Preguntas de Búsqueda

Basado en el método de PICOC (Petticrew & Roberts, 2006)

Preguntas generadas:

1. ¿Qué métodos de recomendación existen?
2. ¿Qué abarca *sentiment analysis*?
3. ¿Qué tipo de interacción se ha desarrollado entre estos dos rubros?
4. ¿Cuál es el aporte del *sentiment analysis* en un sistema de recomendación?
5. ¿En qué medida se está aprovechando los comentarios de los usuarios en las recomendaciones?
6. ¿Cuál método de recomendación es el más óptimo para la problemática en cuestión?

3.2 Estrategia de búsqueda y selección de fuentes

Se definieron los conceptos generales que permiten formular la tabla 3 con base en las preguntas elaboradas en la sección previa.

Estudio preliminar:

Cadena de búsqueda:

(TITLE-ABS-KEY (Sistema AND recomendador) AND TITLE-ABS-KEY (análisis AND sentimiento)) AND (LIMIT-TO (SUBJAREA, “COMP”))

Tabla 3. Criterios PICOC preliminar primera versión (elaboración propia)

Criterio	Descripción
Population	Sistema recomendador, análisis de sentimiento
Intervention	
Comparison	
Outcome	
Context	Ciencias de la computación

Se encontraron 291 resultados en Google, Google Scholar y Scopus con fecha de búsqueda 28/04/2019

Adicionando términos y concatenación de la cadena de búsqueda:

A partir de los títulos, sumarios, palabras clave y contenido de las publicaciones relevantes identificadas en la búsqueda preliminar, se identificaron términos que ayudan a reducir el espectro de posibilidades a solo aquellas que son relevantes para el estudio en cuestión como por ejemplo, colaborativo, que es el enfoque recomendado por la literatura para este tipo de problemáticas y que ha demostrado ser el más óptimo. Así mismo, se decidió como criterio de exclusión, en conjunto con el asesor, considerar solo artículos del año 2016 en adelante. A partir de esto, se obtuvo la siguiente cadena de búsqueda y tabla 4:

Cadena de búsqueda:

(TITLE-ABS-KEY (Sistema AND recomendador) AND TITLE-ABS-KEY (Análisis AND sentimiento) AND TITLE-ABS-KEY (colaborativo)) AND (LIMIT-TO (PUBYEAR, 2019) OR LIMIT-TO (PUBYEAR, 2018) OR LIMIT-TO (PUBYEAR, 2017) OR LIMIT-TO (PUBYEAR, 2016)) AND (LIMIT-TO (SUBJAREA, “COMP”))

Tabla 4. Criterios PICOC preliminar segunda versión (elaboración propia)

Criterio	Descripción
Population	Sistema recomendador, análisis de sentimiento, colaborativo
Intervention	
Comparison	
Outcome	
Context	Ciencias de la computación

Se encontraron 48 resultados en Google, Google Scholar y Scopus con fecha de búsqueda 28/04/2019

3.3 Selección de fuentes

La selección de base de datos fue realizada a partir de otras revisiones sistemáticas del área de ciencias de la computación y de las sugerencias de la web de la Pontificia Universidad Católica del Perú para esta área. Las bases de datos consultadas fueron:

- ACM Digital Library (http://dl.acm.org/)
- IEEE Xplore (http://ieeexplore.ieee.org/)
- ScienceDirect (http://www.sciencedirect.com/)
- Scopus (https://www.scopus.com/)
- Google Scholar (https://scholar.google.com.pe/)

3.4 Criterios de inclusión y exclusión

Como criterios de exclusión, se determinó con el asesor tomar en cuenta solo artículos científicos del año 2016 hasta la fecha, a menos que sean de carácter explicativo por ser los precursores y/o formar parte del origen del término a usar, en su ámbito. Así mismo, se excluyeron los artículos con idioma diferente al inglés o en español.

Por otro lado, dentro de los términos de inclusión planteados con el asesor, se añadió el artículo de Bamane (Bamane, 2016) puesto que presentaba un estudio de la información de los usuarios

de Amazon (empresa y *dataset* que se usará en el presente trabajo) de forma visual, de manera que nos permita observar cómo se comporta la información de los comentarios de los usuarios a lo largo de los años.

Así mismo, se decidió incluir algunos artículos que cumplan con el requisito de la cadena de búsqueda y cuya información sea relevante para el estudio a realizar, así como también aquellos donde se plantearon por primera vez los términos, independientemente de su año de publicación. De esta forma, se incluyó artículos como los de la segunda conferencia internacional en control automático, telecomunicaciones y señales (Ziani et al., 2017).

Al finalizar, se obtuvieron los siguientes artículos de las bases de datos consultadas (tabla 5):

Tabla 5. Artículos finales (elaboración propia)

ID	Año	Autor(es) y título
01	1994	Ekman, P. All Emotions Are Basic. In Ekman, P. & Davidson, R. (Eds.), The Nature of Emotion: Fundamental Questions (pp. 15-19)
02	1999	Pazzani, M. J. A Framework for Collaborative, Content-Based and Demographic Filtering
03	2000	Picard, R. W. Affective Computing. MIT Press
04	2004	Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. Evaluating Collaborative Filtering Recommender Systems.
05	2007	Vozalis, M. G., & Margaritis, K. G. Using SVD and Demographic Data for the Enhancement of Generalized Collaborative Filtering.
06	2009	Su, X., & Khoshgoftaar, T. M. A Survey of Collaborative Filtering Techniques.
07	2010	Wei, W., & Gulla, J. A. Sentiment Learning on Product Reviews via Sentiment Ontology Tree
08	2011	Marović, M., Mihoković, M., Mikša, M., Pribil, S., & Tus, A. Automatic movie ratings prediction using machine learning
09	2011	Bobadilla, J., Hernando, A., Ortega, F., & Bernal, J. A framework for collaborative filtering recommender systems.
10	2012	Thelwall, M., Buckley, K., & Paltoglou, G. Sentiment Strength Detection for the Social Web
11	2013	Bobadilla, J., Ortega, F., Hernando, A., & GutiéRrez, A. Recommender Systems Survey
12	2013	Liu, H., He, J., Wang, T., Song, W., & Du, X. Combining User Preferences and User Opinions for Accurate Recommendation
13	2013	Ericson, K., & Pallickara, S. On the Performance of High Dimensional Data Clustering and Classification Algorithms.

14	2015	Leotta, M., Beux, S., Mascardi, V., & Briola, D. My MOoD, a Multimedia and Multilingual Ontology Driven MAS: Design and First Experiments in the Sentiment Analysis Domain
15	2015	Liu, B. Sentiment Analysis: Mining Opinions, Sentiments, and Emotions.
16	2016	Poggi, O., & Augusto, C. Revisión sistemática sobre la aplicación de ontologías de dominio en el análisis de sentimiento
17	2016	Bamane, P. A Study of Amazon User Review Data using Visualization
18	2017	Ait Hammou, B., & Ait Lahcen, A. FRAIPA: A fast recommendation approach with improved prediction accuracy
19	2017	Ziani, A., Azizi, N., Schwab, D., Aldwairi, M., Chekkai, N., Zenakhra, D., & Cheriguene, S. Recommender System Through Sentiment Analysis
20	2018	Nehete, S. P., & Devane, S. R. Recommendation Systems: Past, Present and Future.
21	2018	Srifi, M., Hammou, B. A., Mouline, S., & Lahcen, A. A. Collaborative Recommender Systems Based on User-Generated Reviews: A Concise Survey
22	2019	Harrage, F., Alsalman, A., & Alqahtani, A. Rating Predictor: Sentiment Analysis of Product Reviews in Arabic.

3.5 Proceso de selección

En una primera fase de selección, se revisó los títulos y sus sumarios de los estudios recuperados y se eliminó aquellos que no cumplían con los criterios de inclusión y exclusión

En una segunda etapa del proceso de selección se recuperó y reviso el texto completo de los estudios restantes para verificar el cumplimiento de los criterios de inclusión y exclusión. Los estudios fueron evaluados según los criterios de calidad como:

- ¿Se indica claramente el objetivo del estudio?
- ¿El estudio indica su contribución en relación con el estado del arte actual?
- ¿Se describe con claridad la técnica propuesta?
- ¿Se discuten los resultados del estudio?
- ¿La discusión y conclusiones son claras y coherentes?

Cada pregunta tuvo tres opciones de respuesta que se muestran en la siguiente escala: si (1), no (0), parcialmente (0.5). Por lo tanto, los resultados variaron entre 0 (pésima calidad) y 8 (calidad óptima).

3.6 Revisión y discusión

En esta sección se presenta la revisión de investigaciones recientes sobre sistemas de recomendación y análisis de sentimiento. Se utilizó como principal buscador Scopus y la búsqueda se realizó entre los meses de abril y mayo de 2019.

Recommendation system based on data analysis-Application on tweets sentiment analysis (Nabil, Elbouhdidi, & Yassin, 2018)

En el presente artículo científico se busca presentar los diferentes tipos de enfoques en los que están basados los sistemas de recomendación, incluyendo sus ventajas y desventajas. Con ello, se tuvo como objetivo presentar su enfoque de recomendación basado en el análisis de sentimiento de los comentarios de los usuarios de Twitter.

La investigación comienza brindando una breve introducción de los enfoques de los sistemas de recomendación basados en colaboración y realiza lo respectivo con el enfoque basado en contenido y el enfoque híbrido, indicando en ambos casos sus ventajas y desventajas. Luego, muestran cómo usaron la herramienta Spark (*Framework* de código libre dedicado a *big data*) y el API de Twitter para extraer una gran cantidad de información generada en las redes sociales, Twitter. Una vez extraídos, se calcula la puntuación del *tweet* (positivo, negativo o neutro).

Finalmente, la salida que genera el presente trabajo es una puntuación (positivo, negativo o neutro) para los *tweets* y concluyen que el objetivo de su trabajo es usar estos resultados obtenidos a través de análisis de sentimiento para una futura recomendación.

Rating Predictor: Sentiment Analysis of Product Reviews in Arabic (Harrage, Alsalman, & Alqahtani, 2019)

El presente trabajo tiene como objetivo combinar los sistemas de recomendación con el análisis de sentimiento implementando una solución informática que permita predecir la puntuación de un producto considerando el sentimiento que llevan los comentarios de este producto.

Con respecto a los componentes del sistema, este cuenta con 3 los cuales son: Procesamiento de texto, análisis de sentimiento y predicción de la puntuación. En el primero, se preprocesa la información textual limpiándola de ruido, removiendo *stop words* y haciendo *stemming* (proceso a través del cual extraes la raíz de una palabra). Luego, en el segundo componente se analiza los sentimientos de los comentarios usando un algoritmo supervisado de aprendizaje de máquina, SVM. Finalmente, en el último componente se predice la puntuación del producto basado en múltiples entradas (puntuación directa y puntuación indirecta del comentario).

Finalmente, para la validación del análisis de sentimiento se usaron las métricas de exactitud, precisión, *recall*, medida-F y desviación de la raíz cuadrada media. Siendo el mejor resultado el de precisión, con 93.651%, el cual es la precisión del SVM. Para el tercer componente de predicción de las puntuaciones, se validó con el MAE, siendo este de 4.22.

Recommender System Through Sentiment Analysis (Ziani et al., 2017)

En esta investigación el objetivo principal fue combinar los sistemas de recomendación y el análisis de sentimiento de manera que se pueda generar recomendaciones más precisas para los usuarios. Para esto, implementaron un sistema de recomendación multilenguaje basado en filtros colaborativos.

La investigación empieza usando la similitud de Spearman para encontrar los usuarios similares al evaluado y con ello procede a filtrar la base de datos (limpiar la información, preprocesarla, extraer las características relevantes de la información textual). Luego, realiza

el análisis de opinión usando las características extraídas, clasificando estos comentarios usando máquinas de vectores de soporte semi supervisados en positivos, negativos y neutros. Como último punto, se usa la similitud de Spearman para obtener los usuarios más similares al evaluado y se generan los productos a recomendar ordenados descendentemente por el valor de su puntuación predicha (donde 1 es el más bajo y 10 es más alto).

Finalmente, como métricas de evaluación usaron el MAE, precisión y *recall*. Se puede observar los resultados en la tabla 6:

Tabla 6. Resultados de métricas de evaluación (Ziani et al., 2017)

TABLE I. THE EXPERIMENTAL RESULTS

	MAE	Precision	Recall
English	0.52	0.96	1.0
French	0.50	1.0	1.0
Arabic and dialect	0.60	0.90	1.0

Donde la cantidad de datos en inglés fue 2000 comentarios de 50 personas en 40 restaurantes, en francés 50 comentarios de 10 usuarios y en Arábigo 50 comentarios de 10 usuarios.

3.7 Conclusiones

Se concluye que, dadas las necesidades del entorno actual, es aconsejable el uso de un sistema de recomendación capaz de analizar los sentimientos no solo para ahorrar tiempo al usuario de no leer los millones de comentarios que pueda haber sobre un producto, sino para realizar una recomendación más acorde a sus necesidades y expectativas. He ahí la importancia del análisis textual, ya que un producto puede tener una puntuación general positiva, pero si dentro de las cualidades negativas que se menciona en los comentarios, se encuentra alguna que no cumpla con las necesidades del usuario, o simplemente no sea de su agrado, no sería un producto idóneo pese a que solo por su puntuación, pueda parecerlo a primera vista.

Capítulo 4. Experimentos y resultados obtenidos

4.1 Conjunto de datos

Este resultado se relaciona al objetivo 1 del trabajo, el cual consiste en caracterizar perfiles de usuarios con base en sus comentarios y puntuaciones sobre los productos ofrecidos en un *marketplace*, consistentes en los *reviews* de productos, de manera que se permita el análisis de estos. Para ello se hace uso de marcos de datos (*dataframes*) que facilitan la visualización de los mismos.

La ilustración 5 muestra un diagrama de entidad-relación entre los dos conjuntos de datos más importantes del trabajo: Productos y *Reviews* (comentarios). Esta corresponde a una relación de uno a muchos, puesto que un producto puede tener muchos *reviews*. Así mismo, se puede verificar que los datos a ser analizados corresponden a *reviews*, especialmente los atributos de "rating_producto" y "texto_review", los mismos que sirven para el proceso de recomendación.

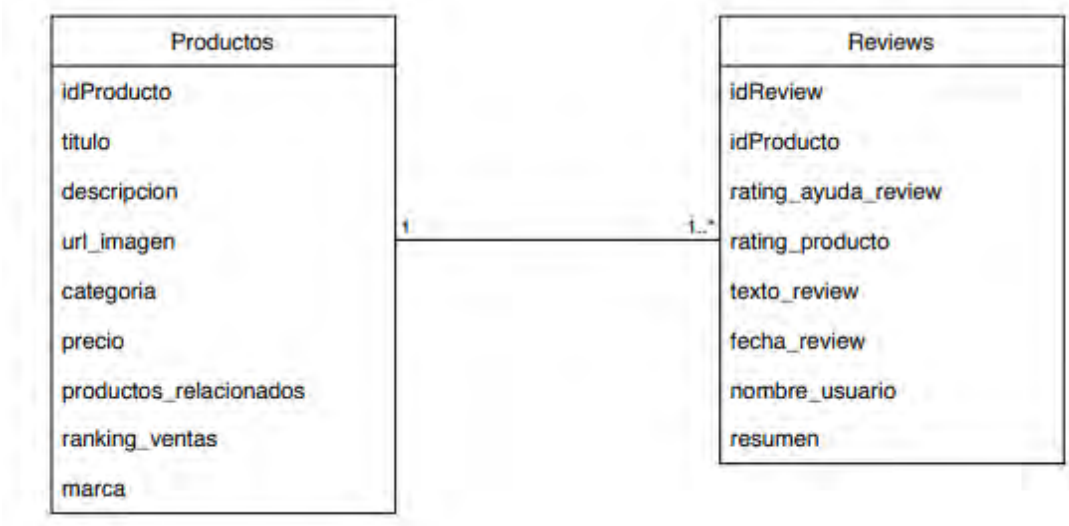


Ilustración 5. Diagrama de entidad-relación de los datos. (Elaboración propia).

Para observar la distribución de cantidad de comentarios por número de palabras se procede a graficar la información (ilustración 6), de manera que podamos tener un mejor entendimiento de los datos trabajados. Así podemos apreciar que la mayoría de los *reviews* de los usuarios

contiene entre 10 a 20 palabras, teniendo que cantidades mayores a estos van decreciendo su frecuencia de forma continua.

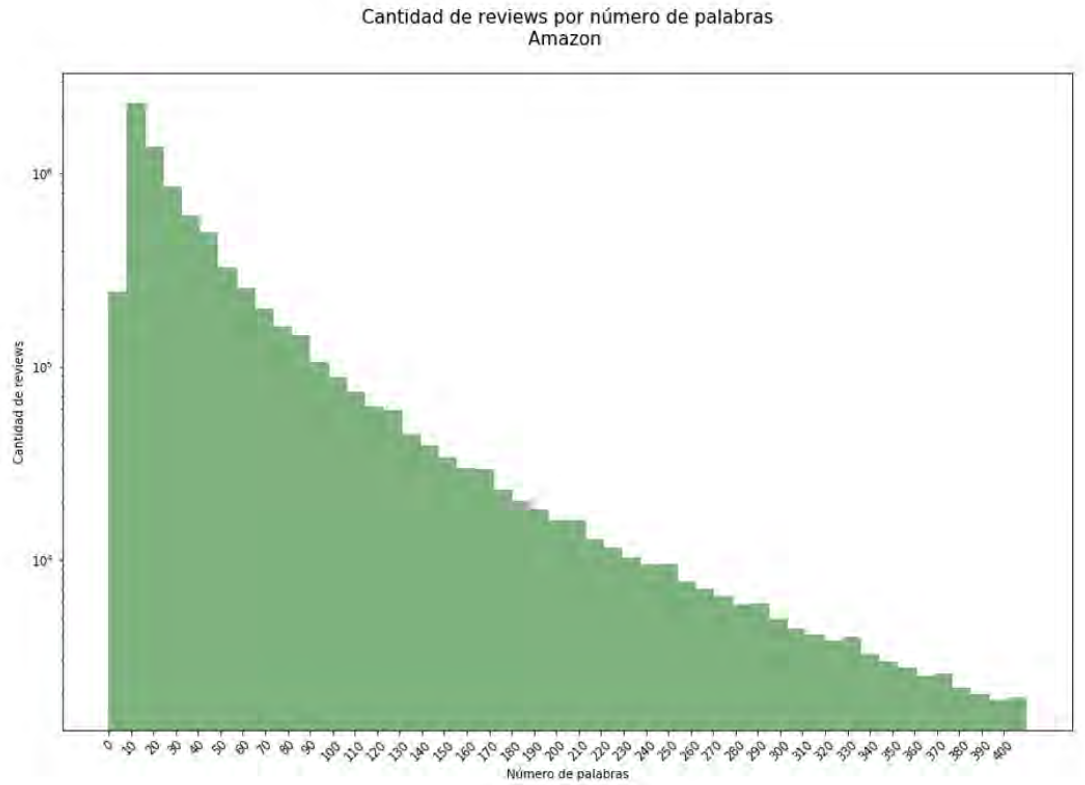


Ilustración 6. Distribución de la cantidad de comentarios por palabras. (Elaboración propia).

Se observa en la ilustración 7 el top 20 usuarios con número de comentarios realizados. Siendo el mayor de estos, un usuario con 520 comentarios. Así mismo, hay varios usuarios con un solo comentarios, razón por la cual el conjunto de datos fue filtrado en el tercer resultado a aquellos que tengan más de 10 comentarios, con el objetivo de poder contar con información más consistente.

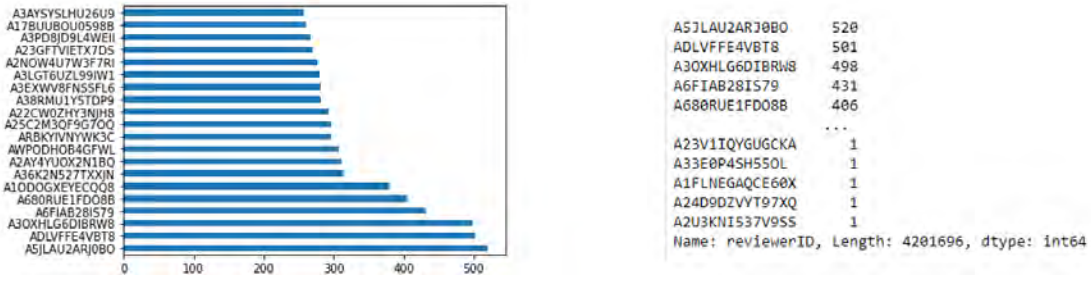


Ilustración 7. Distribución de la cantidad de comentarios por usuario. (Elaboración propia).

Esto indica que, de querer realizar una distribución de usuario por producto, esta matriz será dispersa dado que hay muchos más productos que la máxima cantidad de comentarios realizados por un usuario.

En la ilustración 8 se muestra la distribución de puntuaciones:

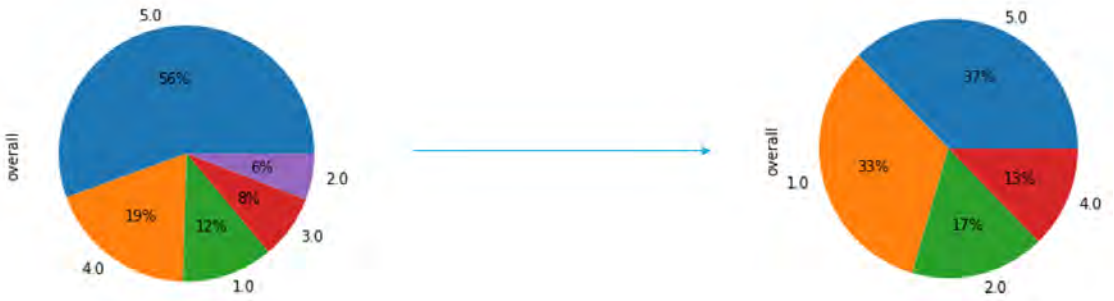


Ilustración 8. Distribución de la cantidad de puntuaciones en el conjunto de datos. (Elaboración propia).

El primer cambio que se hizo fue reducir aleatoriamente la información de 7 824 482 millones de comentarios a 2 716 174, siendo la mitad positiva y la otra mitad negativa (gráfico de la derecha).

Finalmente, se puede observar en la tabla 7 el *dataframe* resultante (id del *review*, id del producto, puntuación):

Tabla 7. Conjunto de datos de puntuaciones procesado. (Elaboración propia).

	reviewerID	asin	score
189	A2IDCSC6NVONIZ	0972683275	5
200	A3BMUBUC1N77U8	0972683275	4
262	AYQNWE3AX4H08	0972683275	5
274	AQBLWW13U66XD	0972683275	5
283	AUKEU9CW56TT4	0972683275	5

El cual fue usado para el posterior análisis.

4.2 Preprocesamiento de comentarios de usuarios en un *marketplace*

Este resultado se relaciona con el objetivo 1 del trabajo, el cual consiste en caracterizar perfiles de usuarios con base en sus comentarios y puntuaciones sobre los productos ofrecidos en un *marketplace*. Para ello se utilizó marcos de datos (*dataframes*) que facilitan la visualización de los mismos.

En esta parte se procedió a preprocesar la información textual de los comentarios para poder:

- Eliminar información no útil (*stopwords*)
- Remover las puntuaciones
- Remover comillas
- Remover 2 o más espacios en blanco contiguos
- Lematización de las palabras (con la consideración de que estas tuvieran como mínimo, 2 letras)
- Convertir a minúscula el texto

La ilustración 9 muestra el procedimiento con un ejemplo del conjunto de datos.

Dado el primer comentario, se tiene el siguiente conjunto de datos antes de preprocesar la información (ilustración 9):

	reviewerID	asin	reviewerName	helpful	reviewText	overall	summary	unixReviewTime	reviewTime
0	AKM1MP6P0OYPR	0132793040	Vicki Gibson "momo4"	[1, 1]	Corey Barker does a great job of explaining Bl...	5.0	Very thorough	1365811200	04 13, 2013
1	A2CX7LUOHB2NDG	0321732944	Bernie	[0, 0]	While many beginner DVDs try to teach you ever...	5.0	Adobe Photoshop CS5 Crash Course with master P...	1341100800	07 1, 2012
2	A2NWSAGRHCP8N5	0439886341	bowmans2007	[1, 1]	It never worked. My daughter worked to earn th...	1.0	absolutely horrible	1367193600	04 29, 2013
3	A2WNBOD3WNDNKT	0439886341	JAL	[1, 1]	Some of the functions did not work properly. ...	3.0	Disappointing	1374451200	07 22, 2013
4	A1GI0U4ZRJA8WN	0439886341	Truthfull	[4, 4]	Do not waste your money on this thing it is te...	1.0	TERRIBLE DONT WASTE YOUR MONEY	1334707200	04 18, 2012

```
0    Corey Barker does a great job of explaining Bl...
1    While many beginner DVDs try to teach you ever...
2    It never worked. My daughter worked to earn th...
3    Some of the functions did not work properly.  ...
4    Do not waste your money on this thing it is te...
```

Ilustración 9. Conjunto de datos de comentarios sin procesar. (Elaboración propia).

Luego de preprocesar la información (ilustración 10):

	reviewerID	asin	reviewerName	helpful	reviewText	overall	summary	unixReviewTime	reviewTime
0	AKM1MP6P0OYPR	0132793040	Vicki Gibson "momo4"	[1, 1]	corey barker great job explain blend modes dvd...	5.0	Very thorough	1365811200	04 13, 2013
1	A2CX7LUOHB2NDG	0321732944	Bernie	[0, 0]	many beginner dvds try teach everything know p...	5.0	Adobe Photoshop CS5 Crash Course with master P...	1341100800	07 1, 2012
2	A2NWSAGRHCP8N5	0439886341	bowmans2007	[1, 1]	never work daughter work earn money get never ...	1.0	absolutely horrible	1367193600	04 29, 2013
3	A2WNBOD3WNDNKT	0439886341	JAL	[1, 1]	function work properly daughter buy money disa...	3.0	Disappointing	1374451200	07 22, 2013
4	A1GI0U4ZRJA8WN	0439886341	Truthfull	[4, 4]	waste money thing terrible boutght product son...	1.0	TERRIBLE DONT WASTE YOUR MONEY	1334707200	04 18, 2012

```
0    corey barker great job explain blend modes dvd...
1    many beginner dvds try teach everything know p...
2    never work daughter work earn money get never ...
3    function work properly daughter buy money disa...
4    waste money thing terrible boutght product son...
```

Ilustración 10. Conjunto de datos de comentarios procesado. (Elaboración propia).

Donde en ella se puede observar que el contenido del comentario ha cambiado según el preprocesamiento aplicado. Se procederá a mostrar el texto original del primer comentario y el texto preprocesado final en la tabla 8:

Tabla 8. Comparación de comentarios (Elaboración propia).

Antes del preprocesamiento	Después del preprocesamiento
Corey Barker does a great job of explaining Blend Modes in this DVD. All of the Kelby training videos are great but pricey to buy individually. If you really want bang for your buck just subscribe to Kelby Training online.	corey barker great job explain blend modes dvd kelby train videos great pricey buy individually really want bang buck subscribe kelby train online

Se observa que las palabras están en minúscula, así como también se han eliminado los signos de puntuación y se ha lematizado las palabras (explaining -> explain) y se han eliminado *stopwords* (Corey Barker does **a** great job -> corey barker great job).

Por lo cual, se ha ejecutado correctamente el preprocesamiento y la información está lista para ser utilizada para los análisis posteriores.

4.3 Algoritmo de recomendación de productos

Este resultado se relaciona al objetivo 2 del trabajo, el cual consiste en predecir productos a recomendar usando solo las puntuaciones otorgadas por los usuarios del *marketplace* como modelo base. Para ello, se usó métricas como el error medio absoluto (MAE por sus siglas en inglés) para validar la efectividad del mismo.

Para este resultado se usó como entrada los datos obtenidos en el resultado 1, representándolos en forma de matriz pivote obteniendo una matriz de usuarios por productos (tabla 9):

Tabla 9. Matriz pivote original. (Elaboración propia).

asin	0972683275	1400501466	1400501520	1400532620	1400532655	140053271X	1400599997	1400698987	3744295508	9573212919	...
reviewerID											
A0251761JI35FM4C8VK2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...
A02712303HM5RXRLNJUB7	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...
A0279100VZXR9A2495P4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...
A0284208PB0CNSHI1OC6	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...
A02970121VCH64N53W9F4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...

El objetivo es llenar esta matriz prediciendo las puntuaciones que le daría un usuario a determinado producto, para ello se procedió (en conjunto con el asesor) a disminuir el conjunto de datos a aquellos usuarios y productos que tuvieran más de 10 comentarios cada uno con el fin de evitar el problema del *cold start*. Es así que la matriz resultante tiene 51322 usuarios × 17156 productos (600 mil comentarios aproximadamente). Posteriormente, se utilizó y comparó 2 formas de llenado de nulos en la matriz, la primera reemplazándolos por ceros y la segunda por valores aleatorios entre 1 a 5. El primero tuvo un resultado de error rmse de 0.90 mientras que el último un rmse de 0.93, considerando estos resultados y el tiempo de ejecución de ambos, se decidió quedarnos con la mejor métrica; es decir, llenar los nulos con ceros.

Luego, se usaron los métodos de matriz de factorización no negativa (representada en la ilustración 11) y gradiente descendente estocástica para poder predecir los valores.

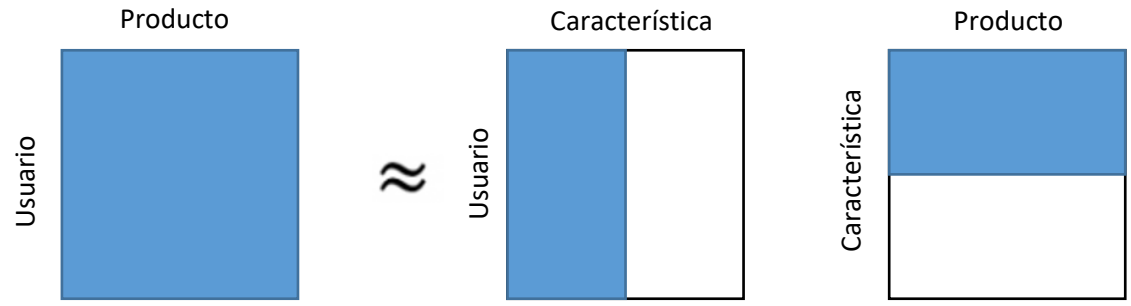


Ilustración 11. Explicación matriz de factorización no negativa. (Elaboración propia).

La matriz de factorización no negativa sirve para representar la información (matriz original) en 2 matrices más pequeñas con el objetivo de reducir la dimensionalidad y que se pueda reconstruir de la mejor forma posible la matriz original a partir de las 2 tablas generadas. Para ello, es importante definir k, donde k es el número de características latentes a usar (características que posee el conjunto de datos, teniendo en cuenta que, si k es muy pequeño, el modelo será *underfitting* y si k es muy alto, el modelo será *overfitting*).

Finalmente, el método de gradiente descendente estocástica permite actualizar los pesos del algoritmo de manera que en cada iteración se calcula el error entre el valor predicho y el valor original de la puntuación, se recalcula la gradiente y se actualizan los valores de los pesos con el objetivo de ir reduciendo el error de la predicción en cada iteración.

Finalmente, en la tabla 10 se muestran los resultados del algoritmo, en ella se puede apreciar que todos los valores faltantes fueron predichos por el modelo.

Tabla 10. Matriz pivote después de predecir puntuaciones. (Elaboración propia).

asin / reviewerID	0972683275	1400501466	1400501520	1400532620	1400532655	140053271X	1400599997	1400698987	3744295508	9573212919	...
A0251761JI35FM4C8VK2	4.706650	4.508446	4.792036	4.324033	4.051137	4.179322	4.270046	4.761762	5.038861	4.433788	...
A02712303HM5RXRLNJUB7	4.741503	4.542646	4.809396	4.332911	4.050438	4.192832	4.301733	4.797574	5.048815	4.460074	...
A0279100VZXR9A2495P4	4.495445	4.293722	4.537172	4.098185	3.830920	3.940022	4.049890	4.527058	4.796654	4.233025	...
A0284208PB0CNSHI1OC6	4.523478	4.336636	4.579095	4.146067	3.838661	3.968005	4.089254	4.531731	4.864285	4.256095	...
A02970121VCH64N53W9F4	4.573627	4.378821	4.604627	4.180112	3.873393	4.061425	4.102780	4.594836	4.877304	4.266451	...

Métricas de error

RMSE = 0.9252657162812686

MAE = 0.6759364201110944

En conclusión, para entender el valor del RMSE de 0.92 con un ejemplo, dado el valor original de una puntuación de 3, en promedio el valor predicho es 3.92 o 2.08.

4.4 Primer reporte de análisis de resultados

Este resultado está relacionado al objetivo 2 del trabajo, el cual consiste en predecir productos a recomendar usando solo las puntuaciones otorgadas por los usuarios del *marketplace* como modelo base. Para ello, se elaboró un reporte de análisis de resultados para evidenciar los parámetros usados, la elección de hiper-parámetros y resultado del algoritmo.

El reporte consta de la presentación de los datos iniciales, distribución y estructura de los mismos. Posteriormente se explica cómo se trató la información y el procedimiento seguido. Luego, se evidencia cómo se validó el algoritmo mediante una técnica estadística conocida como validación cruzada. Finalmente, se presentan los resultados obtenidos.

Uno de los hiper-parámetros más importantes es la elección de k (características latentes), en la ilustración 12 se presenta la evolución del RMSE con respecto a k.

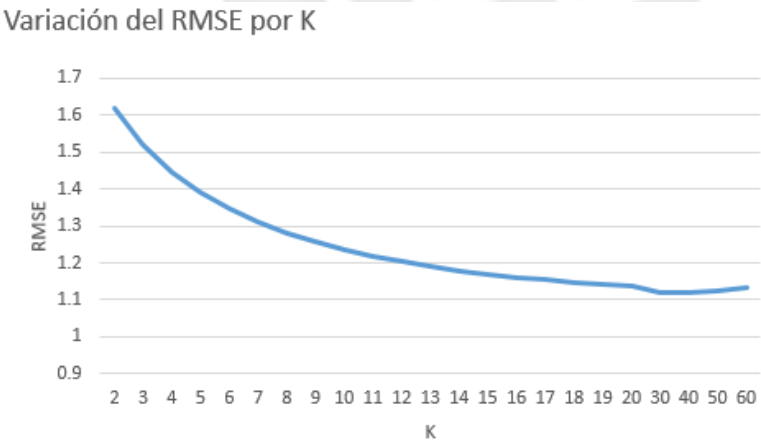


Ilustración 12. Evolución de RMSE con respecto a k. (Elaboración propia).

Con este gráfico, determinamos el valor óptimo de k con base en la información preprocesada, escogiendo así k = 10 y actualizando los valores del RMSE y MAE, para mayor información e informe completo, revisar el anexo B.

Es importante señalar que el módulo para la recomendación de productos con base en la similitud de usuarios será presentado en el resultado 7.

4.5 Cálculo de la polaridad

Este resultado está relacionado el objetivo 3, el cual consiste en predecir la polaridad de los comentarios de los usuarios. Para ello, se utilizaron las métricas de precisión y *recall* para comprobar la efectividad del algoritmo.

Para este resultado, se procedió a:

- Usar lo que se obtuvo en el resultado 2
- Usar Word2Vec para expresar la información textual como vectores
- Usar la técnica de *Random Forest* para clasificar y hallar la polaridad de los comentarios

Word2Vec

Es una técnica que convierte los datos textuales en datos numéricos (vectores), tratando de mantener las características y relaciones semánticas entre palabras.

Por ejemplo, en la ilustración 13 se puede observar cómo las palabras tienen una relación entre ellas (ddr2, 3 y 4 son tipos de RAM, i3, 5 y 7 tipos de procesadores y 980ti y Nvidia son relativos a tarjetas de video).

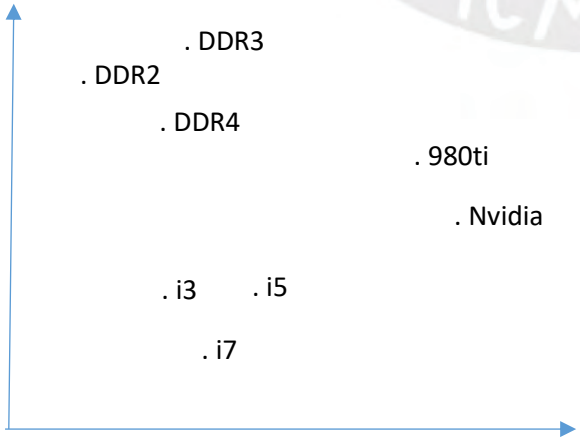


Ilustración 13. Representación gráfica del Word2Vec. (Elaboración propia).

Los parámetros usados son:

- Tamaño del vector = 300

  Significa que cada palabra será representada en un vector de tamaño 300 con el objetivo de tratar de mantener la mayor cantidad de características semánticas y relacionales de esta.

- Cuenta mínima = 8

  Significa que, del conjunto de datos textuales, se usaron aquellas palabras que se repitan como mínimo 8 veces a lo largo de este. De haber palabras que se repitan menos de 8 veces, no serán consideradas en el análisis y por ende, no tendrán una representación vectorial.

- Trabajadores = 12

  Este parámetro determina el número de hilos simultáneos que se estarán usando al momento de generar los vectores, se escogió 12 en base al número de hilos que puede manejar el procesador usado.

Random forest

Es un método de aprendizaje conjunto para clasificación que añade una capa adicional de aleatoriedad al *bagging*. Adicionalmente a construir cada árbol usando una diferente muestra de los datos, este método cambia cómo los árboles de regresión o clasificación son construidos (Liaw, A., & Wiener, M., 2002). La ilustración 14 muestra un ejemplo de cómo 3 de 5 árboles predijeron que la categoría resultante sería “Blue”.

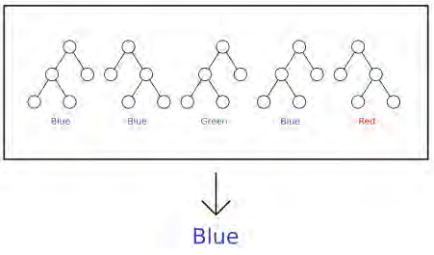


Ilustración 14. Representación del random forest. Adaptado de (https://www.edureka.co/blog/artificial-intelligence-algorithms/, fecha de consulta: 13/05/2020).

Con este método, se clasificaron los comentarios en positivos y negativos, dado que es óptimo al usar una gran cantidad de información. Entonces, la forma en que este algoritmo opera es que existen varios árboles (factores decisores) que generan un resultado independiente (positivo o negativo) ante un comentario y se escogerá como categoría final aquella que haya sido el resultado final de una mayor cantidad de árboles.

Finalmente, con 2 716 174 comentarios (mitad positivo y mitad negativo), se obtuvieron los resultados expuestos en la tabla 11 y 12:

Tabla 11. Resultados predicción de la polaridad negativa. (Elaboración propia)

Predecir negativos	Antes de preprocesar	Después de preprocesar
Precisión	83	85
Recall	85	87
Medida F1	74	86
Accuracy	84	86

Tabla 12. Resultados predicción de la polaridad positiva. (Elaboración propia)

Predecir positivos	Antes de preprocesar	Después de preprocesar
Precisión	85	87
Recall	83	85
Medida F1	84	86
Accuracy	84	86

Se observa un incremento en las métricas utilizadas, lo cual evidencia la importancia del preprocesamiento de los comentarios.

4.6 Algoritmo de recomendación de productos usando puntuaciones y extracción de la polaridad

El objetivo de este resultado es añadir la extracción de la polaridad descrita en 4.5 al algoritmo base propuesto en 4.3 de manera de que se consiga reducir el error entre el valor predicho y el valor real.

Se definió la siguiente fórmula para actualizar los valores de la matriz de puntuación obtenida en R3:

$$newx = x + a * abs(round(x) - x)/2$$

Donde:

- *newx*: Es el nuevo valor predicho de la puntuación
- x = El antiguo valor predicho de la puntuación
- a = Será 1 o -1 dependiendo de si es positivo o negativo (valor resultante de la polaridad obtenida a través de la obtención de la polaridad).

Con ello tenemos que, si el algoritmo predijo una puntuación de 3.40 y la extracción de la polaridad sobre el comentario que dejó el usuario a ese producto resulta en una negativa, se reducirá el valor de 3.40 a 3.20 ya que a tomaría el valor de -1, lo cual indica una disminución del valor predicho y esta cantidad de disminución será el valor absoluto de (3.40 – 3.00) / 2; es decir, 0.20.

Una vez actualizada la matriz, se procedió a realizar el módulo de recomendación de productos dado un usuario. Para ello, se trabajó con 10 mil usuarios y se obtuvo lo siguiente:

1) Se calculó la similitud entre los usuarios mediante la similitud de Pearson con el fin de determinar cuáles son los usuarios más similares entre sí teniendo como base el histórico de compras.
2) Con la lista obtenida en 1, se obtuvieron los productos que estos usuarios habían comprado y otorgado una puntuación mayor igual a 4.
3) Luego, con la lista obtenida en 2, se filtró aquellos productos que el usuario no hubiera comprado antes; es decir, filtrado de manera que todos los productos fueran nuevos para el usuario.

4) Finalmente, se obtuvo la puntuación de la matriz actualizada de la lista de productos obtenida en 3 y se ordenó descendentemente, obteniendo así el top n productos para el usuario objetivo.

Finalmente, en este punto se tuvo la matriz actualizada con la polaridad extraída de los comentarios y el módulo de recomendación de productos. A continuación, se presenta la métrica utilizada para evaluar el modelo y una comparación del RMSE entre el algoritmo base (R3) vs el presente con las mismas 10 muestras aleatorias para ambos (tabla 13):

$$RMSE = \sqrt{\frac{1}{n} + \sum_{x=1}^{n} (newx - x)^2}$$

Tabla 13. Comparación RMSE algoritmo base vs propuesto. (Elaboración propia)

Algoritmo base (x)	Algoritmo propuesto (y)
0.9021673162410346	0.8571659946693936
0.9084158671375143	0.8638309401728261
0.908367203420963	0.8639422065913938
0.9132030967819319	0.8682152407262937
0.9213746662572685	0.8760991721293383
0.91830605354168	0.87290556256022
0.9202644504923404	0.8747059768962058
0.918461683546362	0.8731402298607015
0.9182307125362127	0.8728698922170688
0.9187888083444719	0.8732866518559279

Módulo de recomendación:

Este consta de 2 parámetros, el id del usuario a recomendar y el número de productos recomendados que se desea obtener. El resultado serán 3 conjunto de datos, siendo el primero una lista con el id del producto y el valor predicho por el algoritmo que el usuario le pondrá a este producto de comprarlo (ilustración 15).

```
puntuaciones, prod_recomendados, prod_comprados = recommend_products('A0251761JI35FM4C8VK2', 5)
```

```
puntuaciones
```

```
[['B002WE6D44', 5.129815742490696],
 ['B009O7YUF6', 5.099451501979183],
 ['B00768SBAU', 5.008565742265273],
 ['B004Q0PTD8', 4.85050086097573],
 ['B008R6OPJQ', 4.787568250504141]]
```

Ilustración 15. Top recomendaciones (Elaboración propia).

El segundo elemento será la información de los productos recomendados, donde se puede observar la descripción del producto, a qué categoría pertenece, el título del producto, su precio en dólares, el ranking de ventas que tiene, otros productos relacionados y la marca del producto (tabla 14).

Tabla 14. Top detalle de productos recomendados (Elaboración propia).

```
prod_recomendados
```

	asin	imUrl	description	categories	title	price	salesRank	related	brand
31256	B0002QYS8W	http://ecx.images-amazon.com/images/I/11N68FA8...	Amazon.com Product Description As all-in-one s...	[[Electronics, Car & Vehicle Electronics, Car ...	Bazooka BTA8100 BT Series 8-Inch 100-Watt Ampl...	185.96	NaN	{'also_bought': ['B001JT33RI', 'B000BJJXZI', '...	Bazooka
43187	B000A0UHXU	http://ecx.images-amazon.com/images/I/31GX9Ssj...	The Sigma 70-300mm f4-5.6 DG Macro Lens for Ni...	[[Electronics, Camera & Photo, Lenses, Camera ...	Sigma 70-300mm f/4-5.6 DG Macro Telephoto Zoom...	144.00	NaN	{'also_bought': ['B00004ZCJI', 'B001HSXOZC', '...	Sigma
124237	B001NJ0D0Y	http://ecx.images-amazon.com/images/I/51AE940%...	Cooler Master Hyper N520 RR-920-N520-GP CPU Fa...	[[Electronics, Computers & Accessories, Comput...	Cooler Master Hyper N520 - CPU Cooler with Cop...	36.52	{}	{'also_bought': ['B000OGX5AM', 'B009O7YUF6', '...	Cooler Master
133279	B001V9KG0I	http://ecx.images-amazon.com/images/I/41y5I9nq...		[[Electronics, Camera & Photo, Bags & Cases, C...	Case Logic TBC-302 FFP Compact Camera Case (Bl...	4.99	{}	{'also_bought': ['B007BJHETS', 'B00B5HE2UG', '...	NaN
162157	B002WE6D44	http://ecx.images-amazon.com/images/I/41v5MQEG...		[[Electronics, Computers & Accessories, Cables...	Transcend 8 GB Class 10 SDHC Flash Memory Card...	7.52	{}	{'also_bought': ['B009SQQF9C', 'B0073HSJGU', '...	NaN

Finalmente, podemos contrastar los productos recomendados con el tercer elemento, los productos comprados por el usuario (tabla 15). En el cual se observan los mismos campos que el segundo elemento y podemos notar, por ejemplo, como producto recomendado un *cooler* de CPU de la marca Cooler Master, un producto que va bien con el Intel Core i7-3770k que compró el usuario ya que al ser de la serie k, necesita un *cooler* diferente al base.

Cabe resaltar que al ser una recomendación basada en filtros colaborativos, la cual está basada en lo que otros usuarios similares a ti han comprado, es una recomendación que no está sesgada a un solo topico de producto ni especializada en una sola marca, sino que converva la variedad de ofrecer diversos productos para el usuario, dándole la libertad de escoger el producto que se acomode mejor con sus necesidades personales.

Tabla 15. Top productos comprados (Elaboración propia).

```
prod_comprados
```

	asin	imUrl	description	categories	title	price	salesRank	related	brand
37195	B0007MWE1E	http://ecx.images-amazon.com/images/I/41wclQUj...	Cables Unlimiteds high quality HDMI to DVI cab...	[[Electronics, Accessories & Supplies, Audio &...	Cables Unlimited PCM-2296-06 HDMI to DVI D Cab...	5.49	NaN	{'also_bought': ['B0007MWE14', 'B0035B4LJM', '...	Cables Unlimied
146404	B002HK8TE0	http://ecx.images-amazon.com/images/I/415zsr3A...	NaN	[[Electronics, Computers & Accessories, Cables...	WHITE 100FT CAT6 CAT 6 RJ45 PATCH ETHERNET NET...	11.31	{}	{'also_bought': ['B0083X8VZW', 'B000I97FHY', '...	Citi Electronics
164003	B002YIG9AQ	http://ecx.images-amazon.com/images/I/31VtOscF...	Lite-On Super AllWrite IHAS124-04 24X SATA DVD...	[[Electronics, Computers & Accessories, Comput...	Lite-On Super AllWrite 24X SATA DVD+/-RW Dual ...	20.64	{}	{'also_bought': ['B0088PUEPK', 'B0092ML1SC', '...	Lite-On
172935	B0036Q7MV0	http://ecx.images-amazon.com/images/I/41y2cKTh...		[[Electronics, Computers & Accessories, Data S...	Western Digital WD1002FAEX Caviar Black 1 TB S...	155.00	{}	{'also_bought': ['B0033Z2BAQ', 'B00CRJSX5Q', '...	Western Digital
240756	B004MYFOE2	http://ecx.images-amazon.com/images/I/51mq6guF...		[[Electronics, Computers & Accessories, Comput...	Corsair Cooling Hydro-Series All-in-One High-P...	80.92	NaN	{'also_bought': ['B007SZ0EOW', 'B006EWUO22', '...	Corsair
303510	B006EWUO22	http://ecx.images-amazon.com/images/I/51MXy7QS...		[[Electronics, Computers & Accessories, Comput...	Corsair Vengeance 16GB (2x8GB) DDR3 1600 MHz...	184.77	{}	{'also_bought': ['B00CO8TBQ0', 'B00KPRWAX8', '...	Corsair
323702	B007G51UWY	http://ecx.images-amazon.com/images/I/51sc57VL...	The ASUS P8Z77-V motherboard features the Inte...	[[Electronics, Computers & Accessories, Comput...	ASUS P8Z77-V LGA 1155 Intel Z77 HDMI SATA 6Gb/...	164.84	{}	{'also_bought': ['B007SZ0E1K', 'B007SZ0EOW', '...	Asus
333737	B007SZ0EOW	http://ecx.images-amazon.com/images/I/51PLu0g6...	Intel BX80637I73770K Core i7-3770K Ivy Bridge ...	[[Electronics, Computers & Accessories, Comput...	Intel Core i7-3770K Quad-Core Processor 3.5 GH...	319.95	{}	{'also_bought': ['B007KTY4A6', 'B007G51UWY', '...	Intel
355252	B008HD3CTI	http://ecx.images-amazon.com/images/I/319FKX%2...	The Define R4 is the latest in the Define seri...	[[Electronics, Computers & Accessories, Comput...	Fractal Design Define R4 Cases, Black Pearl (F...	120.78	NaN	{'also_bought': ['B008YH1AV4', 'B00CO8TBQ0', '...	Fractal Design
381188	B009NHAEXE	http://ecx.images-amazon.com/images/I/41yhL6Ui...		[[Electronics, Computers & Accessories, Data S...	Samsung MZ-7TD250BW 840 Series Solid State Dri...	211.99	{}	{'also_bought': ['B002BH3Z8E', 'B00E3W1726', '...	Samsung

4.7 Segundo reporte de análisis de resultados

En el resultado anterior se presentó los resultados de las métricas obtenidas del algoritmo base vs el algoritmo propuesto; sin embargo, ¿es posible afirmar que uno es mejor que otro solo basado en ello?

La respuesta es no, dado 2 algoritmos que son aplicados sobre el mismo conjunto de datos, se debe verificar mediante experimentos la hipótesis de que uno es consistentemente mejor que el otro (Dror R., Baumer G., Shlomov S., Reichart R., 2018).

De acuerdo al artículo científico expuesto anteriormente, se realizó la prueba de Shapiro para determinar la normalidad en la distribución de los datos y posteriormente, la prueba de t-test para datos pareados (puesto que se comparó 2 algoritmos sobre el mismo conjunto de datos) para determinar si el algoritmo propuesto era mejor que el base.

Con los 10 mil usuarios del R6, se realizó las siguientes pruebas:

Shapiro test:

Con un nivel de significancia del 5%:

H0 = los datos siguen una distribución normal

H1 los datos no siguen una distribución normal

p-value = 1.0, no se rechazó H0, los datos siguen una distribución normal

T-test:

Con un nivel de significancia del 5%:

H0 = la media de X es mayor igual a la media de Y

H1 = la media de Y es menor a la media de X

p-value: 1.93352462824e-20, se rechaza H0, la media de Y es menor a la media de X.

Para mayor información e informe completo, revisar el anexo C.

Capítulo 5. Conclusiones y trabajos futuros

5.1 Conclusiones

Con respecto al objetivo 1 (caracterizar perfiles de usuarios con base en sus comentarios y puntuaciones sobre los productos ofrecidos en un *marketplace)* se concluye que:

- Para evitar el problema del *cold start*, se redujo el conjunto de datos a aquellos usuarios y productos que tuvieran más de 10 comentarios cada uno.
- Se debe preprocesar la información textual puesto que ello conlleva a mejores en el algoritmo de predicción de la polaridad.

Así mismo, con respecto al objetivo 2 (predecir productos a recomendar usando solo las puntuaciones otorgadas por los usuarios del *marketplace* como modelo base) se concluye que:

- El valor más adecuado del hiper-parámetro de cantidad de características latentes, k, es 10 con base en la experimentación sobre este conjunto de datos, la cual se presentó y validó en el documento de análisis de resultados (validación cruzada).
- Es adecuado el uso del método de matriz de factorización no negativa con gradiente descendente estocástica para tratar con los datos dispersos del conjunto de datos y poder generar predicciones con base en ello, siendo las métricas de RMSE y MAE de 0.913561484915 y 0.6676463255109276 respectivamente con k = 10.

Luego, con respecto al objetivo 3 (predecir la polaridad de los comentarios de los usuarios), se concluye que:

- Es importante balancear la cantidad de datos de las categorías a predecir puesto que, de lo contrario, el algoritmo estará inclinado a elegir la categoría que más datos tenga, razón por la cual se utilizó la misma cantidad de comentarios positivos y negativos.
- Se probaron 2 algoritmos, Máquina de vectores de soporte (SVM por sus siglas en inglés) y *Random Forest*. Dada la gran cantidad de datos textuales a entrenar y las

características intrínsecas del primer algoritmo, el tiempo de entrenamiento era demasiado alto. Es así que se escogió el segundo algoritmo con base en su óptimo tiempo de entrenamiento, obteniéndose los siguientes resultados expuestos en la tabla 16:

Tabla 16. Resultados finales predicción polaridad. (Elaboración propia)

	Negativo	Positivo
Accuracy	86	86
Precisión	85	87
Recall	87	85
Medida F1	86	86

Finalmente, con respecto al objetivo 4 (predecir productos a recomendar usando las puntuaciones y comentarios de evaluaciones de los usuarios, comparándolo con el modelo base), se combinó exitosamente la recomendación basada en puntuaciones con la extracción de la polaridad realizada sobre los comentarios de los usuarios para mejorar la precisión de la predicción, siendo medido por métricas de error como RMSE y MAE.

Así mismo, se comprobó estadísticamente la mejora en la recomendación contra el algoritmo base implementado en el objetivo 2, el cual incluía la recomendación basada solo en las puntuaciones. Es así que los resultados, con un nivel de significancia del 5%, demostraron que la inclusión de la polaridad tenía un error significativamente menor al algoritmo base.

Luego, se implementó un módulo para la recomendación de productos basada en filtros colaborativos y la similitud de Pearson. Dicho módulo recibe 2 parámetros, el id del usuario a recomendar y el número de recomendaciones que desea obtener. La ventaja de usar este método es que no está sesgado a recomendar solo una categoría de productos. Por ejemplo, dado un usuario “a” que solo ha comprado cpu’s y un usuario “b” que compró varios de estos cpu’s con sus respectivas placas madres, el algoritmo generará una variedad de productos recomendados tal que de recomendarle productos al usuario “a”, no solo sean cpu’s sino también otros

productos como placa madres con base en los comentarios y compras de una comunidad similar a él (usuario “b”).

5.2 Trabajos futuros

El presente trabajo fue hecho con base en productos electrónicos, como trabajo futuro se plantea poder ampliar el análisis sobre otros tópicos como vestimenta, libros, canciones, entre otros. También, experimentar con otros modelos de recomendación que sean más acorde con los conjuntos de datos a analizar.

Así mismo, se plantea realizar una visualización a nivel web del módulo de recomendación.

Referencias

Aguilar, M., & Zapata, C. (2016). Integrating UCD and an Agile Methodology in the Development of a Mobile Catalog of Plants. En M. Soares, C. Falcão, & T. Z. Ahram (Eds.), *Advances in Ergonomics Modeling, Usability & Special Populations: Proceedings of the AHFE 2016 International Conference on Ergonomics Modeling, Usability & Special Populations, July 27-31, 2016, Walt Disney World®, Florida, USA* (pp. 75–87). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-41685-4_8

Hernández Sampieri, R., Fernández Collado, C., & Baptista, M. (2014). *Metodología de la investigación* (6ta ed.). México D.F.: McGraw-Hill.

Krusche, S., & Bruegge, B. (2014). User feedback in mobile development. En *MobileDeLi 2014 - Proceedings of the 2nd International Workshop on Mobile Development Lifecycle, Part of SPLASH 2014* (pp. 25–26). Recuperado de http://www.scopus.com/inward/record.url?eid=2-s2.0-84921489617&partnerID=40&md5=0198a5715bec4bf6608454ce82f098d2

Leotta, M., Beux, S., Mascardi, V., & Briola, D. (2015). My MOoD, a Multimedia and Multilingual Ontology Driven MAS: Design and First Experiments in the Sentiment Analysis Domain. *Proceedings of the 2Nd International Conference on Emotion and Sentiment in Social and Expressive Media: Opportunities and Challenges for Emotion-aware Multiagent Systems - Volume 1351*, 51–66. Recuperado de http://dl.acm.org/citation.cfm?id=3054101.3054109

Bobadilla, J., Ortega, F., Hernando, A., & GutiéRrez, A. (2013). Recommender Systems Survey. *Know.-Based Syst.*, *46*, 109–132. https://doi.org/10.1016/j.knosys.2013.03.012

Ait Hammou, B., & Ait Lahcen, A. (2017). FRAIPA: A fast recommendation approach with improved prediction accuracy. *Expert Systems with Applications*, *87*, 90-97.

https://doi.org/10.1016/j.eswa.2017.06.001

A. Kitchenham, B. (2007). *Kitchenham, B.: Guidelines for performing Systematic Literature Reviews in software engineering. EBSE Technical Report EBSE-2007-01*.

Poggi, O., & Augusto, C. (2016). Revisión sistemática sobre la aplicación de ontologías de dominio en el análisis de sentimiento. *Pontificia Universidad Católica del Perú*. Recuperado de http://tesis.pucp.edu.pe/repositorio//handle/20.500.12404/7514

Thelwall, M., Buckley, K., & Paltoglou, G. (2012). Sentiment Strength Detection for the Social Web. *Journal of the American Society for Information Science and Technology*, *63*, 163-173. https://doi.org/10.1002/asi.21662

Liu, B. (2015). *Sentiment Analysis: Mining Opinions, Sentiments, and Emotions*. Cambridge: Cambridge University Press. doi:10.1017/CBO9781139084789

Ekman, P. (1994). All Emotions Are Basic. In Ekman, P. & Davidson, R. (Eds.), The Nature of Emotion: Fundamental Questions (pp. 15-19). New York: Oxford University Press.

Wei, W., & Gulla, J. A. (2010). Sentiment Learning on Product Reviews via Sentiment Ontology Tree. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, 404–413. Recuperado de http://dl.acm.org/citation.cfm?id=1858681.1858723

Picard, R. W. (2000). *Affective Computing*. MIT Press.

Cambria, E., Mazzocco, T., & Hussain, A. (2013). Application of multi-dimensional scaling and artificial neural networks for biologically inspired opinion mining. Biologically Inspired Cognitive Architectures, 4, 41-53.

Su, X., & Khoshgoftaar, T. M. (2009). A Survey of Collaborative Filtering Techniques. *Adv. in Artif. Intell.*, *2009*, 4:2–4:2. https://doi.org/10.1155/2009/421425

Liu, H., He, J., Wang, T., Song, W., & Du, X. (2013). Combining User Preferences and User Opinions for Accurate Recommendation. *Electron. Commer. Rec. Appl.*, *12*(1), 14–23.

https://doi.org/10.1016/j.elerap.2012.05.002

Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating Collaborative Filtering Recommender Systems. ACM Trans. Inf. Syst., 22(1), 5–53. https://doi.org/10.1145/963770.963772

Nehete, S. P., & Devane, S. R. (2018). Recommendation Systems: Past, Present and Future. *2018 Eleventh International Conference on Contemporary Computing (IC3)*, 1-7. https://doi.org/10.1109/IC3.2018.8530620

Gorodetsky, V., Samoylov, V., & Serebryakov, S. (2010). Ontology–Based Context–Dependent Personalization Technology. 2010 IEEE/WIC/ACM *International Conference on Web Intelligence and Intelligent Agent Technology, 3, 278-283*. https://doi.org/10.1109/WI-IAT.2010.254

Marović, M., Mihoković, M., Mikša, M., Pribil, S., & Tus, A. (2011). Automatic movie ratings prediction using machine learning. *2011 Proceedings of the 34th International Convention MIPRO,* 1640-1645.

Vozalis, M. G., & Margaritis, K. G. (2007). Using SVD and Demographic Data for the Enhancement of Generalized Collaborative Filtering. *Inf. Sci., 177(15), 3017–3037.* https://doi.org/10.1016/j.ins.2007.02.036

Bobadilla, J., Hernando, A., Ortega, F., & Bernal, J. (2011). A framework for collaborative filtering recommender systems. *Expert Systems with Applications, 38(12), 14609-14623*. https://doi.org/10.1016/j.eswa.2011.05.021

Pazzani, M. J. (1999). A Framework for Collaborative, Content-Based and Demographic Filtering. *Artif. Intell. Rev., 13(5-6), 393–408.* https://doi.org/10.1023/A:1006544522159

Ericson, K., & Pallickara, S. (2013). On the Performance of High Dimensional Data Clustering and Classification Algorithms. *Future Gener. Comput. Syst., 29(4), 1024–1034.*

https://doi.org/10.1016/j.future.2012.05.026

Ericson, K., & Pallickara, S. (2011). On the Performance of Distributed Clustering Algorithms in File and Streaming Processing Systems. *2011 Fourth IEEE International Conference on Utility and Cloud Computing, 33-40.* https://doi.org/10.1109/UCC.2011.15

Srifi, M., Hammou, B. A., Mouline, S., & Lahcen, A. A. (2018). Collaborative Recommender Systems Based on User-Generated Reviews: A Concise Survey. *2018 International Symposium on Advanced Electrical and Communication Technologies (ISAECT)*, 1-6. https://doi.org/10.1109/ISAECT.2018.8618822

Petticrew, M., & Roberts, H. (2006). Systematic Reviews in the Social Sciences: A Practical Guide (Vol. 11). https://doi.org/10.1002/9780470754887

Bamane, P.(2016). *A Study of Amazon User Review Data using Visualization*. 39. Retrieved from https://www.cs.rit.edu/usr/local/pub/GraduateProjects/2161/psb4940/Report.pdf

Ziani, A., Azizi, N., Schwab, D., Aldwairi, M., Chekkai, N., Zenakhra, D., & Cheriguene, S. (2017). Recommender System Through Sentiment Analysis. *2nd International Conference on Automatic Control, Telecommunications and Signals*. Presentado en Annaba, Algeria. Recuperado de https://hal.archives-ouvertes.fr/hal-01683511

Harrage, F., Alsalman, A., & Alqahtani, A. (2019). *Rating Predictor: Sentiment Analysis of Product Reviews in Arabic*. 44-49. https://doi.org/10.1109/IALP.2018.8629134

Python Software Foundation. (2019). What is Python? Executive Summary. Recuperado 30 de mayo de 2019, de Python.org website: https://www.python.org/doc/essays/blurb/

Bird, S., Klein, E., & Loper, E. (2009). Natural Language Processing with Python (1st ed.). O'Reilly Media, Inc.

Project Jupyter. (2019). Project Jupyter. Recuperado 30 de mayo de 2019, de https://www.jupyter.org

Miller, G. A. (1995). WordNet: A Lexical Database for English. *Commun. ACM*, *38*(11), 39–

41. https://doi.org/10.1145/219717.219748

Cunningham, H. (2006). Information Extraction, Automatic. En K. Brown (Ed.), *Encyclopedia of Language & Linguistics (Second Edition)* (pp. 665-677). https://doi.org/10.1016/B0-08-044854-2/00960-3

Chinchor, N., & Sundheim, B. (1993). MUC-5 evaluation metrics. Recuperado 2 de junio de 2019, de https://dl.acm.org/citation.cfm?doid=1072017.1072026

Breese, J. S., Heckerman, D., & Kadie, C. (1998, July). Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence (pp. 43-52). Morgan Kaufmann Publishers Inc.

Aggarwal, C. C., Wolf, J. L., Wu, K. L., & Yu, P. S. (1999, August). Horting hatches an egg: A new graph-theoretic approach to collaborative filtering. In Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 201-212). ACM.

Li, S., Huang, C. R., Zhou, G., & Lee, S. Y. M. (2010, July). Employing personal/impersonal views in supervised and semi-supervised sentiment classification. In Proceedings of the 48th annual meeting of the association for computational linguistics (pp. 414-423). Association for Computational Linguistics.

Van Dongen, S., & Enright, A. J. (2012). Metric distances derived from cosine similarity and Pearson and Spearman correlations. arXiv preprint arXiv:1208.3145.

Sheng Zhang, Weihong Wang, Ford, J., Makedon, F., & Pearlman, J. (2005). Using singular value decomposition approximation for collaborative filtering. Seventh IEEE International Conference on E-Commerce Technology (CEC'05), 257-

264. https://doi.org/10.1109/ICECT.2005.102

Nabil, S., Elbouhdidi, J., & Yassin, M. (2018). Recommendation System Based on Data Analysis-Application on Tweets Sentiment Analysis. 2018 IEEE 5th International Congress on Information Science and Technology (CiSt), 155-160. https://doi.org/10.1109/CIST.2018.8596418

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. the Journal of machine Learning research, 12, 2825-2830.

Rehurek, R., & Sojka, P. (2010). Software framework for topic modelling with large corpora. In In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks.

Liaw, A., & Wiener, M. (2002). Classification and regression by randomForest. R news, 2(3), 18-22.

Paatero, P., & Tapper, U. (1994). Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. Environmetrics, 5(2), 111-126.

Robbins, H., & Monro, S. (1951). A stochastic approximation method. The annals of mathematical statistics, 400-407.

Dror, R., Baumer, G., Shlomov, S., & Reichart, R. (2018, July). The hitchhiker's guide to testing statistical significance in natural language processing. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (pp. 1383-1392).

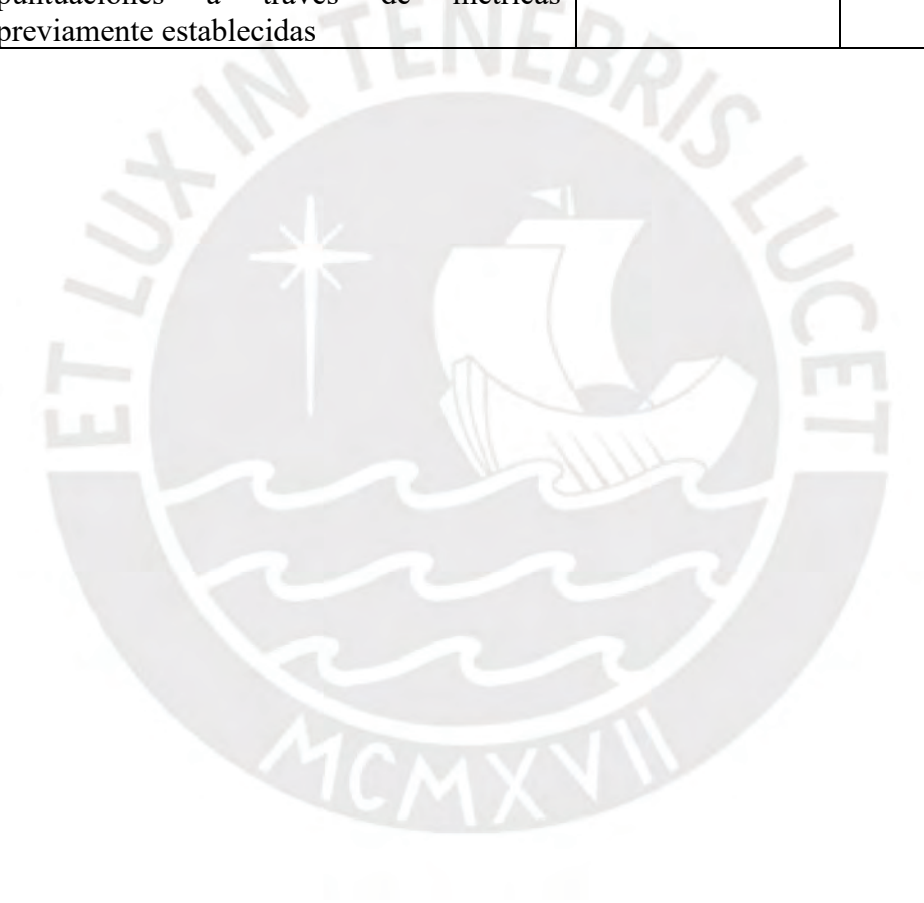
Anexos

Anexo A: Planificación de tareas

Tabla A17. Planificación de tareas (Elaboración propia).

Actividad	**Fecha de inicio**	**Fecha de fin**
Gestión del proyecto y documentación		
Investigación sobre cómo analizar el sentimiento de un texto, así como los diversos enfoques de sistemas de recomendación y cómo juntar ambos rubros de investigación	01/04	27/04
Definición del objetivo general, objetivos específicos y resultados esperados	28/04	12/05
Investigación de las herramientas, métodos y procedimientos utilizados en el presente proyecto, así como el alcance y las limitaciones presentes	13/05	02/06
Análisis de la viabilidad del proyecto	03/06	16/06
Correcciones de las observaciones recibidas sobre el proyecto	17/06	21/06
O 1. Caracterizar perfiles de usuarios con base en sus comentarios y puntuaciones sobre los productos ofrecidos en un *marketplace*.		
Exploración de los comentarios y puntuaciones de productos	16/07	22/07
Preprocesamiento de los datos (extracción, limpieza y *tokenización* de los comentarios)	23/07	29/07
O 2. Predecir productos a recomendar usando solo las puntuaciones otorgadas por los usuarios del *marketplace* como modelo base.		
Selección y entrenamiento de un modelo de aprendizaje de máquina para utilizar las puntuaciones en las recomendaciones	30/07	05/08
Creación de la relación de similitud entre usuarios para su uso en la recomendación	06/08	12/08
Implementar y validar el sistema de recomendación simple	27/08	02/09
O 3. Predecir la polaridad de los comentarios de los usuarios.		
Aplicación de un modelo algorítmico que permita determinar la polaridad de un texto	03/09	09/09
O 4. Predecir productos a recomendar usando las puntuaciones y comentarios de evaluaciones de los usuarios, comparándolo con el modelo base.		

Seleccionar y entrenar un modelo de aprendizaje de máquina para utilizar las puntuaciones y comentarios de usuarios en las recomendaciones	10/09	23/09
Ponderar las puntuaciones de los comentarios y del producto y generar la relación de usuarios por producto	24/09	30/09
Tratamiento de los datos nulos y realizar la similitud entre usuarios	01/10	07/10
Implementar y validar el sistema de recomendación usando comentarios y puntuaciones a través de métricas previamente establecidas	01/10	07/10



Anexo B: Primer informe de análisis de resultados

Resumen

Se experimentó con información de puntuaciones y comentarios de Amazon, categoría electrónicos, con el fin de implementar un algoritmo capaz de recomendar un producto dado el historial de puntuación del usuario. En el presente documento se detalla las métricas usadas y la elección de los hiper-parámetros así como los resultados obtenidos.

Datos iniciales

Inicialmente, se obtuvo el siguiente *dataframe* de comentarios (tabla B18):

Antes (7 824 482 de comentarios)

Tabla B18. Conjunto de datos base (Elaboración propia).

reviewerID	asin	reviewerName	helpful	reviewText	overall	summary	unixReviewTime	reviewTime
AKM1MP6P0OYPR	0132793040	Vicki Gibson "momo4"	[1, 1]	Corey Barker does a great job of explaining Bl...	5.0	Very thorough	1365811200	04 13, 2013
A2CX7LUOHB2NDG	0321732944	Bernie	[0, 0]	While many beginner DVDs try to teach you ever...	5.0	Adobe Photoshop CS5 Crash Course with master P...	1341100800	07 1, 2012
A2NWSAGRHCP8N5	0439886341	bowmans2007	[1, 1]	It never worked. My daughter worked to earn th...	1.0	absolutely horrible	1367193600	04 29, 2013
A2WNBOD3WNDNKT	0439886341	JAL	[1, 1]	Some of the functions did not work properly. ...	3.0	Disappointing	1374451200	07 22, 2013
A1GI0U4ZRJA8WN	0439886341	Truthfull	[4, 4]	Do not waste your money on this thing it is te...	1.0	TERRIBLE DONT WASTE YOUR MONEY	1334707200	04 18, 2012

Con la siguiente proporción de puntuaciones (ilustración B16):

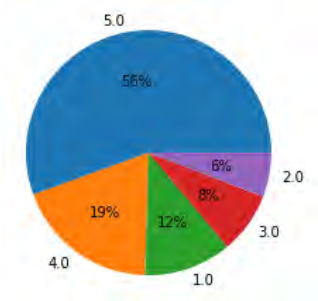


Ilustración B16. Gráfico de distribución pie de puntuaciones (Elaboración propia).

Se procedió a transformar los datos iniciales de la siguiente forma:

Con el propósito de que la información que sirva de entrada al algoritmo de predicción sea relevante, se optó en conjunto con el asesor filtrar el conjunto de datos de manera que tengamos aquellos usuarios y productos (asin) que tuvieran más de 10 comentarios, teniendo así un total de 51322 usuarios y 17156 productos (tabla B19).

Después (610 837 de comentarios)

Tabla B19. Datos después de la limpieza (Elaboración propia).

reviewerID	asin	score
A2IDCSC6NVONIZ	0972683275	5
A3BMUBUC1N77U8	0972683275	4
AYQNWE3AX4H08	0972683275	5
AQBLWW13U66XD	0972683275	5
AUKEU9CW56TT4	0972683275	5

Posteriormente, se transformó los datos en forma matriz usuario por producto (tabla B20).

Tabla B20. Matriz usuario por producto (Elaboración propia).

asin	0972683275	1400501466	1400501520	1400532620	1400532655	140053271X	1400599997	1400698987	3744295508	9573212919	...
reviewerID											
A0251761JI35FM4C8VK2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...
A02712303HM5RXRLNJUB7	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...
A0279100VZXR9A2495P4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...
A0284208PB0CNSHI1OC6	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...
A02970121VCH64N53W9F4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...
...	...	...	...	...	...	...	...	...	...	...	...
AZZTC2OYVNE2Q	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...
AZZV2NAQL8KHQ	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...
AZZVLOF3WKLFW	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...
AZZX23UGJGKTT	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...
AZZYW4YOE1B6E	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...

Como se puede apreciar, es natural que se encuentren valores nulos dado que es prácticamente imposible que un usuario haya comprado y dejado su puntuación a 17 mil productos. Para tener una visión más clara de esto, se presenta un gráfico del top 20 usuarios con mayor cantidad de puntuaciones (ilustración B17) realizado con base en los 7,824,482 de comentarios, es decir, antes de realizar filtros y transformaciones:

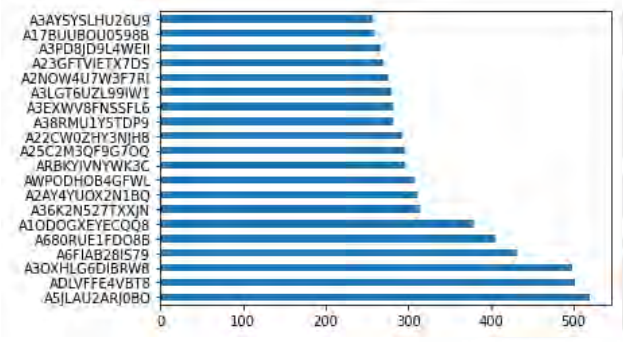


Ilustración B17 Top comentarios por usuario (Elaboración propia).

Se observa que la mayor cantidad de comentarios que un usuario ha dejado es 520.

Es por ello que se aplicará la técnica de matriz de factorización no negativa para entrenar un modelo que, mediante el uso del gradiente descendente estocástico, sea capaz de predecir la puntuación asociada a un producto dado un usuario y completar así la matriz previamente mostrada.

Procedimiento

Matriz de factorización no negativa

Como su nombre lo indica, en ella no habrá números negativos, una característica intrínseca a una puntuación dejada a un producto en un *Marketplace*. El concepto se basa en tener una matriz original de usuario por producto a partir de la cual se van a generar 2 matrices más pequeñas, usuario por característica y característica por producto, representación gráfica en la ilustración B18.

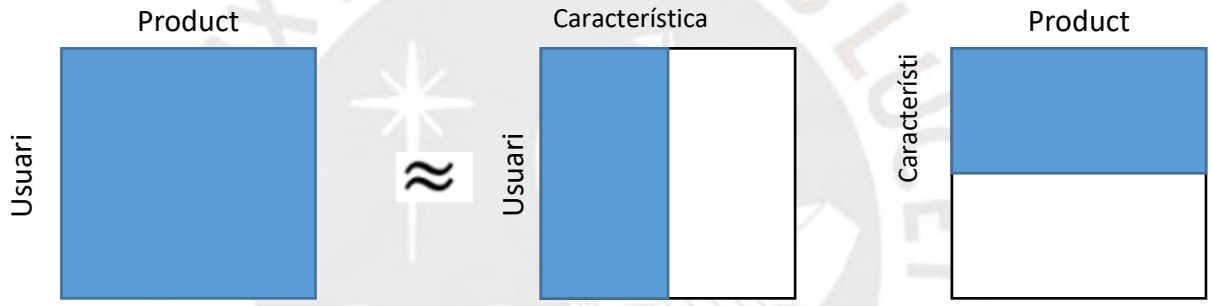


Ilustración B18. Matriz de factorización no negativa (Elaboración propia).

¿Cuál es el objetivo? Que, al usar estas 2 matrices generadas, se pueda reconstruir la matriz original.

Para ello, es importante entender el concepto de característica o *feature*. Este concepto se representa por un valor entero que denominaremos k, el cual es el número de características latentes que presenta la información. K tiene algunas restricciones, por ejemplo, tiene que ser menor al mínimo entre el número de usuarios y el número de productos, ya que de no serlo puede ocasionar el problema de tener información redundante.

Ejemplo:

Supongamos que tenemos un conjunto de datos simple de la siguiente forma (tabla B21)

Tabla B21. Ejemplo matriz de factorización no negativa (Elaboración propia).

x	y	z	
2	1	0	3
5	1	2	4

Donde X, Y y Z pueden indicar características latentes como el tipo o la categoría a la que pertenece nuestros productos del conjunto de datos. Si infringimos la restricción mencionada previamente, observamos que si k = 4, la cuarta columna es información redundante (x+y-z).

Como se puede observar, la elección del valor de k es fundamental como hiper-parámetro, es por ello que se realizó un experimento para escoger el valor ideal con respecto a los datos, lo cual analizaremos en la siguiente sección.

Validación cruzada

Es una técnica estadística para la validación de modelos que permite observar cómo este se comporta al momento de generalizar los resultados hacia un conjunto de datos diferente. En este caso hemos tomado 5 como la cantidad de divisiones del conjunto de datos, como se muestra en la siguiente tabla B22:

Tabla B22. Validación cruzada (Elaboración propia).

Conjunto de datos original							
Prueba	Entrenamiento	Entrenamiento	Entrenamiento	Entrenamiento	->	err1	
					->	err2	
					->	err3	Error medio
					->	err4	
					->	err5	

La idea es que, teniendo un conjunto de datos, se hagan 5 divisiones al mismo, dejando siempre una para el conjunto de datos de prueba (naranja) y el resto para el entrenamiento (azul). Este proceso se repetirá tantas veces como divisiones se haga, teniendo un error del conjunto de prueba en cada una de estas y al final, obteniendo un error ponderado del proceso total.

En este caso, se ha tomado como métrica el RMSE, cantidad de divisiones = 5 y se ha variado K para observar cómo esto afecta al RMSE, expresado en la ilustración B19.

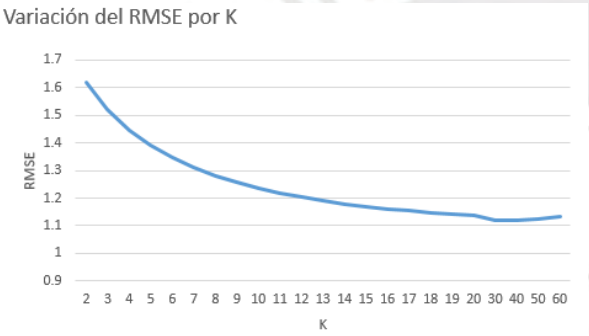


Ilustración B19. Evolución de RMSE con respecto a k. (Elaboración propia).

Como se puede observar, el error disminuye significativamente hasta llegar a 10, después del cual disminuye levemente. Es por ello que, gracias a la técnica estadística de validación cruzada, se ha escogido como hiper-parámetro k = 10 dado que un alto valor de k puede resultar en un *overfitting* (datos no generalizables, muy sesgados al conjunto de entrenamiento) y un valor muy bajo puede resultar en *underfitting (*no optimizado).

Resultados

Tabla B23. Resultados de recomendación usuario por producto (Elaboración propia).

asin	0972683275	1400501466	1400501520	1400532620	1400532655	140053271X	1400599997	1400698987	3744295508	9573212919	...
reviewerID											
A0251761JI35FM4C8VK2	4.706650	4.508446	4.792036	4.324033	4.051137	4.179322	4.270046	4.761762	5.038861	4.433788	...
A02712303HM5RXRLNJUB7	4.741503	4.542646	4.809396	4.332911	4.050438	4.192832	4.301733	4.797574	5.048815	4.460074	...
A0279100VZXR9A2495P4	4.495445	4.293722	4.537172	4.098185	3.830920	3.940022	4.049890	4.527058	4.796654	4.233025	...
A0284208PB0CNSHI1OC6	4.523478	4.336636	4.579095	4.146067	3.838661	3.968005	4.089254	4.531731	4.864285	4.256095	...
A02970121VCH64N53W9F4	4.573627	4.378821	4.604627	4.180112	3.873393	4.061425	4.102780	4.594836	4.877304	4.266451	...

Se puede observar la matriz llena (tabla B23) con los valores predichos que un usuario dará a un determinado producto. A continuación, las métricas usadas para la evaluación del modelo fueron:

$$RMSE = \sqrt{\frac{1}{n} + \sum_{i=1}^{n} (Fi - Yi)^2}$$

RMSE = 0.913561484915

$$MAE = \frac{1}{n} + \sum_{i=1}^{n} |Fi - Yi|$$

MAE = 0.6676463255109276

A continuación, se presenta un ejemplo de la matriz original (tabla B24) para el usuario con id = ADLVFFE4VBT8 y la matriz predicha (tabla B25)

Original:

Tabla B24. Resultado recomendación, ejemplo original (Elaboración propia).

asin	B00005114Z	B000067RVL	B000067SGI	B00006B8K2	B00009V2W9	B0001IXUDK	B0001LTT64	B000261N6M	B0002CPBUK	B0002KKIUA	.
reviewerID											
ADLVFFE4VBT8	5	5	5	4	5	1	5	5	5	5	.

Predicha:

Tabla B25. Resultado recomendación, ejemplo predicho (Elaboración propia).

asin	B00005114Z	B000067RVL	B000067SGI	B00006B8K2	B00009V2W9	B0001IXUDK	B0001LTT64	B000261N6M	B0002CPBUK	B0002KKIUA	.
reviewerID											
ADLVFFE4VBT8	4.692941	4.523712	4.752322	4.661409	4.503017	3.014367	4.329027	3.245001	4.538254	4.290204	.

Anexo C: Segundo informe de análisis de resultados

Resumen

Se experimentó con información de puntuaciones y comentarios de Amazon, categoría electrónicos, con el fin de implementar un algoritmo capaz de recomendar un producto dado el historial de puntuación del usuario y la determinación de la polaridad de los comentarios realizados. En el presente documento se detallará las métricas usadas y se realizara un análisis estadístico para determinar qué algoritmo es mejor con un nivel de significancia del 5%.

Procedimiento

Una vez se obtuvo la matriz explicada en el primer informe de análisis de resultados, se procedió a incluir la valoración la extracción de la polaridad de los comentarios dejados por los usuarios sobre los productos con el fin de reducir el error entre el valor predicho y el valor real de la puntuación dada la polaridad resultante del comentario. Para ello, se tiene la siguiente fórmula:

$$newx = x + a * abs(round(x) - x)/2$$

Donde:

- newx: Es el nuevo valor predicho de la puntuación
- x = El antiguo valor predicho de la puntuación
- a = Será 1 o -1 dependiendo de si es positivo o negativo, valor resultante de la polaridad.

Con ello tenemos que, si el algoritmo predijo una puntuación de 3.4 para un usuario y se tiene una polaridad negativa para el comentario de dicho usuario, se reducirá el valor de

3.4 a 3.2 ya que “a” tomaría el valor de -1, lo cual indica una disminución del valor predicho cuya cantidad será el valor absoluto de (3.4 – 3) / 2; es decir, 0.2.

Una vez obtenida la nueva matriz actualizada, se procedió a realizar el módulo de recomendación de productos. Para ello, se trabajó con 10 mil usuarios y se obtuvo lo siguiente:

1. Se calculó la similitud entre los usuarios mediante la similitud de Pearson con el fin de determinar cuáles son los usuarios más similares al usuario al que se quiere recomendar con base en su historial de compras.

2. Con la lista obtenida en 1, se obtuvieron los productos que estos usuarios habían comprado y otorgado una puntuación mayor igual a 4.

3. Luego, con la lista obtenida en 2, se filtró aquellos productos que el usuario no hubiera comprado antes; es decir, filtrado de manera que todos los productos fueran nuevos para él.

4. Finalmente, se obtuvo la puntuación de la matriz actualizada de la lista de productos obtenida en 3 y se ordenó descendentemente, obteniendo así el top n productos para el usuario objetivo.

Finalmente, una vez actualizada la matriz y el módulo de recomendación, se procedió a realizar el análisis estadístico entre el algoritmo base vs el algoritmo propuesto para determinar si el último era significativamente mejor que el primero.

Para ello, se utilizó el artículo académico The Hitchhiker's Guide to Testing Statistical Significance in Natural Language Processing publicado en el ACL 2018, el cual anima a todos a brindar datos estadísticos sobre los estudios realizados con el fin de poder hacer una aseveración con base estadística sobre si un algoritmo es mejor que otro, ya que no basta solo con decir que la métrica utilizada mayor o menor. Así mismo, explica e implementa qué tipo de prueba estadística usar dado la información manejada en los estudios y/o experimentaciones. Es así que, dado las directrices planteadas en el trabajo mencionado, se realizó la prueba de Shapiro para determinar la normalidad en la distribución de los datos y posteriormente, la prueba de t-test para datos pareados (puesto que se comparó 2 algoritmos sobre el mismo conjunto de datos) para determinar si el algoritmo con la extracción de la polaridad sobre los comentarios era mejor que el base.

Resultados

A continuación, las métricas para evaluar el modelo:

$$RMSE = \sqrt{\frac{1}{n} + \sum_{x=1}^{n} (newx - x)^2}$$

Para 10 muestras aleatorias de los datos, se obtuvo los siguientes valores para RMSE (Tabla C26):

Tabla C26. Comparación RMSE algoritmo base vs propuesto. (Elaboración propia)

Algoritmo base (x)	Algoritmo propuesto (y)
0.9021673162410346	0.8571659946693936
0.9084158671375143	0.8638309401728261
0.908367203420963	0.8639422065913938
0.9132030967819319	0.8682152407262937

0.9213746662572685	0.8760991721293383
0.91830605354168	0.87290556256022
0.9202644504923404	0.8747059768962058
0.918461683546362	0.8731402298607015
0.9182307125362127	0.8728698922170688
0.9187888083444719	0.8732866518559279

Shapiro test:

Con un nivel de significancia del 5%:

H0 = los datos siguen una distribución normal

H1 los datos no siguen una distribución normal

p-value = 1.0, no se rechazó H0, los datos siguen una distribución normal

T-test:

Con un nivel de significancia del 5%:

H0 = la media de X es mayor igual a la media de Y

H1 = la media de Y es menor a la media de X

p-value: 1.93352462824e-20, se rechaza H0, la media de Y es menor a la media de X.

Conclusión:

Se determinó estadísticamente que, con un nivel de significancia del 5%, el algoritmo propuesto tiene un RMSE menor al algoritmo base.