# Modelo de Recomendación de Nuevos Productos a Clientes Actuales

New Product Recommendation Model

Products to Existing Customers

PABLO ISAZA HIGUERA

PROYECTO DE GRADO

Asesor, docente

LINA MARIA SEPULVEDA CANO

UNIVERSIDAD EAFIT

ESCUELA DE CIENCIAS APLICADAS E INGENIERÍA

MAESTRÍA EN CIENCIAS DE LOS DATOS Y LA ANALÍTICA

MEDELLÍN

2024

# Modelo de Recomendación de Nuevos Productos a Clientes Actuales

**Estudiante:** Pablo Isaza Higuera

**Director:** Lina María Sepúlveda Cano

Área de Gestión de la información y riesgos

lmsepulvec@eafit.edu.co

Categoría MINCIENCIAS: ASOCIADA

Categoría EAFIT: TITULAR

**Codirector:** Mateo León

Presidente compañía DISAN

Magister en Administración (MBA)

mateo.leon@disan.com.co

**Palabras clave:** *Machine Learning;* Sistemas de Recomendación (RS); Arranque en frio, RS Filtro colaborativo, RS Basado en contenido, RS Hibrido

# Tabla de Contenido

# Modelo de recomendación de nuevos productos a clientes actuales

## 1. Resumen

La compañía DISAN se enfrenta al desafío de adaptar sus portafolios de productos de una región a otra, un proceso que actualmente toma alrededor de 5 meses. Esto ha generado la necesidad de desarrollar un sistema eficiente de recomendación de nuevos productos para sus clientes actuales.

Este trabajo de grado busca abordar la problemática mencionada mediante la combinación de diferentes estrategias y técnicas de sistemas de recomendación. Se espera que este sistema permita sugerir productos nuevos incluso en situaciones donde no se disponga de información previa sobre las preferencias de los usuarios.

Aunque existen sistemas de recomendación en el mercado, la adaptación específica de los productos a las preferencias de los usuarios en nuevas regiones presenta un desafío único para DISAN. Por lo tanto, existe un vacío en cuanto a la disponibilidad de un sistema que pueda abordar eficazmente esta necesidad específica de la empresa.

En este trabajo de grado se desarrollará un sistema de recomendación que utiliza datos sobre el comportamiento de los usuarios, las características de los productos existentes y los nuevos productos a introducir. Este sistema puede predecir las preferencias de los usuarios y recomendar los mejores productos nuevos a cada usuario, teniendo en cuenta sus características individuales y sus preferencias históricas. Se espera que este sistema contribuya significativamente a mejorar la experiencia del usuario y fortalecer la fidelidad de los clientes hacia DISAN.

## 2. Descripción del Proyecto

### 2.1 Planteamiento del Problema

El problema central al que se enfrenta la compañía DISAN es la necesidad de incrementar la relevancia y eficacia de las recomendaciones de productos para sus clientes actuales. Aunque la compañía ya cuenta con un sistema de recomendación en funcionamiento, este enfrenta limitaciones significativas, especialmente en lo que respecta a la recomendación

de nuevos productos y el inicio en frío de estos elementos. Adicional, el tiempo de implementación de estos nuevos productos es considerable (5 meses). Esto se traduce en oportunidades perdidas para aumentar las ventas y mejorar la satisfacción del cliente en el menor tiempo posible.

Actualmente, la metodología para introducir un nuevo portafolio se basa en llevar a cabo un análisis del caso de negocio para evaluar el valor comercial de la iniciativa del nuevo producto, garantizando que las actividades de desarrollo se ajusten a la estrategia de la empresa y tengan como objetivo mejorar la rentabilidad global; calcular los parámetros financieros del nuevo artículo de venta teniendo en cuenta los cambios en las ventas de la cartera de productos; desarrollar un concepto que compare los flujos de caja netos previstos para las partidas de ventas afectadas por la introducción del nuevo artículo en dos escenarios: con y sin el nuevo artículo. Esto ayuda a evaluar el impacto en la cartera global, aplicando un concepto para la planificación y el seguimiento de las ventas a nivel de cartera de productos, centrándose en los volúmenes de ventas previstos, los precios, la fijación de objetivos y el análisis del rendimiento real de las ventas a diferentes niveles de la cartera de productos [20].

## 2.2 Justificación

El proyecto propuesto se centra en eliminar el dolor corporativo asociado con la distribución eficiente de una nueva cartera de productos. Este dolor se manifiesta en una falta de enfoque claro en dirigirse a los clientes con más probabilidades de comprar nuevos productos, lo que resulta en un uso ineficiente de los recursos y un mayor tiempo de conversión de clientes potenciales. Sin un modelo estructurado para identificar y priorizar a los clientes más adecuados para los nuevos productos, la empresa enfrenta varios desafíos. Estos incluyen extender los esfuerzos de marketing a clientes que tal vez no estén interesados o no necesiten nuevos productos, lo que genera costos operativos más altos y un ciclo de ventas más largo. Al implementar este modelo de referencia, la empresa puede enfocar efectivamente a sus vendedores en los clientes más prometedores, aquellos que tienen más probabilidades de comprar nuevos productos. Esto optimiza el uso de recursos y reduce el trabajo asociado con la entrega de productos a clientes que no los reciben. Al orientar con mayor precisión los esfuerzos comerciales, la empresa puede acelerar la conversión de clientes potenciales en clientes reales, lo que conduce a un crecimiento significativo de las ventas.

## 2.3 Pregunta de investigación

¿Cómo puede un sistema de recomendación impulsar el lanzamiento exitoso de nuevos productos en una empresa?

## 2.4 Objetivos

### 2.4.1 Objetivo General

- Desarrollar un modelo de recomendación de nuevos productos basado en las interacciones del cliente y el producto, con el fin de focalizar el recurso de la compañía.

### 2.4.2 Objetivos Específicos

- Seleccionar un portafolio que sea representativo para la compañía (ventas, volumen, diversidad de clientes, etc) para el entrenamiento del modelo.
- Desarrollar un modelo de recomendación que combine la matriz de interacción cliente-producto actual con la matriz de interacción de productos nuevos para la recomendación de portafolio.
- Evaluar el desempeño del modelo utilizando métricas de calidad en conjuntos de datos de prueba, con el objetivo de medir su eficacia y precisión en entornos del mundo real.

## 2.5 Estado del Arte y Marco Teórico

Un sistema de recomendación es un conjunto de técnicas y herramientas de software que proveen sugerencias de ítems o elementos a los usuarios [1], con el objetivo de mejorar la experiencia del usuario al facilitar la búsqueda de productos relevantes, promover el descubrimiento de nuevos *ítems* y fomentar el compromiso y la satisfacción del cliente. Estos sistemas ayudan tanto a los usuarios a encontrar productos de su interés como a las empresas a incrementar la fidelidad de sus clientes y a optimizar sus estrategias de ventas.

El problema del arranque en frío (*cold-start*) en los sistemas de recomendación se refiere a la dificultad de hacer recomendaciones precisas para usuarios o artículos nuevos sin datos de interacción previos. Los métodos tradicionales de filtrado colaborativo no son efectivos en estas situaciones, ya que se basan en interacciones anteriores. En este trabajo de grado, se propone una metodología que aborda este problema aprovechando patrones frecuentes discriminantes entre grupos de usuarios. Al agrupar a los usuarios antiguos y encontrar patrones frecuentes específicos para cada grupo, el sistema puede predecir eficazmente el comportamiento de compra de los nuevos usuarios. El enfoque presentado en este trabajo de grado combina la agrupación, la minería de conjuntos de elementos discriminantes y el contexto usuario/producto para abordar eficazmente los desafíos del arranque en frío y la escasez de datos en la recomendación [2].

Existen varias generaciones de los sistemas de recomendación [3]:

**Sistemas de recomendación de primera generación:**

Los siguientes son los diferentes métodos de toma de decisiones utilizados en el sistema de recomendación de primera generación: filtrado basado en el contenido, filtrado colaborativo y métodos híbridos como lo menciona en [4]. Se agrupan en primera generación debido a que son las primeras aproximaciones para desarrollar el desafío de la creación de recomendación y porque son sistemas muy limitados en cuanto a la personalización y la adaptación del contexto. La similitud coseno, la distancia euclidiana y el coeficiente de correlación de Pearson (*PCC* por sus siglas en inglés) son las diferentes medidas de similitud utilizadas por estos métodos [5,6]. El *PCC* tiene el gran inconveniente de sobrestimar la preferencia del usuario cuando éste ha valorado sólo unos pocos ítems de forma idéntica [7].

**Sistemas de recomendación de segunda generación:**

En la segunda generación de un sistema de recomendación, se hizo énfasis en el conocimiento sobre el interés del usuario y el contexto de la tarea que hacía la recomendación como se menciona en [4]. Algunos sistemas de recomendación que entran en esta generación son: personalización basada en ubicación, personalización basa en contexto de compra. Esta generación se centró en mantener el perfil del usuario y recomendó los elementos basándose en la lista de usuarios.

**Sistemas de recomendación de tercera generación:**

Este tipo de sistema tiene en cuenta el estado emocional, monetario, el comportamiento psicológico y otros factores afines asociados al proceso de toma de decisiones. Algunos sistemas de recomendación que entran en esta generación son: Técnicas de lenguaje natural y análisis de sentimientos, con el fin de detectar el estado emocional del usuario a partir de sus interacciones. En este sistema, el modelo ontológico se considera el modelo general para representar todos los componentes del conocimiento [8].

## Sistemas más relevantes:

### Filtrado colaborativo

Este tipo de sistema de recomendación pertenecen a la primera generación y este hace recomendaciones al usuario activo utilizando información sobre otros usuarios y su relación con el artículo [9]. La idea principal detrás del funcionamiento del filtrado colaborativo es la presunción de que los usuarios que tenían preferencias similares en el pasado es de suponer que tendrán preferencias similares en el futuro [10,11].

El filtrado colaborativo ofrece la ventaja de no requerir el conocimiento del dominio del usuario para hacer predicciones, al mismo tiempo que presenta una escalabilidad de los elementos que no requiere intervención humana. Sin embargo, este enfoque conlleva desventajas, ya que enfrenta problemas de escasez, dado que la mayoría de los artículos en sitios web de comercio electrónico o tiendas no están valorados y, por lo tanto, no pueden ser recomendados hasta que obtengan valoraciones de otros usuarios o tengan similitudes con otros productos. Los algoritmos de filtrado colaborativo se dividen principalmente en sistemas basados en modelos, sistemas basados en la memoria y sistemas híbridos [6] (Todos estos entrar en la clasificación de sistemas de primera generación).

El sistema de filtrado colaborativo basado en memoria o heurístico utiliza los datos del elemento o usuario que permanecen disponibles durante el tiempo de ejecución del algoritmo en la memoria del sistema proporcionando una recomendación con alta precisión [8].

Los métodos basados en modelos explotan la selección grupal de valoraciones para predecir las valoraciones [23]. Realizan cálculos fuera de línea para el entrenamiento. Los sistemas basados en modelos tienden a ser más precisos en comparación con los basados en memoria, especialmente cuando se aplican técnicas avanzadas como el aprendizaje automático o la factorización de matrices. Estos modelos pueden generalizar mejor a nuevos datos (por ejemplo, nuevos usuarios o productos) y tienen un mejor rendimiento en escenarios de esparcidad de datos, ya que no dependen directamente de las interacciones explícitas entre usuarios y productos [12].

Los modelos basados en memoria enfrentan problemas de escalabilidad principalmente debido a la necesidad de calcular similitudes entre todos los usuarios o productos. Por otro lado, los modelos basados en modelos, como los que utilizan factorización de matrices o redes neuronales, pueden escalar mejor en cuanto a la cantidad de datos, pero enfrentan problemas en el entrenamiento debido a la alta complejidad computacional.

El filtrado colaborativo híbrido mejora la eficacia de la recomendación al resolver el problema de escalabilidad al que se enfrentan los métodos anteriores [13].

En la siguiente imagen, se muestra la idea central que está detrás de un sistema de filtrado colaborativo basado en memoria usando un ranking de elementos recomendados para entregar una lista de ítems recomendados:

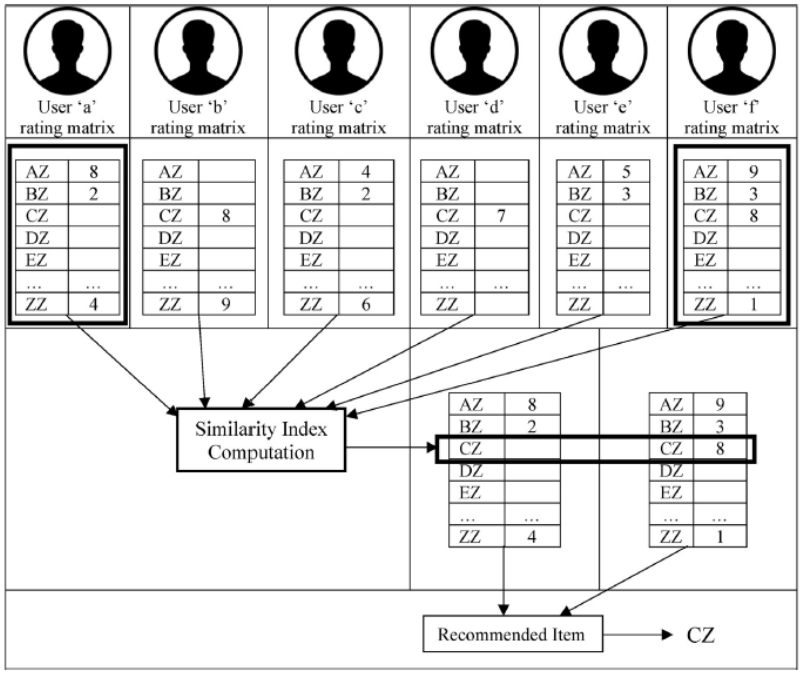



*Ilustración 1 Filtrado Colaborativo [3]*

La imagen muestra en la parte superior usuarios con una matriz de calificación de ítems, busca usuarios que tengan ítems y calificaciones similares para después buscar un ítem que no ha consumido el usuario y poder recomendarle el ítem (como se muestra en la parte inferior derecha de la imagen).

**Basado en contenido:**

Estos sistemas se basan en dos conceptos generales:

El primer concepto propone que los sistemas basados en el contenido examinan los atributos de los artículos que han sido valorados por el usuario en el pasado y construyen una lista de los intereses del usuario [14].

El segundo por su lado dice que el "portafolio de usuario", que comprende la información sobre los intereses del usuario en un sistema de recomendación, es esencial para analizar

el comportamiento pasado del usuario [6,15]. El portafolio de usuario consiste en los elementos por los cuales el usuario ha mostrado interés previamente, lo que proporciona una visión detallada de sus elecciones y actividades anteriores en el sistema. Esto permite personalizar y mejorar las recomendaciones al considerar los intereses específicos del usuario, ofreciendo así recomendaciones más relevantes y satisfactorias.

Este ofrece la ventaja de poder recomendar a los usuarios artículos nuevos e impopulares, sin depender de datos de otros usuarios [5,16]. Sin embargo, este enfoque presenta desventajas, ya que los artículos están condicionados por sus descripciones o características iniciales, lo que requiere especificar explícitamente todas estas descripciones [5]

A continuación, se muestra la idea central que está detrás de un sistema de filtrado basado en contenido usando un ranking de elementos recomendados para entregar una lista de ítems recomendados:

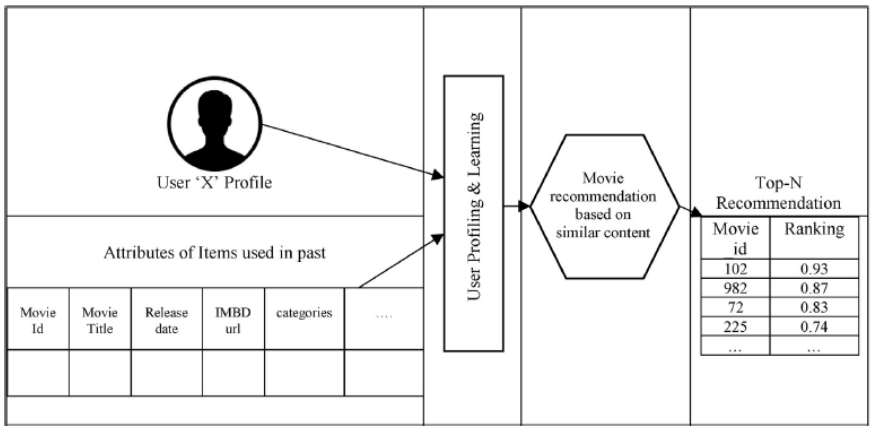



*Ilustración 2 Basado en Contenido [3]*

En general, los sistemas de recomendación basados en contenido y de filtrado colaborativo son excelentes cuando se tiene la data completa, pero tienen un grave desempeño cuando se presenta el arranque en frio, es decir, cuando tienes un cliente nuevo que no tiene interacción con ningún producto o un producto nuevo que no tiene interacción con ningún cliente.

**Híbridos:**

Un sistema híbrido combina diferentes modelos de personalización, como el basado en contenido y el filtro colaborativo, para superar los inconvenientes de estos sistemas individuales. Puede utilizar métodos como la factorización matricial y el filtrado colaborativo para generar recomendaciones personalizadas. Los sistemas híbridos son altamente eficaces al combinar los beneficios de diferentes enfoques de recomendación y proporcionan una plataforma para optimizar el modelo de recomendación [24]. Sin embargo, también tienen algunas desventajas, como un mayor costo de implementación, alta complejidad en términos de tiempo y espacio, y la necesidad de recopilar información explícita que puede plantear problemas de privacidad. En resumen, los sistemas híbridos son una solución efectiva pero costosa para la personalización de recomendaciones [3].

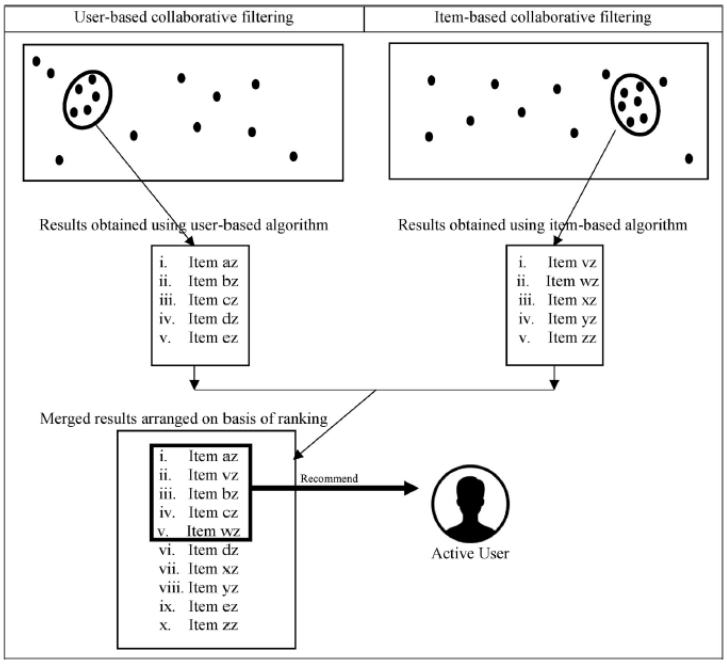



*Ilustración 3 Sistema Hibrido [3]*

Resumiendo, se muestra una taxonomía de los diferentes sistemas de recomendación:

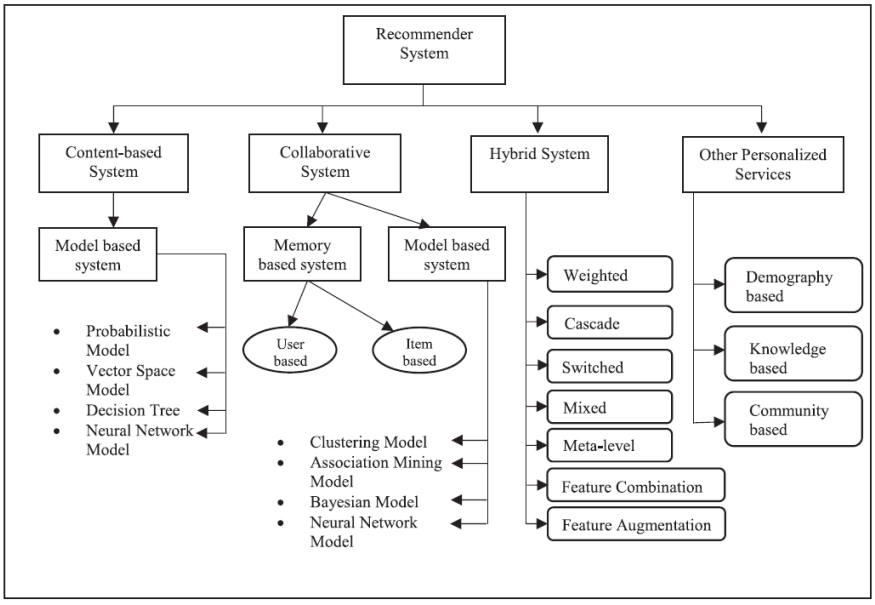



*Ilustración 4 Resumen diferentes sistemas de recomendación [3]*

Existen otros enfoques para abordar la creación de un sistema de recomendación con problema de arranque en frio, a continuación, se explicarán más a detalle estos enfoques:

**Modelos de Factorización de Matrices:**

Los modelos de factorización matricial y mapeo de características factorizan la matriz usuario-producto en dos representaciones latentes. las representaciones latentes son una forma de extraer información oculta de los datos y de comprender mejor las preferencias de los usuarios y las características de los productos. Al factorizar la matriz usuario-producto, podemos obtener una representación más compacta y significativa de los datos, lo que nos permite hacer recomendaciones más precisas y personalizadas. A partir de ahí, se puede aprender una representación latente utilizando la información implícita y utilizarla para predecir la matriz usuario-producto de los usuarios en frío. [2]

La idea principal detrás de los modelos de factorización de matrices es descomponer una matriz grande y dispersa de interacciones usuario-item (como calificaciones de películas por usuarios) en el producto de dos matrices más pequeñas y densas de factores latentes. Estos factores latentes representan características ocultas tanto de los usuarios como de los ítems. Este proceso se realiza mediante el siguiente paso a paso:

1. Obtener la matriz de interacción original (R, matriz de calificación de usuario ítem)
2. Factorización de esa matriz ($R = U_u V_i^T$, $U_u$ matriz mxk donde m es número de usuarios y k número factores latentes, $V_i$ es una matriz nxk donde n es el número de ítems y k número factores latentes. Estos factores latentes es un hiperparámetros que se debe de definir por medio de un proceso de ajustes de hiperparámetros)
3. Optimización (Esto se hace minimizando una función de pérdida, típicamente la suma de los errores cuadráticos entre las calificaciones reales y las predichas, a menudo con regularización para evitar el sobreajuste).

$$\min_{U,V} \sum_{(u,i)\in K} (R_{u,i} - U_u \cdot V_i^T)^2 + \lambda(\|U\|^2 + \|V\|^2)$$

   La función que se está minimizando es una función de pérdida, específicamente la suma de los errores cuadráticos. Esta función mide la diferencia entre las calificaciones reales de los usuarios (Ru,i) y las calificaciones predichas por el modelo (U_u * V_i^T). El proceso de encontrar el mínimo de esta función se conoce como optimización. En este caso, se trata de un problema de optimización convexa, ya que la función de pérdida es convexa.
4. Predicción ($\widehat{R_{u,i}} = U_u \cdot V_i^T$; $U_u$ es el vector de factores latentes del usuario u y $V_i$ es el vector de factores latentes del item i)

**Clustering y Minería de Reglas de Asociación (ARM):**

*Clustering* y Minería de Reglas de Asociación (ARM) son técnicas de minería de datos utilizadas para analizar y extraer información útil de grandes conjuntos de datos. Ambas técnicas pueden ser utilizadas para mejorar los sistemas de recomendación, especialmente en la resolución de problemas como el *cold-start.*

*Clustering:*

*Clustering* es un método de aprendizaje no supervisado que agrupa un conjunto de objetos de tal manera que los objetos en el mismo grupo (o clúster) son más similares entre sí que

con los objetos de otros grupos [17,19]. Es útil para descubrir estructuras subyacentes en los datos. Para el *clustering*, es necesario una suma de pasos que se explica a continuación:

1. Definir la similitud: Decidir una función de distancia (como Euclidiana, Manhattan, Coseno) para medir la similitud entre los objetos.
2. Selección del Método de *Clustering*: como el *k-means* o *Hierarchical Clustering*
3. Asignación de Objetos a Clústeres, se ejecuta el algoritmo para agrupar los objetos
4. Evaluación de resultados: Validar los clústeres usando índices como la Silhouette, la suma de cuadrados dentro del clúster (inertia), o visualizaciones como gráficos de dispersión.

Minería de Reglas de Asociación (ARM):

Minería de Reglas de Asociación (ARM) es una técnica para descubrir relaciones interesantes, útiles y no triviales entre los elementos de un conjunto de datos. Es comúnmente utilizada en el análisis de cestas de mercado [22]. Como en el clustering, aquí también es necesario realizar un proceso paso a paso el cual se explica a continuación:

1. Identificación de *Itemsets* Frecuentes: Lo que se busca es Encontrar todos los conjuntos de *items (itemsets*) que tienen una frecuencia de ocurrencia mayor que un umbral mínimo de soporte. El umbral de soporte se define como un porcentaje del total de transacciones en una base de datos. Por ejemplo, si tenemos una base de datos de 1000 transacciones y establecemos un umbral mínimo de soporte del 5%, entonces un itemset debe aparecer al menos en 50 transacciones (5% de 1000) para ser considerado frecuente.
2. Generación de Reglas de Asociación: A partir de los *itemsets* frecuentes, generar reglas de asociación que cumplen con un umbral mínimo de confianza. El umbral mínimo de confianza es un valor porcentual que indica la probabilidad mínima con la que se espera que el consecuente de una regla ocurra dado que el antecedente ya ha ocurrido. En otras palabras, es una medida de la certeza con la que podemos afirmar que si se cumple el antecedente, es muy probable que también se cumpla el consecuente.
3. Evaluación de Reglas: Evaluar la utilidad de las reglas usando métricas como soporte, confianza y lift.

Combinación de *Clustering* y ARM:

La combinación de *clustering* y ARM puede ser muy poderosa en sistemas de recomendación, particularmente para abordar problemas como el cold-start [22].

Para realizarlo es necesario realizar un procedimiento que se explicara a continuación:

1. *Clustering* de Usuarios y *Items:* Se busca agrupar usuarios e ítems en clústeres basados en similitudes en sus características o comportamientos (calificaciones de productos, perfiles de usuario).
2. Minería de Reglas de Asociación en Clústeres: Dentro de cada clúster, aplicar ARM para descubrir patrones y relaciones frecuentes que son específicos de ese grupo. Con el fin de revelar qué combinaciones de items son populares entre usuarios con características similares.

3. Recomendación Basada en Patrones Discriminantes: Utilizar los patrones discriminantes (patrones frecuentes específicos de un clúster) para hacer recomendaciones personalizadas. Para un nuevo usuario, determinar a qué clúster es más similar y usar los patrones de ese clúster para realizar recomendaciones. Para calcular los patrones discriminantes, primero se agrupan a los usuarios en distintos clústeres. Luego, se identifican los conjuntos de productos (patrones) que ocurren con mayor frecuencia en cada clúster. Finalmente, se comparan estos patrones entre los diferentes clústeres para encontrar aquellos que son únicos o mucho más frecuentes en un clúster específico. Estos patrones únicos son los discriminantes, ya que nos permiten distinguir un clúster de otro y entender las preferencias particulares de cada grupo de usuarios.

Para que quede más claro, se explicara un ejemplo práctico de cómo funciona estos modelos para un sistema de recomendación:

Se tiene un sistema de recomendación para una empresa que vende insumos para diferentes industrias y se quiere mejorar la recomendación para un nuevo usuario. El primer paso es realizar el *clustering*, para esto, se utiliza un algoritmo de *k-means* para agrupar los usuarios existentes basados en sus historiales de compra y el resultado nos da 3 *clusters* (1. Usuarios que compran pinturas, 2. Usuarios que compran fragancias y 3. Usuarios que compran alimentos). Posteriormente se debe de aplicar el ARM en los *clusters* que se encontraron para descubrir que reglas de asociación especificas tienen y como resultado obtenemos que para el clúster 1 una regla frecuente es que {compra temple} ->{compra Esmalte graso}; para el clúster 2 una regla frecuente es que {compra frutal}->{compra vegetal}; para el clúster 3 una regla frecuente es que {compra benzoato de sodio} -> {compra sorbato de potasio}. Para realizar ya la recomendación, es necesario primero evaluar el comportamiento de ese nuevo cliente (primera compra o primer artículo que ha visto) y se le asigna provisionalmente a un clúster, supongamos que compro temple, la recomendación seria esmalte graso.

## Parámetros para tener en cuenta en los sistemas de recomendación:

1. Medidas de similitud que se emplean en los sistemas de recomendación

   Las medidas de similitud son fundamentales en los sistemas de recomendación, ya que nos permiten cuantificar el grado de semejanza entre dos elementos (usuarios o ítems) y, así, realizar predicciones sobre las preferencias de un usuario. Se utilizan en el filtrado colaborativo para encontrar usuarios similares a un usuario objetivo y recomendarle los ítems que han gustado a esos usuarios similares. En filtrado basado en contenido para encontrar ítems similares a un ítem que le gusta al usuario y recomendarle otros ítems similares. El rendimiento de la mayoría de los sistemas de recomendación depende de estas medidas de similitud. Existen dos métodos populares, similitud por distancia y similitud basada en correlación. Las medidas de similitud varían dependiendo del tipo de datos y del contexto, pero en general cuantifican la cercanía entre dos elementos en un espacio vectorial. Un valor de similitud cercano a 1 indica una alta similitud, mientras que un valor cercano a 0 indica una baja similitud.

Los métodos de distancia calculan la diferencia entre un registro a con un registro b. Para esto existen muchos tipos de distancias, entre las más frecuentes están: Distancia Euclídea, Distancia de Manhattan, Distancia de Mahalanobis, entre otras más. Además, se pueden realizar estrategias de robustecimiento para evitar suposiciones iniciales de una distribución normal en los datos [18].

Mientras que la correlación basada en similitud es una técnica utilizada para medir cuán relacionados están dos elementos (por ejemplo, usuarios o productos) en términos de su comportamiento o atributos. Es similar a la similitud del coseno, pero en lugar de usar la magnitud de los vectores, se enfoca en la correlación entre ellos, ajustando por la media de las valoraciones o interacciones. Se suele utilizar la correlación de Pearson.

2. Métricas de evaluación de los sistemas de recomendación

Estas métricas de evaluación dependen del escenario o del tipo de modelo que se vaya a emplear; sin embargo, a continuación, se muestra las métricas de evaluación más usadas:

- Error Absoluto Medio (MAE):
  - $MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$ donde n es el número de observaciones, i es la observación, $y_i$ es el valor real y $\hat{y}_i$ es el valor pronosticado
- Raíz del Error Cuadrático Medio (RMSE):
  - $RMSE = \sqrt{\sum_{i=1}^{n}\frac{(\hat{y}_i - y_i)^2}{n}}$ donde n es el número de observaciones, i es la observación, $y_i$ es el valor real y $\hat{y}_i$ es el valor pronosticado
- Precisión:
  - $Precision = \frac{TP}{TP+FP}$
- Recall:
  - $Recall = \frac{TP}{TP+FN}$
- F1 score:
  - $F1\ score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$

Donde para la precisión, el recall y el f1 score, los TP (verdaderos positivos), FP (falsos positivos), FN (falsos negativos) se basan en la matriz de confusión:

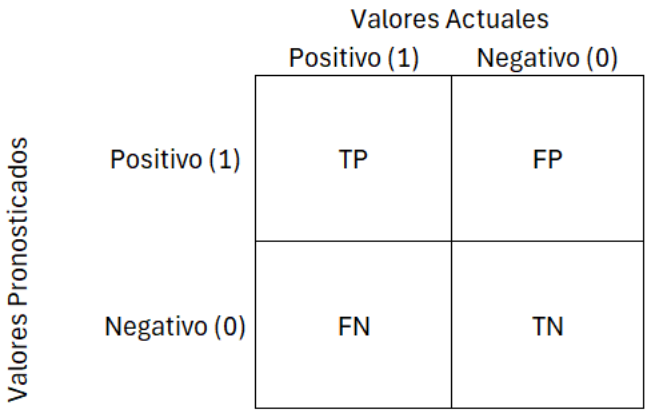



*Ilustración 5 Matriz de Confusión*

## 2.6 Metodología

La metodología que se utilizo fue la metodología CRISP-DM (*Cross-Industry Standard Process for Data Mining*) para proyectos de ciencia de datos implica varios pasos clave. Estos pasos incluyen la comprensión del negocio (*Business Understanding*), donde se definen los objetivos y requisitos del proyecto; la comprensión de los datos(*Data Understanding*), donde se recopilan y exploran los datos; la preparación de los datos (*Data Preparation*), que implica la limpieza, la transformación y la selección de datos; la modelización(*Modeling*), donde se aplican varias técnicas de modelado a los datos; la evaluación(*Evaluation*), donde se evalúa la eficacia de los modelos; y la implementación(*Deployment*), donde el modelo final se implementa en el proceso empresarial. La metodología CRISP-DM proporciona un enfoque estructurado para guiar a los científicos de datos y a los gerentes de proyectos a través de las diferentes fases de un proyecto de ciencia de datos, garantizando un proceso sistemático y efectivo de principio a fin [21].

Sin embargo, en este trabajo, debido a los alcances y los objetivos no se realizó la etapa de despliegue del modelo (*Deployment).* Se pasa a explicar en detalle que se realizó en cada una de las etapas para el óptimo desarrollo del trabajo de grado:

**Comprensión del negocio (*Business Understanding*):** en esta primera etapa se realizó reuniones con los dueños del problema, en este caso específico con el director de la unidad de desarrollo técnico e innovación, con el gerente de analítica y con el presidente de la organización (en este caso el co-director del trabajo de grado), con un único fin de plantear la situación actual del negocio y como por medio de los datos podríamos darle una solución a este dolor que tiene la compañía.

**Comprensión de los datos (Data Understanding):** en esta segunda etapa se desarrolló un análisis exploratorio de los datos, un análisis del ciclo de vida de los datos para poder determinar el origen, la cantidad, la calidad y la relevancia de los datos enfocando la respuesta del problema en los datos disponibles.

**Preparación de los datos (*Data Preparation*):** en esta etapa se debe garantizar la limpieza y depuración de la data y su disponibilidad para el modelo o los modelos que sean necesarios en la siguiente etapa.

**Modelización (*Modeling*):** en esta etapa se creó un sistema de recomendación hibrido en donde recopila información de los gustos de los clientes y su similitud entre zonas (nuevas y actuales), además de la similitud de los productos de ambas zonas.

**Evaluación (*Evaluation*):** por último, se evaluó el rendimiento del sistema realizando una validación de las recomendaciones de los productos que recomienda a un cliente actual, para determinar cuántas recomendaciones están dentro de la compra real de esos clientes.

## 2.8 Plan de Gestión de Datos

La empresa entregó los datos necesarios al autor del proyecto de grado con fines únicamente académicos, estos datos se entregaron en un formato csv donde el autor, profesores y director del proyecto de grado fueron los únicos que manipularon los archivos. Se entregaron 3 archivos csv, en donde cada archivo tendrá información de clientes (anónimamente, es decir, no tendrá nombre de los clientes sino un id), información de los productos (anónimamente, es decir, no tendrá nombre de los productos sino un id) y por último información de la interacción entre producto y cliente, es decir, información transaccional de compras de clientes con su fecha. Los resultados obtenidos por el modelo fueron compartidos únicamente con la universidad, compartiendo la documentación del proyecto de grado para su evaluación. Posteriormente, cuando se terminó el proyecto de grado, se destruyeron los archivos csv (no se guardó ninguna copia de estos). Al final, se entregó el modelo a la compañía para que esta determine su funcionamiento dentro de la organización. Siendo más detallados con los rubros que pide la universidad, a continuación, se responde a cada item:

1. **Tipos de datos y materiales producidos:**
    - **Datos de clientes:** Información anónima de clientes identificados únicamente mediante un ID.
    - **Datos de productos:** Información anónima de productos identificados únicamente mediante un ID.
    - **Datos de interacción:** Información transaccional de compras de clientes, incluyendo fechas de las transacciones.
    - **Programas informáticos:** Código del modelo de recomendación.
    - **Materiales de currículum:** Documentación y reportes del proyecto de grado para fines académicos.
2. **Normas para formato y contenido:**
    - **Formato de datos:** Archivos en formato CSV.
    - **Contenido de los datos:** Datos anonimizados para proteger la privacidad de los clientes y productos.
    - **Metadatos:** Descripción clara del significado de cada campo en los CSV.
3. **Políticas de acceso e intercambio:**
    - **Acceso restringido:** Solo el autor, profesores y director del proyecto de grado manipularon los archivos CSV.

- **Privacidad y confidencialidad:** Se asegurararon medidas para proteger la privacidad y confidencialidad de los datos.
- **Seguridad:** Implementación de medidas de seguridad para evitar accesos no autorizados.
- **Propiedad intelectual:** Los datos son propiedad de la empresa y su uso fue limitado a los fines del proyecto de grado.

4. **Políticas y disposiciones para la reutilización y redistribución:**
    - **Reutilización:** Los datos no se reutilizaron fuera del ámbito del proyecto de grado.
    - **Redistribución:** No se permitió la redistribución de los datos a terceros.
    - **Producción de derivados:** Los resultados y derivados del proyecto se compartieron únicamente con la universidad para evaluación académica.
5. **Planes de archivado y preservación del acceso:**
    - **Archivado:** Al finalizar el proyecto de grado, se destruyeron los archivos CSV, sin conservar ninguna copia.
    - **Preservación del acceso:** La documentación y los resultados del proyecto de grado se archivó para su evaluación académica, pero los datos originales no se conservaron.
    - **Entrega del modelo:** El modelo desarrollado se entregó a la empresa para que determine su implementación y funcionamiento dentro de la organización.

## 2.9 Aspectos éticos

La ética en este tipo de proyectos es crucial para asegurar que las aplicaciones de la tecnología beneficien a la sociedad, área organizacional o conjunto de personas con la problemática a resolver respetando los derechos de las personas y de las organizaciones, es por eso por lo que es importante definir los siguientes puntos:

**• Para qué se van a usar los datos?**

**• ¿Cuáles son los beneficios y quién se beneficiará?**

**• ¿Quién estará usando los datos?**

Los datos se usaron para crear, entrenar, evaluar un modelo de recomendación de productos para clientes internos de la empresa, los beneficiados los determina la empresa si se está desea implementar el proyecto como mecanismo de solución de la problemática que se presenta, estos datos los utilizaron únicamente el autor del proyecto de grado con sus profesores de la universidad. El consentimiento del propietario de los datos se obtuvo sin necesidad de realizar un documento de entrega formal y como se mencionó antes en la gestión de los datos, se entregaron los datos anonimizados con el fin de proteger la información de los clientes y de los productos de la organización.

# 3. Desarrollo de la solución

## 3.1.1 Datos

Como es de conocimiento, el primer paso que se debe de realizar en un proyecto de ciencia de datos es la recolección de los datos. En este proyecto se realizó una consulta directa a la base de datos interna de la compañía por medio de SQL server. Esta consulta permitió crear los conjuntos de datos necesarios. A continuación, se explicará cada conjunto de datos con sus características correspondientes:

### Df_caracteristicas_clientes_actuales

Este conjunto de datos contiene 15152 filas con 6 columnas. Las características son:

1. Clienteid (representa el cliente):
    a. Son 15152 valores distintos
2. Industria (representa el key de la industria del cliente):
    a. Son 20 industrias distintas
    b. Distribuidas:

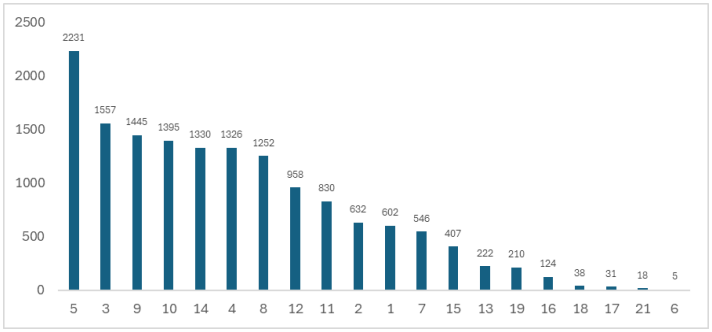



*Ilustración 6 Distribución de Industrias*

    c. Se observa que la distribución de las industrias de los clientes de la región actual está dividida en varias industrias por lo que no se tuvo sesgo alguno por esta variable.
3. Sub-industria (representa el key de la Sub-Industria del cliente):
    a. Son 59 Sub-Industrias distintas
    b. Distribuidas:

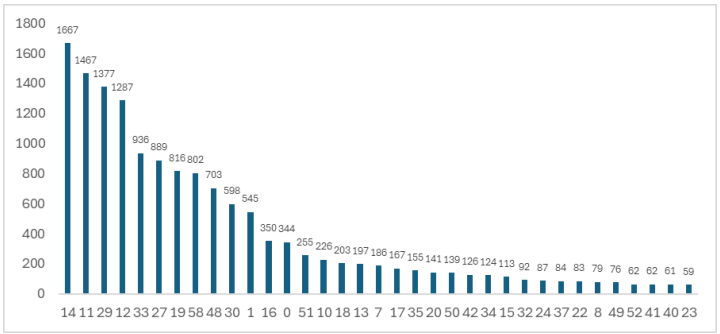



*Ilustración 7 Distribución de Sub-Industrias*

c. Se observa que la distribución de las sub industrias de los clientes de la región actual está dividida en varias industrias por lo que no se tuvo sesgo alguno por esta variable.

4. Segmento (representa el key del segmento del cliente):
    a. Son 7 segmentos distintos
    b. Distribuidos:

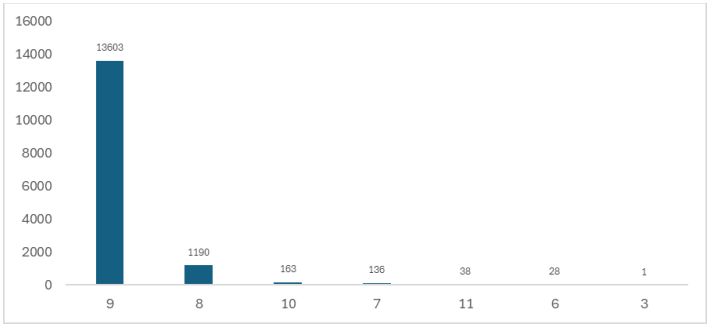



*Ilustración 8 Distribución de Segmento*

    c. Esta variable solo tiene representación en el key 9, esto puede incurrir al sistema a tener sesgos por la key 9 por lo que se descartó del modelo para que no se tuviera sesgo alguno.

5. Clasificación (representa el key de la clasificación del cliente):
    a. Son 8 Clasificaciones distintas
    b. Distribuidos:

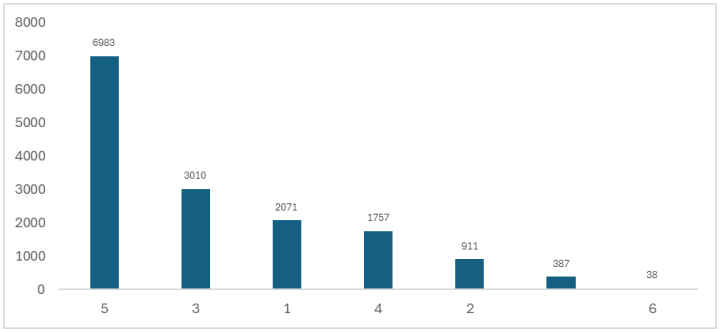



*Ilustración 9 Distribución de Clasificación*

    c. La distribución de esta variable tiene un ligero sesgo con el valor 5 pero no se descarta ya que los otros valores son representativos en el conjunto de datos,
6. Tipo_Segmento (representa el key del tipo de segmento del cliente)
    a. Son 7 Tipos de segmentos distintos
    b. Distribuidos:

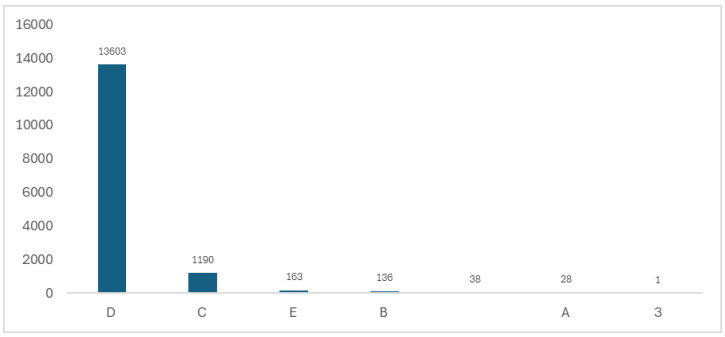



*Ilustración 10 Distribución de Tipo de segmento*

c. Esta variable solo tiene representación en el key D, esto puede incurrir al sistema a tener sesgos por la key_D, por lo que se descartó del modelo para que no se tuviera sesgo alguno.

**Df_caracteristicas_clientes_nuevos**

Este conjunto de datos contiene 181 filas con 6 columnas. Las características son:

1. Clienteid (representa el cliente):
   a. Son 181 valores distintos
2. Industria (representa el key de la industria del cliente):
   a. Son 8 industrias distintas
   b. Distribuidas:

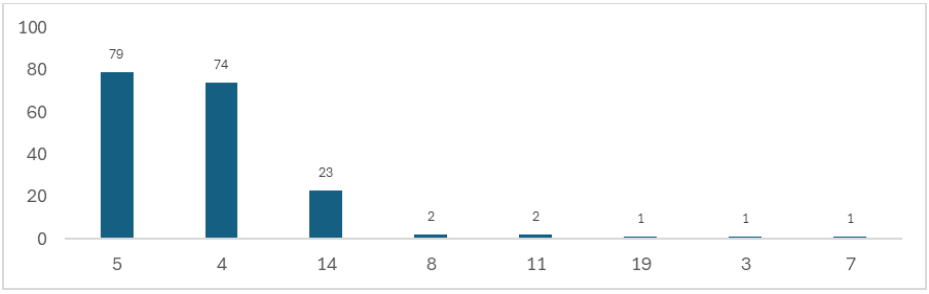



*Ilustración 11 Distribución de Industrias*

3. Se observa que la distribución de las industrias de los clientes de la región nueva está dividida en varias industrias por lo que no se tuvo sesgo alguno por esta variable.Subindustria (representa el key de la subindustria del cliente):
   a. Son 14 subindustrias distintas
   b. Distribuidas:

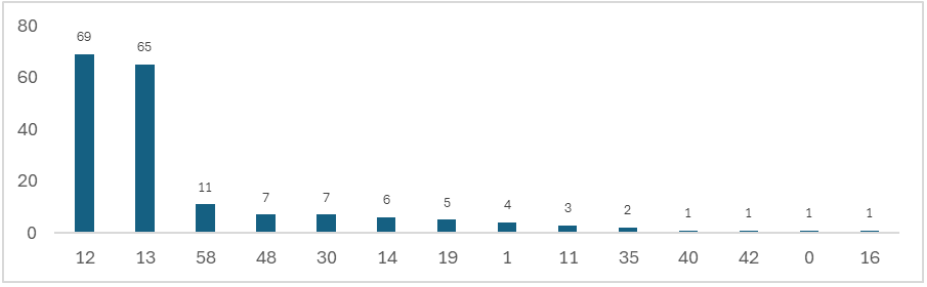



*Ilustración 12 Distribución de Sub-Industrias*

c. Se observa que la distribución de las sub-industrias de los clientes de la región nueva está dividida en varias industrias por lo que no se tuvo sesgo alguno por esta variable.

4. Segmento (representa el key del segmento del cliente):
   a. Son 6 segmentos distintos
   b. Distribuidos:

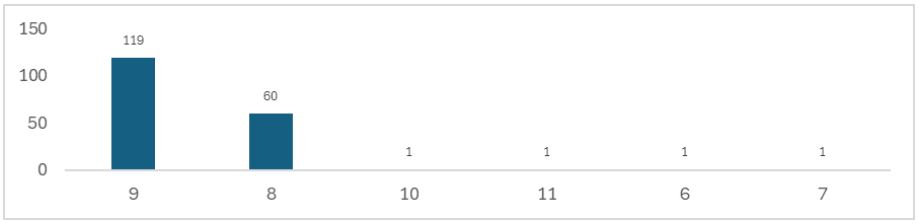



*Ilustración 13 Distribución de Segmento*

   c. Se observa que los key 9 y 8 son los valores que se llevan la mayoría del conjunto de datos

5. Clasificación (representa el key de la clasificación del cliente):
   a. Son 5 Clasificaciones distintas
   b. Distribuidos:

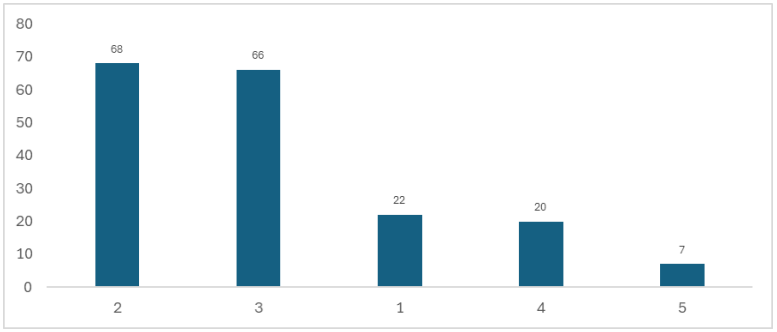



*Ilustración 14 Distribución de Clasificación*

   c. Se observa que la distribución de la clasificación de los clientes de la región nueva está dividida en varias industrias por lo que no se tuvo sesgo alguno por esta variable.

6. Tipo_Segmento (representa el key del tipo de segmento del cliente)
   a. Son 6 Tipos de segmentos distintos
   b. Distribuidos:

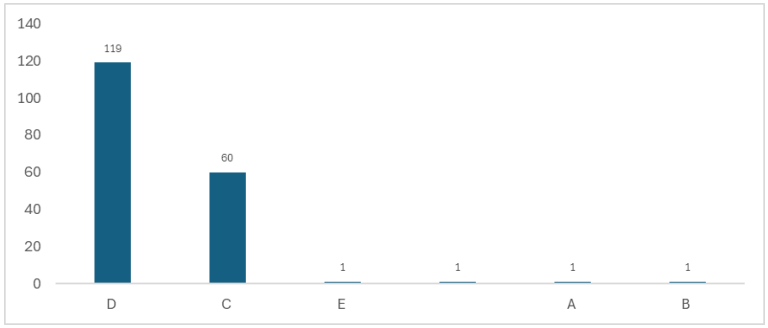



*Ilustración 15 Distribución de Tipo de segmento*

c. Se observa que los key D y C son los valores que se llevan la mayoría del conjunto de datos

**Df_caracteristicas_productos**

Este conjunto de datos contiene los productos de la compañía, cuenta con 14021 materiales con 6 columnas. A continuación, se explicará cada una de las características:

1. Materialid (representa los materiales):
   a. Son 14021 materiales distintos
2. TipoMaterialid (representa el tipo del material):
   a. Son 7 valores distintos
   b. Distribuidos:

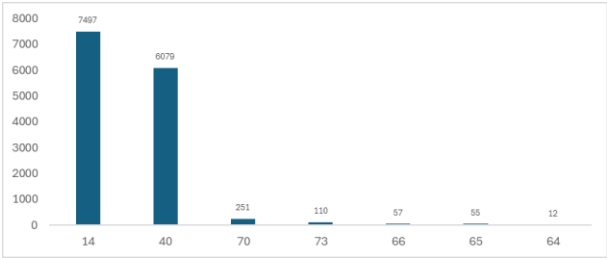



*Ilustración 16 Distribución de Tipo Material*

   c. Se observa que los key 14 y 40 son los valores que se llevan la mayoría del conjunto de datos
3. GrupoArticuloid (representa el grupo de los artículos):
   a. Son 82 valores distintos
   b. Distribuidos:

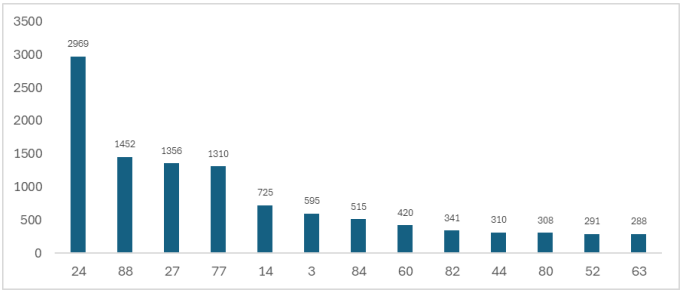



*Ilustración 17 Distribución de Grupo articulo*

   c. Se observa que la distribución de los valores esta entre varios grupos por lo que no existe ningún sesgo en esta variable.
4. JerMatDescripcion1 (representa una agrupación de abuelo del material):
   a. Son 325 valores distintos
   b. Distribuidos:

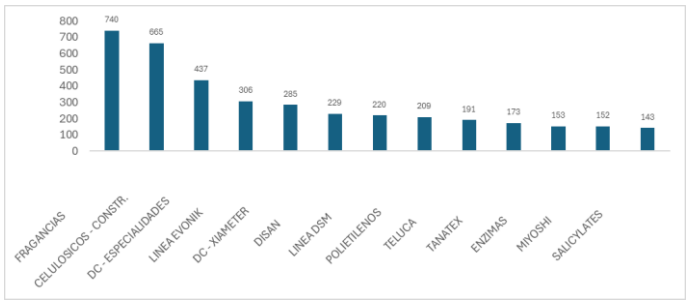



*Ilustración 18 Distribución de Jerarquía Material 1*

c. Se observa que la distribución de los valores esta entre varias jerarquías por lo que no existe ningún sesgo en esta variable.

5. JerMatDescripcion3 (Representa una agrupación de padre del material):
   a. Son 960 valores distintos
   b. Distribuidos:

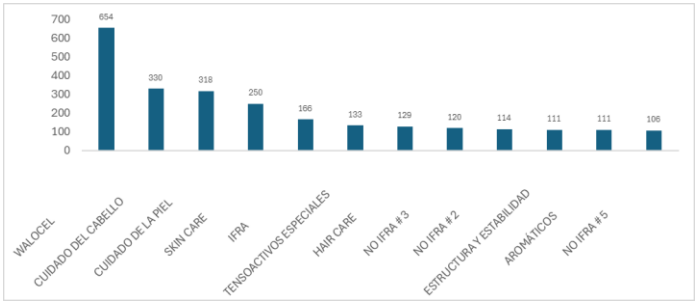



*Ilustración 19 Distribución de Jerarquía Material 3*

   c. Se observa que la distribución de los valores esta entre varias jerarquías por lo que no existe ningún sesgo en esta variable.

6. Portafolio (representa si el material está dentro del portafolio deseado).

**Df_interaccion_cliente-producto_actual**

Este es un conjunto de datos con la compra de materiales por cada cliente de la región actual, es decir existe una columna con los valores de clienteid y otra columna con los materialesid. Este conjunto de datos cuenta con 81800 registros de compras.

**Df_interaccion_cliente-producto_nuevo**

Este es un conjunto de datos con la compra de materiales por cada cliente de la nueva región, es decir existe una columna con los valores de clienteid y otra columna con los materialesid. Este conjunto de datos cuenta con 2384 registros de compras.

### 3.1.2 Transformación de datos

Se realizó varias transformaciones a los conjuntos de datos con el fin de limpiar y adecuar la data para que sea útil en el modelo que se planea crear. A continuación, se describen los procesos de transformación que se recibió cada conjunto de datos.

Se empezó con el conjunto de datos de productos, ya que como se ve en la descripción de los datos, existen 14021 materiales distintos por lo que es necesario agruparlo por jerarquías de material y se crea un nuevo conjunto de datos con valores únicos de TipoMaterialid, GrupoArticuloid, JerMatDescripcion 1 y JerMatDescripcion 3. A esta relación única se le asigna un nuevo id (prod_id), esto logra pasar de 14021 registros únicos a 1393 registros distintos, cerca del 90% de reducción, por lo que vamos a cambiar la distribución de esos materiales, posteriormente, se homologa el materialid con el prod_id correspondiente en el df de productos y en el df interacción cliente-producto, pasando de 81800 registros a 865 registros para la región actual y de 2384 a 622 registros en la nueva región.

Posteriormente, se desea aplicar la técnica de *one hot encoding* a los conjuntos de datos de las características tanto de los clientes como de los productos. Esta técnica se aplica debido a que los valores del conjunto de datos son valores de id, haciéndose referencia a una categoría. Para esto lo primero que debemos de hacer es unir los conjuntos de datos de características de clientes actuales con el de características de clientes nuevos. Después, por cada columna sacar los valores únicos de estos, con el fin de garantizar que todos los valores (tanto del conjunto de datos de clientes actuales como el de clientes nuevos) compartan las mismas características.

Con esto, se logró pasar de 6 características a 99 características un aumento considerable en la dimensión del conjunto de datos. Ahora las características pasan a ser los valores únicos de las 6 columnas, y el valor es un 1 si ese registro tenía ese valor único en la columna y un 0 en caso contrario. El mismo procedimiento se realiza para el conjunto de datos de producto y nos da como resultado 1370 características.

## 3.2 Modelado

Como los datos que tenemos a disposición son registros de compra y no de calificaciones del producto, es necesario enfocar en el proyecto en diseñar un modelo que se adapte a las condiciones de nuestros datos, para esto, se creó un modelo de recomendación de filtrado colaborativo híbrido basado en similitudes (tanto del usuario como del producto).

El primer paso es crear las matrices de similitud, tanto del usuario como del producto. Para esto se empleó la librería cosine_similarity de sklearn para realizar la función de similitud del coseno. A la matriz de similitud de usuarios se le pasa el conjunto de datos de características de usuarios actuales y el conjunto de datos de características de usuarios nuevos. Lo que nos devuelve una matriz de tamaño (#ClientesActuales, #ClientesNuevos); sus valores están entre -1 y 1 donde -1 corresponde a que son clientes totalmente opuestos y 1 son clientes iguales. Así mismo, se crea la matriz de similitud de productos. Como resultado la dimensión de esta matriz es de (#Productos, #Productos).

Posteriormente, se creó la matriz de interacción de clientes con los productos Actuales y Nuevos. Está se realiza pivoteando los conjuntos de datos de interacción clientes productos. Los valores son 1 si ese cliente ha comprado algún producto y 0 si no ha comprado ese producto, las matrices queda de dimensión (#Clientes, #Productos).

Estas matrices quedan con muchos registros en 0, son denominadas como matrices SPARSE, por lo que se puede trabajar con una librería csr_matrix de scipy para optimizar los recursos de la máquina.

Después, se procede a programar el algoritmo de recomendación, el cual utiliza las matrices de similitud para generar una matriz de puntuaciones de recomendación. Para cada cliente de la región 2 (cliente nuevo), se combinan la similitud de clientes y productos para generar una puntuación que mide cuán probable es que al cliente de la región 2 (cliente nuevo) le interese un producto de la región 1 (del portafolio). La fórmula general de este cálculo se basa en la siguiente expresión:

$$Puntuacion_{cliente,producto} = \sum similitud_{cliente} \times similitud_{producto}$$

Donde la similitud de los clientes se calcula entre un cliente de la región 1 (actuales) y un cliente de la región 2 (nuevo), esto lo hace iterando cada cliente de la región 2 y trayendo su similitud con el cliente de la región 1 (el valor de la matriz de similitud entre clientes) y la similitud de los productos se mide entre productos de ambas regiones (mismo proceso pero con la matriz de similitud de productos).

Entonces se evalúa para cada cliente de la región 2 (nuevo), para cada producto del portafolio, se calcula la similitud del cliente nuevo con todos los clientes actuales y se calcula la similitud de cada producto del portafolio con los productos de la región 2 (nueva) y se ponderan las dos similitudes.

Posteriormente, ese resultado se guarda en una matriz, en donde al final se va a obtener para cada cliente de la región 2 (nuevos) cuál va a ser su *score* para los productos del portafolio de la región 1.

## 3.3 Resultados

Se obtiene 2 resultados que a la compañía le interesa:

1. Para cada cliente de la región 2 (nuevo) el top 5 de productos del portafolio para recomendarles
2. Para cada producto del portafolio el top 5 de clientes de la región 2

Para esto se realiza para cada fila el top 5 de valores (responde al resultado 1) y después para cada columna el top 5 de valores (responde al resultado 2).

Obteniendo un resultado de esta forma:

Para el resultado 1:

*Cliente_r2_1: ['Producto_r1_5', 'Producto_r1_1', 'Producto_r1_3', 'Producto_r1_7', 'Producto_r1_15']*

*Cliente_r2_2: ['Producto_r1_5', 'Producto_r1_1', 'Producto_r1_3', 'Producto_r1_7', 'Producto_r1_15']*

*Cliente_r2_3: ['Producto_r1_5', 'Producto_r1_3', 'Producto_r1_1', 'Producto_r1_15', 'Producto_r1_14']*

*Cliente_r2_4: ['Producto_r1_5', 'Producto_r1_1', 'Producto_r1_3', 'Producto_r1_7', 'Producto_r1_15']*

*Cliente_r2_5: ['Producto_r1_5', 'Producto_r1_1', 'Producto_r1_3', 'Producto_r1_7', 'Producto_r1_15']*

*Nota: se muestran solo los 5 primeros clientes, pero el resultado es para cada cliente (181)

Para el resultado 2:

*Producto_r1_1: ['Cliente_r2_5', 'Cliente_r2_29', 'Cliente_r2_10', 'Cliente_r2_47', 'Cliente_r2_65']*

*Producto_r1_2: ['Cliente_r2_2', 'Cliente_r2_4', 'Cliente_r2_5', 'Cliente_r2_10', 'Cliente_r2_29']*

*Producto_r1_3: ['Cliente_r2_5', 'Cliente_r2_29', 'Cliente_r2_10', 'Cliente_r2_4', 'Cliente_r2_2']*

*Producto_r1_4: ['Cliente_r2_11', 'Cliente_r2_18', 'Cliente_r2_20', 'Cliente_r2_22', 'Cliente_r2_25']*

*Producto_r1_5: ['Cliente_r2_5', 'Cliente_r2_29', 'Cliente_r2_10', 'Cliente_r2_4', 'Cliente_r2_1']*

*Producto_r1_6: ['Cliente_r2_4', 'Cliente_r2_5', 'Cliente_r2_29', 'Cliente_r2_10', 'Cliente_r2_47']*

*Producto_r1_7: ['Cliente_r2_4', 'Cliente_r2_1', 'Cliente_r2_10', 'Cliente_r2_2', 'Cliente_r2_5']*

*Producto_r1_8: ['Cliente_r2_1', 'Cliente_r2_2', 'Cliente_r2_5', 'Cliente_r2_29', 'Cliente_r2_10']*

*Producto_r1_9: ['Cliente_r2_5', 'Cliente_r2_29', 'Cliente_r2_7', 'Cliente_r2_19', 'Cliente_r2_35']*

*Producto_r1_10: ['Cliente_r2_1', 'Cliente_r2_5', 'Cliente_r2_29', 'Cliente_r2_84', 'Cliente_r2_101']*

*Producto_r1_11: ['Cliente_r2_5', 'Cliente_r2_29', 'Cliente_r2_7', 'Cliente_r2_8', 'Cliente_r2_10']*

*Producto_r1_12: ['Cliente_r2_8', 'Cliente_r2_14', 'Cliente_r2_16', 'Cliente_r2_21', 'Cliente_r2_33']*

*Producto_r1_13: ['Cliente_r2_5', 'Cliente_r2_29', 'Cliente_r2_10', 'Cliente_r2_2', 'Cliente_r2_1']*

*Producto_r1_14: ['Cliente_r2_1', 'Cliente_r2_5', 'Cliente_r2_29', 'Cliente_r2_2', 'Cliente_r2_10']*

*Producto_r1_15: ['Cliente_r2_1', 'Cliente_r2_2', 'Cliente_r2_5', 'Cliente_r2_29', 'Cliente_r2_10']*

Para validar la información se extrajeron 15 clientes de la región 1 con sus compras y se comparó con los resultados que arroja el algoritmo obteniendo:

**Resultado del algoritmo:**

Cliente_val_1: ['Producto_r1_1', 'Producto_r1_3', 'Producto_r1_5', 'Producto_r1_7', 'Producto_r1_4']

Cliente_val_2: ['Producto_r1_1', 'Producto_r1_5', 'Producto_r1_3', 'Producto_r1_4', 'Producto_r1_7']

Cliente_val_3: ['Producto_r1_1', 'Producto_r1_5', 'Producto_r1_3', 'Producto_r1_7', 'Producto_r1_14']

Cliente_val_4: ['Producto_r1_1', 'Producto_r1_3', 'Producto_r1_5', 'Producto_r1_7', 'Producto_r1_14']

Cliente_val_5: ['Producto_r1_1', 'Producto_r1_3', 'Producto_r1_5', 'Producto_r1_7', 'Producto_r1_4']

Cliente_val_6: ['Producto_r1_1', 'Producto_r1_5', 'Producto_r1_3', 'Producto_r1_7', 'Producto_r1_14']

Cliente_val_7: ['Producto_r1_1', 'Producto_r1_5', 'Producto_r1_3', 'Producto_r1_14', 'Producto_r1_8']

Cliente_val_8: ['Producto_r1_1', 'Producto_r1_3', 'Producto_r1_5', 'Producto_r1_7', 'Producto_r1_14']

Cliente_val_9: ['Producto_r1_1', 'Producto_r1_3', 'Producto_r1_5', 'Producto_r1_7', 'Producto_r1_8']

Cliente_val_10: ['Producto_r1_1', 'Producto_r1_5', 'Producto_r1_3', 'Producto_r1_7', 'Producto_r1_14']

Cliente_val_11: ['Producto_r1_1', 'Producto_r1_3', 'Producto_r1_5', 'Producto_r1_7', 'Producto_r1_14']

Cliente_val_12: ['Producto_r1_1', 'Producto_r1_3', 'Producto_r1_5', 'Producto_r1_7', 'Producto_r1_14']

Cliente_val_13: ['Producto_r1_1', 'Producto_r1_3', 'Producto_r1_5', 'Producto_r1_7', 'Producto_r1_4']

Cliente_val_14: ['Producto_r1_1', 'Producto_r1_5', 'Producto_r1_4', 'Producto_r1_14', 'Producto_r1_8']

Cliente_val_15: ['Producto_r1_1', 'Producto_r1_3', 'Producto_r1_5', 'Producto_r1_7', 'Producto_r1_14']

**Resultado real:**

Cliente_val_1: ['Producto_r1_8', 'Producto_r1_10', 'Producto_r1_5', 'Producto_r1_7', 'Producto_r1_4']

Cliente_val_2: ['Producto_r1_6', 'Producto_r1_5', 'Producto_r1_3', 'Producto_r1_4', 'Producto_r1_7']

Cliente_val_3: ['Producto_r1_11', 'Producto_r1_2', 'Producto_r1_9', 'Producto_r1_7', 'Producto_r1_14']

Cliente_val_4: ['Producto_r1_1', 'Producto_r1_3', 'Producto_r1_5', 'Producto_r1_7', 'Producto_r1_14']

Cliente_val_5: ['Producto_r1_1', 'Producto_r1_2', 'Producto_r1_11', 'Producto_r1_13', 'Producto_r1_5']

Cliente_val_6: ['Producto_r1_13', 'Producto_r1_5', 'Producto_r1_3', 'Producto_r1_7', 'Producto_r1_14']

Cliente_val_7: ['Producto_r1_12', 'Producto_r1_9', 'Producto_r1_3', 'Producto_r1_14', 'Producto_r1_8']

Cliente_val_8: ['Producto_r1_2', 'Producto_r1_3', 'Producto_r1_5', 'Producto_r1_7', 'Producto_r1_14']

Cliente_val_9: ['Producto_r1_11', 'Producto_r1_2', 'Producto_r1_5', 'Producto_r1_7', 'Producto_r1_8']

Cliente_val_10: ['Producto_r1_1', 'Producto_r1_5', 'Producto_r1_3', 'Producto_r1_7', 'Producto_r1_14']

Cliente_val_11: ['Producto_r1_1', 'Producto_r1_2', 'Producto_r1_5', 'Producto_r1_7', 'Producto_r1_14']

Cliente_val_12: ['Producto_r1_2', 'Producto_r1_8', 'Producto_r1_5', 'Producto_r1_7', 'Producto_r1_14']

Cliente_val_13: ['Producto_r1_1', 'Producto_r1_3', 'Producto_r1_5', 'Producto_r1_15', 'Producto_r1_4']

Cliente_val_14: ['Producto_r1_12', 'Producto_r1_13', 'Producto_r1_4', 'Producto_r1_14', 'Producto_r1_8']

Cliente_val_15: ['Producto_r1_11', 'Producto_r1_15', 'Producto_r1_5', 'Producto_r1_7', 'Producto_r1_14']

**Evaluación de cada cliente:**

Cliente_val_1 = 3/5 (60%)

Cliente_val_2 = 4/5 (80%)

Cliente_val_3 = 2/5 (40%)

Cliente_val_4 = 5/5 (100%)

Cliente_val_5 = 2/5 (40%)

Cliente_val_6 = 4/5 (80%)

Cliente_val_7 = 3/5 (60%)

Cliente_val_8 = 4/5 (80%)

Cliente_val_9 = 3/5 (60%)

Cliente_val_10: = 5/5 (100%)

Cliente_val_11 = 4/5 (80%)

Cliente_val_12 = 3/5 (60%)

Cliente_val_13 = 4/5 (80%)

Cliente_val_14 = 3/5 (60%)

Cliente_val_15 = 3/5 (60%)

Para una ponderación total de:

Aciertos: 52

Total de muestras: 75

***Accuracy*** = 69.3%

Se escoge la métrica *accuracy* por el tipo de datos que se tiene y el tipo de problema, ya que se quiere validar de las compras ya ejecutadas, cuantas recomendó el sistema de recomendación. Se descartaron las otras, como MAE Y RMSE por que no es un sistema de recomendación que predice algún tipo de rating por lo que no hay forma de saber cuál es el error entre la predicción y el valor real.

## 4. Conclusiones:

1. Este modelo de recomendación es una herramienta valiosa para el éxito del lanzamiento de un nuevo portafolio de productos. Al identificar similitudes entre los usuarios y sus patrones de compra, el modelo proporciona, para cada producto del portafolio, un top 5 de clientes potenciales, y para cada nuevo cliente, un top 5 de productos que probablemente compraría. Esto facilita que el equipo comercial enfoque sus esfuerzos en ofrecer productos relevantes, maximizando las oportunidades de éxito en la entrada al mercado.

2. El algoritmo desarrollado utiliza la similitud entre los clientes de la región actual y los de la nueva región, combinada con la similitud entre los productos de ambas regiones. Esto permite crear un modelo de recomendación basado en la similitud del coseno que sugiere productos del nuevo portafolio a clientes con mayor potencial de compra. El modelo optimiza el enfoque de la compañía, permitiendo dirigir eficientemente los recursos hacia los clientes más prometedores.

3. El modelo alcanzó un 69.3% de *accuracy*, lo que significa que aproximadamente el 70% de las compras recomendadas a los clientes actuales se concretaron. Este resultado es muy positivo para el negocio, ya que reduce la incertidumbre al ofrecer productos a los clientes. En lugar de enfrentar un 100% de incertidumbre al hacer una recomendación, el equipo comercial ahora trabaja con solo un 30% de margen de error, mejorando significativamente la efectividad de sus esfuerzos de venta.

4. El modelo actual de la compañía tiene una métrica de 53.4% de *accuracy*, por lo que este modelo superó en 15 puntos porcentuales el anterior, lo cual a la compañía le favorece ya que se podrán realizar mejores recomendaciones a los comerciales.

## 5. Código fuente

Para facilitar la revisión y reproducción de los resultados obtenidos en este trabajo de grado, se ha creado un repositorio en GitHub que contiene todo el código fuente, scripts y notebooks utilizados durante el desarrollo del proyecto. El repositorio está estructurado de manera que sea fácil de navegar y utilizar.

El acceso al repositorio se puede realizar a través del siguiente enlace:

*https://github.com/pisazah/Trabajo_Grado_Pablo_Isaza.git*

## 6. Referencias bibliográficas


[1] Ricci, Francesco & Rokach, Lior & Shapira, Bracha. (2010). Recommender Systems Handbook. 10.1007/978-0-387-85820-3_1.

[2] Antiopi, Panteli & Boutsinas, Basilis. (2023). Addressing the Cold-Start Problem in Recommender Systems Based on Frequent Patterns. Algorithms. 16. 182. 10.3390/a16040182.

[3] Sinha, Bam & Dhanalakshmi, R.. (2019). Evolution of recommender system over the time. Soft Computing. 23. 10.1007/s00500-019-04143-8.

[4] M. Shvarts, M. Lobur and Y. Stekh, "Some trends in modern recommender systems," 2017 XIIIth International Conference on Perspective Technologies and Methods in MEMS Design (MEMSTECH), Lviv, Ukraine, 2017, pp. 167-169, doi: 10.1109/MEMSTECH.2017.7937559. keywords: {Recommender systems;Semantics;Context;Decision making;Ontologies;Biological system modeling;Recommender system;user profile;rating score;similarity coefficient;collaborative filtering;content filtering;active structures of knowledge;semantic models of interests}

[5] Webb, Geoffrey & Pazzani, Michael & Billsus, Daniel. (2001). Machine Learning for User Modeling. User Model. User-Adapt. Interact.. 11. 19-29. 10.1023/A:1011117102175.

[6] Adomavicius, Gediminas & Tuzhilin, Alexander. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. Knowledge and Data Engineering, IEEE Transactions on. 17. 734-749. 10.1109/TKDE.2005.99.

[7] Ma, Hao & King, Irwin & Lyu, Michael. (2007). Effective missing data prediction for collaborative filtering. 39-46. 10.1145/1277741.1277751.

[8] Mehrbakhsh Nilashi, Othman Ibrahim, Karamollah Bagherifard, A recommender system based on collaborative filtering using ontology and dimensionality reduction techniques, Expert Systems with Applications,2018, https://doi.org/10.1016/j.eswa.2017.09.058.

[9] Isinkaye, Folasade & Folajimi, Yetunde & Ojokoh, Bolanle. (2015). Recommendation systems: Principles, methods and evaluation. Egyptian Informatics Journal. 16. 10.1016/j.eij.2015.06.005.

[10] Schafer, Ben & Konstan, Joseph & Riedl, John. (1999). Recommender Systems in E-Commerce. 1st ACM Conference on Electronic Commerce, Denver, Colorado, United States. 10.1145/336992.337035.

[11] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, and John Riedl. 1994. GroupLens: an open architecture for collaborative filtering of netnews. In Proceedings of the 1994 ACM conference on Computer supported cooperative work (CSCW '94). Association for Computing Machinery, New York, NY, USA, 175–186. https://doi.org/10.1145/192844.192905

[12] Breese, J. & Heckerman, David & Kadie, Carl. (1998). Empiricial analysis of predictive algorithms for collaborative ltering.

[13] Gui-Rong Xue, Chenxi Lin, Qiang Yang, WenSi Xi, Hua-Jun Zeng, Yong Yu, and Zheng Chen. 2005. Scalable collaborative filtering using cluster-based smoothing. In Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '05). Association for Computing Machinery, New York, NY, USA, 114–121. https://doi.org/10.1145/1076034.1076056

[14] D. Mladenic, "Text-learning and related intelligent agents: a survey," in IEEE Intelligent Systems and their Applications, vol. 14, no. 4, pp. 44-54, July-Aug. 1999, doi: 10.1109/5254.784084.

keywords: {World Wide Web;Intelligent agent;Internet;Marine animals;National electric code;Machine learning;Information retrieval;Computer science;Speech;Calendars},

[15] Aljunid, M.F., Manjaiah, D.H. (2019). Movie Recommender System Based on Collaborative Filtering Using Apache Spark. In: Balas, V., Sharma, N., Chakrabarti, A. (eds) Data Management, Analytics and Innovation. Advances in Intelligent Systems and Computing, vol 839. Springer, Singapore. https://doi.org/10.1007/978-981-13-1274-8_22

[16] Shoham, Y. (1997). Combining content-based and collaborative recommendation. Communications of the ACM.

[17] Fayyaz, Zeshan, Mahsa Ebrahimian, Dina Nawara, Ahmed Ibrahim, and Rasha Kashef. 2020. "Recommendation Systems: Algorithms, Challenges, Metrics, and Business Opportunities" Applied Sciences 10, no. 21: 7748. https://doi.org/10.3390/app10217748

[18] Fethi Fkih, Similarity measures for Collaborative Filtering-based Recommender Systems: Review and experimental comparison, Journal of King Saud University - Computer and Information Sciences, Volume 34, Issue 9, 2022, Pages 7645-7669, ISSN 1319-1578, https://doi.org/10.1016/j.jksuci.2021.09.014.

[19] Jaeger, A., & Banks, D. (2023). Cluster analysis: A modern statistical review. WIREs Computational Statistics, 15(3), e1597. https://doi.org/10.1002/wics.1597

[20] (Tatu, Kokkonen. (2016). Business case sales planning concept for new products and product portfolio).

[21] (Fatima, Sajid, Butt., Jörg, Schäfer., Matthias, Wagner., Dirk, Stegelmeyer., D., Gómez-Ullate, Oteiza. (2023). Application of CRISP-DM and DMME to a Case Study of Condition Monitoring of Lens Coating Machines.)

[22] Yang, S., Zhou, Q., Wang, Q. (2023). Clustering of Bandit with Frequency-Dependent Information Sharing. In: Kamps, J., et al. Advances in Information Retrieval. ECIR 2023. Lecture Notes in Computer Science, vol 13981. Springer, Cham. https://doi.org/10.1007/978-3-031-28238-6_18

[23] Goldberg, K., Roeder, T., Gupta, D., & Perkins, C. (2001). Eigentaste: A constant time collaborative filtering algorithm. information retrieval, 4, 133-151.

[24] Shu, Z. (2021). Perspectiva del DoubleDQN en los sistemas de recomendación para resolver el problema de item cold-start.

Medellín, 08 de octubre de 2024

**Señores:**

Centro Cultural Biblioteca Luis Echavarría Villegas

Coordinador Maestría en Ciencia de los Datos y Analítica

Universidad EAFIT

Medellín

**Asunto:** Entrega documento final trabajo de grado

Apreciados señores:

En mi condición de asesor del proyecto de grado titulado: "***Modelo de Recomendación de Nuevos Productos a Clientes Actuales***", certifico que el trabajo realizado cumple con las exigencias académicas y metodológicas establecidas; así como con los requisitos de forma del trabajo, de citación y de bibliografía. Por lo anterior, confirmo que el documento puede ser aceptado para que sus autores opten al título al cual aspiran.

A continuación, confirmo los datos del (los) autor (es):

Pablo Isaza Higuera

Campo registro autores

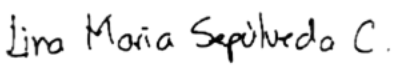
**Nombres y apellidos completos:** Pablo Isaza Higuera

**No. Documento de identidad:** 1040758693

**Programa académico:** Maestría en Ciencia de los Datos y Analítica

**Correo electrónico institucional:** pisazah@eafit.edu.co

Atentamente,



**Firma:** _____ **Nombre:** Lina María Sepúlveda Cano

**No. Documento de identidad:** 41’954.575

**Correo electrónico:** lmsepulvec@eafit.edu.co