ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

Diseño de un Sistema de Recomendaciones de productos utilizando Redes Neuronales de Grafos para clientes de campañas de puntos

PROYECTO DE TITULACIÓN

Previo la obtención del Título de:

Magister en Ciencias de Datos

Presentado por:

Angel Jonathan Merizalde Medina

GUAYAQUIL - ECUADOR

Año: 2024

DEDICATORIA

A mi esposa e hijos, padres y hermanos, familia, amigos y a todos aquellos que me han dado inspiración y fuerzas a través de los años.

AGRADECIMIENTOS

Mi sincero agradecimiento al MsC. Eduardo Cruz, por su guía y apoyo en el desarrollo de este proyecto.

Y a todos los que compartieron conmigo su conocimiento y experiencia, en lo académico, profesional y personal.

DECLARACIÓN EXPRESA

“Los derechos de titularidad y explotación, me corresponde conforme al reglamento de propiedad intelectual de la institución; *Angel Jonathan Merizalde Medina,* doy mi consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual"

Firmado electrónicamente por:
ANGEL JONATHAN
MERIZALDE MEDINA

Angel Jonathan Merizalde Medina

COMITÉ EVALUADOR

EDUARDO SEGUNDO CRUZ RAMIREZ

Firmado digitalmente por EDUARDO SEGUNDO CRUZ RAMIREZ
Fecha: 2024.06.28 10:22:25 -05'00'

MsC. Eduardo Cruz Ramírez

PROFESOR TUTOR

Firmado electrónicamente por:
ALLAN ROBERTO AVENDAÑO SUDARIO

MsC. Allan Avendaño Sudario

PROFESOR EVALUADOR

RESUMEN

Los sistemas de recomendación se han convertido en soluciones tecnológicas altamente utilizadas en el sector de *retail*, generando sugerencias personalizadas de compra enfocadas en las preferencias de los clientes, con el propósito de incrementar las ventas. Esto es relevante para artículos de baja rotación o con problemas de mercadeo. Diversas arquitecturas se implementan para desarrollar estos sistemas; sin embargo, aquellas basadas en modelos de *Deep learning* y redes neuronales artificiales permiten extraer características adicionales de los datos, obteniendo una representación más acertada de patrones ocultos en las ventas. Este documento describe el proceso de diseño de un sistema de recomendaciones para una empresa de *retail* basada en Ecuador, enfocado en artículos que los clientes pueden adquirir canjeando puntos de sus cuentas del programa de fidelización. Para ello, cada transacción fue considerada como un grafo, cargado en una Red neuronal de grafos para predecir el departamento del artículo canjeado a través de una tarea de clasificación de grafos. Se realizaron múltiples experimentos durante el entrenamiento para seleccionar el algoritmo base, obteniendo los mejores resultados de predicciones para las métricas seleccionadas con GraphSAGE a través de su variante SAGEConv. Las recomendaciones de compra generadas se ofrecieron a un grupo de clientes seleccionado usando una variante del análisis RFM. Se construyó un *dashboard* interactivo para el análisis semanal que realiza el área de publicidad. Estos resultados se almacenaron en una base de datos dedicada. Las compras realizadas posteriormente mostraron una tasa de conversión significativa, demostrando el impacto de las recomendaciones en las ventas.

Palabras Clave: Sistemas de recomendación, Red neuronal de grafos, clasificación de grafos, GraphSAGE, RFM.

ABSTRACT

Recommender systems have become highly utilized technological solutions in the retail sector, generating personalized offerings focused on customer preferences, with the purpose of increasing sales. This is relevant for low inventory turnover items or those with marketing issues. Diverse architectures are implemented to develop these systems; however, those based on Deep learning models and artificial neural networks allow additional characteristics to be extracted from the data, resulting in a more accurate representation of hidden sales patterns. This document describes the process of designing a recommender system for a retail company based in Ecuador, focused on items that customers can redeem using points from their loyalty program accounts. To achieve this, each transaction would be treated as a graph, loaded into a Graph neural network to predict the department of the redeemed item through a graph classification task. Numerous experiments were conducted during training to select the base algorithm, and the best prediction results for the selected metrics were obtained with GraphSAGE through the SAGEConv variant. The generated purchase recommendations were offered to a group of customers selected using a variant of RFM analysis. An interactive dashboard was developed for the weekly analysis conducted by the advertising department. The results were registered in a dedicated database. Subsequent purchases showed a significant conversion rate, proving the impact of recommendations on sales.

Keywords: *Recommender systems, Graph neural network, graph classification, GraphSAGE, RFM.*

ÍNDICE GENERAL

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

CAPÍTULO 1

1. PLANTEAMIENTO DEL PROBLEMA

1.1 Introducción

Retailers S.A., empresa ecuatoriana dedicada a la venta al por menor a nivel nacional, ofrece a sus clientes un programa de recompensas basado en puntos, los cuales se acumulan en su cuenta de fidelización de acuerdo con el monto de sus compras. La empresa realiza varias veces al año campañas promocionales con el objetivo que los clientes canjeen sus puntos a través de la compra de ciertos artículos, recibiendo a cambio un descuento especial.

La mayoría de los artículos pertenecientes a las campañas de canje corresponden a artículos de hogar, electrodomésticos, ferretería, juguetes, entre otros. La duración de los artículos en las campañas es variable: pueden mantenerse un mínimo de un mes, ser renovados mensualmente o retirados de la lista de canjeables luego de varias semanas. El seguimiento de los canjes realizados está a cargo de la Gerencia Comercial, la cual recibe semanalmente reportes generados por el área de TI para determinar si se deben realizar cambios en los artículos de la campaña.

Las ventas realizadas bajo la modalidad descrita se consideran como una venta especial de canje de puntos; sin embargo, estas ventas no contienen solamente a los artículos participantes en las campañas, sino que también pueden incluir artículos sin descuentos o participantes en otras promociones no relacionadas.

1.2 Justificación

La sobre existencia de artículos y la dificultad en comercializarlos efectivamente, ocasiona que el costo del inventario no pueda transformarse en ingresos generados por ventas, aumentando su obsolescencia. Algunas causas posibles son un precio inadecuado, baja demanda, posicionamiento incorrecto, pero también una inadecuada estrategia de mercadeo. Si bien la compañía realiza una publicidad de los productos canjeables, las técnicas actuales son estáticas e iguales para todos los clientes, y, por

consiguiente, tienen un bajo nivel de convencimiento de compra al ignorar sus preferencias de consumo.

Los clientes participantes en campañas de canje tienen diferentes opciones de categorías y precios de artículos, por lo cual es importante conocer qué buscan comprar y analizar las opciones a ofrecer. Sin embargo, la analítica de ventas actual de la compañía realiza un enfoque basado solo en monto y volúmenes de venta, dejando de lado el profundizar en las relaciones entre los clientes y los artículos que adquieren, y al mismo tiempo, de las relaciones entre los diferentes grupos de artículos.

Para cumplir los objetivos estratégicos de la empresa, tales como afianzar su posición de empresa líder en el mercado de *retail*, ofreciendo beneficios nuevos y exclusivos que consoliden su cercanía con las necesidades y preferencias de los clientes, la recomendación de artículos, basada en los patrones ocultos o el descubrimiento de asociaciones entre los artículos de una venta y los artículos de canje puede ayudar al desarrollo de nuevas estrategias de marketing gracias al conocimiento adquirido de los artículos que se adquieren juntos [1], buscando incrementar efectivamente las ventas de estos productos.

1.3 Solución propuesta

Los sistemas de recomendaciones basados en modelos de *Machine learning* o Aprendizaje automático constituyen una herramienta de análisis con la que no cuenta la compañía, por consiguiente, una solución desarrollada con este objetivo proveerá información valiosa para explotar el conocimiento inherente a los hábitos de consumo de clientes, y determinar qué tipos de artículos estarían interesados en adquirir bajo este esquema, de un modo más objetivo del que pueden ofrecer estudios de mercadeo contratados para el efecto.

Los resultados entregados por el sistema constituirán además un insumo en el diseño de las comunicaciones que elabora el área de Marketing para llegar a los clientes, ya sea mediante correos electrónicos, información en el sitio web de la compañía u otras estrategias de comunicación. A diferencia del marketing tradicional en el que se envía el mismo mensaje a una gran audiencia, con el marketing de precisión se entrega diferente

información a varios segmentos de una audiencia objetivo [2]. Estos resultados podrán ser visualizados a medida que la información se cargue en el sistema, a diferencia de los retrasos actuales ocasionados por la solicitud de información manual, su extracción y manipulación, y almacenamiento en medios dependientes de los usuarios, garantizando una entrega oportuna que permita responder de manera inmediata interrogantes sobre el cumplimiento de los objetivos particulares de una campaña, o generales de la empresa.

El sistema propuesto se compone de tres módulos de procesamiento:

- **Módulo de Captura.** Su función es extraer la información del sistema de Punto de Venta referente a las compras realizadas por los clientes, seleccionando los atributos necesarios para el análisis.
- **Motor de Recomendaciones.** Analiza las relaciones entre grupos de artículos presentes en una transacción de compra (*baskets*), y genera las listas de tipos de artículos recomendados para compra.
- **Módulo de Visualización.** Permite visualizar los resultados de canjes y recomendaciones obtenidas para la toma de decisiones.

1.4 Objetivos

1.4.1 Objetivo general

Construir un modelo para un Sistema de recomendación de tipos de productos de canjes, utilizando Redes Neuronales basadas en Grafos.

1.4.2 Objetivos específicos

1. Analizar la información del Sistema de Punto de Venta para obtener los atributos necesarios para la construcción del modelo.
2. Determinar las asociaciones entre los tipos de artículos adquiridos en una compra como una tarea de clasificación de grafos, utilizando filtros basados en contenido.
3. Obtener las recomendaciones de tipos de productos generadas a partir del análisis de las compras realizadas por los clientes.

4. Implementar el modelo seleccionado como base del motor de recomendaciones y su correspondiente método de visualización de resultados.

1.5 Descripción del problema

Las campañas de canje generalmente están compuestas de artículos de baja rotación, o que corresponden a importaciones antiguas o con sobre existencias, a los que la compañía decide otorgar un descuento que incentive la compra por parte de sus clientes fidelizados, con la consiguiente reducción de los costos asociados de inventario. Múltiples estrategias se aplican para lograr este objetivo: reducción de la cantidad necesaria de puntos para canje, ubicación destacada en perchas, o renovación frecuente de la permanencia del artículo en la lista. Es común que muchos artículos se mantengan en campaña durante varios meses, por cuanto no se cumplen las expectativas de compra definidas por la Gerencia. Esto tiene como efecto secundario la poca variabilidad de la lista de artículos canjeables, volviéndola poco atractiva a los clientes, reduciendo aún más las posibilidades de compra en el tiempo.

Los puntos otorgados a los clientes tienen su origen en la compra mensual de una cantidad equivalente en dólares que la empresa hace a otra compañía afiliada, y que debe ser devengada a través de su utilización a través de los canjes. de modo que pueda recuperarse la inversión, mediante el monto parcial pagado por el cliente en el canje, y el monto de los descuentos que se cobran a los proveedores de los artículos, conforme a acuerdos comerciales preexistentes. La empresa fija mensualmente las metas que deben lograrse para el consumo de esos puntos, sin embargo, estas no siempre se cumplen, debido a que muchos clientes nunca llegan a utilizar sus puntos acumulados dentro del período de vigencia desde la compra en la cual fueron ganados, siendo una de las razones la falta de un conocimiento oportuno sobre en cuáles artículos pueden utilizarlos. Esto ocasiona que al final del mes, además de la sobre existencia de artículos, se tenga una sobre existencia de puntos, que representan un costo contable importante de valores invertidos y que no se han podido recuperar.

La empresa es consciente que la tendencia en el sector de *retail* está cambiando del concepto de la solicitud/provisión de productos hacia una visión orientada a servicios habilitados en redes y sistemas de información [3], con el objetivo de generar un mayor

sentido de pertenencia hacia la marca, buscando potenciar los beneficios para sus clientes dentro de un marco de soluciones digitales. Conoce además que las estrategias de mercadeo genéricas muchas veces no tienen los resultados esperados, al no ser específicas para las personas [4, p. 572], contribuyendo a la sobrecarga de información el ofrecer artículos que los clientes no estarían dispuestos a comprar o no necesitarían en absoluto. El impulso de las campañas es genérico para todos los clientes, sin importar su historial de consumos, o los artículos que puede canjear con puntos. El no disponer de una manera efectiva de hacer conocer a los clientes los artículos que estarían más interesados en canjear no contribuye a mejorar las ventas de los artículos que la compañía desea ofrecer.

Los reportes para seguimiento semanal que recibe la Gerencia Comercial comprenden los totales de canjes generados desde el sistema de Punto de Venta, y de estadísticas de ventas, generados por el área de Analítica. La información recibida se unifica manualmente a través de Hojas de cálculo, lo cual implica un tiempo determinado de atención de un día dedicado a esta actividad al inicio de cada semana, incluyendo el tiempo que le toma a las áreas de Sistemas y Analítica entregar la información requerida, y que puede aumentar en virtud de la carga de trabajo del personal a cargo de estas actividades.

1.6 Método

La solución propuesta tomará en consideración las compras efectuadas por los clientes durante un período de cinco meses correspondientes a la última campaña definida de artículos de canje, con el objetivo de construir una Red Neuronal basada en Grafos (GNN) que analice las interacciones homogéneas entre los tipos de artículos adquiridos, con el objetivo de recomendar, a través de un método de filtrado basado en contenido (*Content-based filtering*) los tipos de artículos de canje sugeridos para sus compras. Los datos con los que se establecerá y probará la solución estarán basados en información real de ventas, correspondientes a información de campañas en curso y futuras.

El sistema se compone de tres fases de implementación. En la primera, los datos de ventas provenientes del sistema de Punto de Venta de la empresa son analizados y

transformados para cargar la Red Neuronal. Luego, se modelarán los datos dentro de la Red Neuronal para generar como resultados los artículos de canje sugeridos. En la última fase, las listas de recomendación de productos para clientes son preparadas y puestas a disposición para consumo de diferentes canales de presentación.

1.6.1 Red neuronal basada en grafos

Investigaciones recientes han demostrado que uno de los mejores métodos para modelar las entidades que intervienen en problemas relacionados a venta de artículos, sus interacciones, y, sobre todo para el diseño de sistemas de recomendaciones, es identificar la intención de la cesta de compra (*basket*), siendo esta cesta el conjunto de ítems adquiridos por el cliente en una visita de compra [5]. Una vez definida la noción de un *basket*, se busca recomendar otro artículo para agregar a dicho *basket* [6]. Definiendo las diferentes clases de artículos de la venta como nodos en un grafo, se podrá modelar la solución como una tarea de clasificación de grafos (*graph classification*), donde se busque establecer la clase de artículos de canje que se recomienda comprar. La red neuronal irá aprendiendo de las interacciones entre los tipos de artículos del *basket*, comparándolos con los tipos de artículos canjeables que se incluyeron en la transacción de venta original, con el objetivo de generar la lista de recomendaciones para canje.

1.6.2 Filtrado de datos

Los sistemas de recomendación (*Recommendation Systems*, RS) comprenden tecnologías de filtrado de datos que permiten predecir el interés de los usuarios por artículos que no hayan visto antes [4, p. 571]. La lista de productos recomendados se obtiene mediante algoritmos heurísticos basados en reglas.

Los algoritmos de filtrado colaborativo (*Collaborative Filtering*) permiten obtener excelentes resultados al momento de determinar las preferencias de los clientes, sin embargo, estos métodos requieren información de calificaciones u opiniones de clientes que actualmente no registra la empresa, o retroalimenta a través del procesamiento de datos a través de algún canal de comunicación (redes sociales o sitio web). Por consiguiente, para esta solución se utilizarán técnicas de filtros basados en contenido (*Content-based filtering*), donde los resultados se obtienen a partir del historial de compras de los clientes que incluyan al menos un artículo de la lista de canjeables.

1.7 Resultados esperados

La solución propuesta y sus componentes deberán configurarse en un servidor con acceso a la información de ventas de la compañía, de modo que se puedan realizar las tareas de extracción de datos y procesamiento de los modelos de Inteligencia artificial utilizados. Del mismo modo, el *dashboard* definido como capa de presentación del sistema tendrá acceso a los resultados generados por los modelos, para que puedan ser consultados por los usuarios finales del área de Gerencia Comercial.

Los resultados generados por el Sistema de Recomendación incluirán lo siguiente:

- Determinar los tipos de artículos canjeables recomendados para compra de los clientes, de acuerdo con las compras realizadas durante la campaña de artículos de canje en curso.
- Los resultados obtenidos serán registrados en una base de datos de recomendaciones para posterior uso del *dashboard* interactivo de la solución u otros sistemas.
- El *dashboard* definido como método de visualización de resultados mostrará los tipos de artículos recomendados de tipo canje.
- Finalmente, se debe detallar el proceso de carga de nuevos datos para evaluación periódica de campañas en curso o futuras.

En la industria de *retail*, uno de los problemas fundamentales es presentar al cliente el producto adecuado, en el lugar y tiempo apropiados. En el caso particular de las campañas de canjes, se busca finalmente que las recomendaciones ofrecidas promuevan la compra de este tipo de artículos, metas difíciles de alcanzar sin un plan de conocimiento y llegada a los clientes a un nivel más personalizado. Es claro, además, que el modo en que se presentan los resultados al usuario final o cliente interno de la compañía tiene la misma importancia que el método utilizado para determinarlos.

1.8 Conjunto de datos

El conjunto de datos o *dataset* comprende 291317 registros obtenidos de 161 tiendas de la empresa, en el período comprendido entre los meses de junio a octubre del 2023, correspondiente al inicio de la última campaña de artículos canjeables. Los registros pertenecen a cada una de las facturas o transacciones que tienen un artículo de canje, identificado su ID de departamento. Los demás artículos o ítems incluidos en la venta también se encuentran agrupados de acuerdo con su departamento, pero considerando además el rango de precios al que pertenece.

El *dataset* final cuenta con los siguientes atributos:

Tabla 1.1 Descripción del Dataset

Campo	Tipo de Dato	Descripción
ID de venta	Numérico	Identificador de la transacción de venta
Fecha	Fecha	Fecha de la venta
Hora	Carácter	Hora de la venta
ID de almacén	Numérico	Identificador del almacén donde se hizo la venta
ID de ciudad	Numérico	Identificador de la ciudad del almacén
Nombre de ciudad	Carácter	Nombre de la ciudad del almacén
ID de cliente	Numérico	Identificador del cliente que realizó la compra
Género	Carácter	Género del cliente
ID de almacén preferido	Numérico	Almacén donde más compras ha efectuado el cliente
ID de departamento	Numérico	Identificador del departamento del ítem vendido
ID departamento de canje	Numérico	Identificador del departamento del artículo de canje
Nombre de departamento	Carácter	Nombre del departamento del artículo vendido
Precio bajo	Numérico	Precio límite del primer cuartil para el ítem vendido
Precio medio bajo	Numérico	Precio límite del segundo cuartil para el ítem vendido
Precio medio alto	Numérico	Precio límite del tercer cuartil para el ítem vendido
Precio alto	Numérico	Precio límite del último cuartil para el ítem vendido
ID de línea de negocio	Numérico	Línea de negocio a la que pertenece el ítem vendido
Nombre de línea	Carácter	Nombre de la línea de negocio del ítem vendido
Cantidad	Numérico	Cantidad vendida de los ítems en la transacción
Monto	Numérico	Valor vendido de los ítems en la transacción
Ítem Canjeable	Carácter	Indicador si el ítem fue canjeado en la transacción

CAPÍTULO 2

2. ESTADO DEL ARTE

El objetivo de este capítulo es detallar las metodologías que se utilizarán en el desarrollo de la solución, enmarcadas dentro de un conjunto de técnicas aplicadas a los sistemas de recomendación.

2.1 Marco teórico

Una gran cantidad de las investigaciones enfocadas en el ámbito de *retail*, está constituida por la aplicación de herramientas basadas en Inteligencia Artificial y Aprendizaje Automático, que buscan aumentar la propuesta de valor para el negocio, a la vez de adaptar o mejorar los algoritmos existentes, ejecutados en el entorno de un problema real y resolviendo los inconvenientes derivados de la abstracción de un problema y definiendo las métricas adecuadas para evaluar los resultados, entre los que destacan las técnicas de modelado de grafos como insumo para los sistemas de recomendaciones.

2.1.1 Sistemas de recomendaciones

Un sistema de recomendaciones es un tipo de sistema de filtrado cuyo objetivo principal es mostrar información personalizada, mejorando la experiencia de los usuarios y buscando rentabilidad para el negocio. Estos sistemas comprenden una de las aplicaciones típicas del aprendizaje automático en el mundo real, siendo de gran utilidad en los campos de la industria y académicos [7, p. 3].

La información que estos sistemas proporcionan se vuelve una ayuda para los usuarios en sus procesos de toma de decisiones. Esto, sin embargo, plantea a la vez algunas interrogantes, como el número de opciones que deben mostrarse al usuario, de igual manera que el cómo y cuándo se presentan estas opciones. Por consiguiente, un sistema efectivo de recomendaciones requiere conocer en profundidad la idiosincrasia del negocio para ofrecer recomendaciones que generen valor al mismo y al consumidor, sin descuidar el cómo se puede medir el impacto y el éxito de tales recomendaciones, o posibles riesgos derivados de ofrecerlas [8].

2.1.2 Tipos de rating

Los *ratings* de un cliente son utilizados por ciertos modelos de recomendación para identificar sus preferencias, y pueden ser de dos tipos [9, p. 19]:

- ***Explícito:*** Por ejemplo, cuando se le solicita al cliente retroalimentación sobre su nivel de satisfacción en una compra.
- ***Implícito:*** No se interactúa con el cliente, sino que se deduce su comportamiento, por ejemplo, analizando los artículos incluidos en su compra.

Los *ratings* explícitos tienen el inconveniente de ser difíciles de obtener, sobre todo para soluciones diferentes de aquellas orientadas al comercio en línea, ya que no todos los clientes realizan comentarios o evalúan su compra, y aún en este ámbito, la mayoría de los comentarios se registran debido a experiencias negativas, por lo cual conlleva el problema de sesgos en los datos de origen, problemas por los cuales se elige analizar el comportamiento de los clientes. Por otra parte, los implícitos evalúan solamente la elección del cliente, y no el motivo que lo llevó a tomar esa decisión, por ejemplo, si la compra tiene como destino a un tercero, por tanto, no pueden diferenciar de qué usuario o cliente son las preferencias [9, p. 19]

Si bien en la literatura científica existen diferentes clasificaciones en lo que respecta a sistemas de recomendación, se pueden diferenciar dependiendo de las interacciones que analicen. En este sentido, existen dos tipos de interacciones [9, p. 16]: aquellas basadas en el usuario o cliente (*user based*), evaluando directamente sus preferencias o *ratings* explícitos o implícitos, conocidos como métodos de filtrado colaborativo (*Collaborative Filtering*), y aquellas basadas en los ítems (*item based*) que tienen en cuenta atributos de ellos de los clientes, que se denominan métodos basados en contenido (*Content-based filtering*).

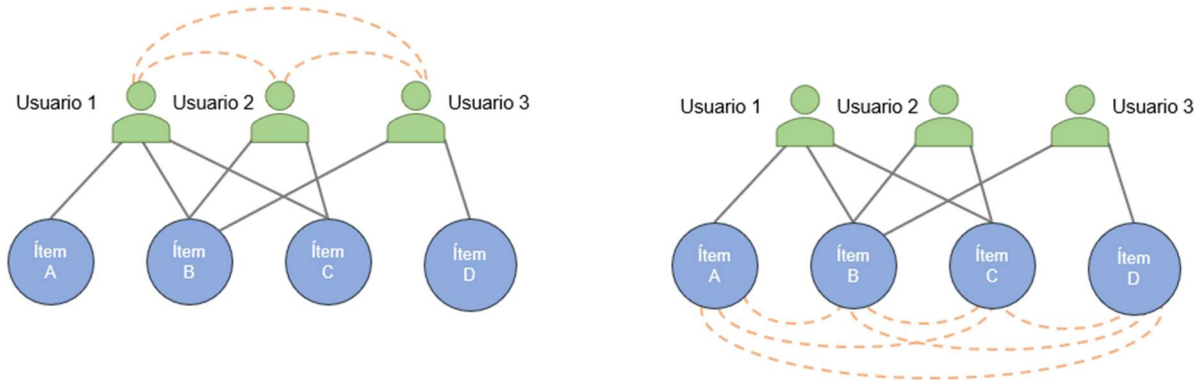


Figura 2.1. Interacciones basadas en usuarios o en ítems

2.1.3 Filtrado colaborativo (*Collaborative filtering*)

Estas técnicas tienen en cuenta las calificaciones de los clientes sobre un artículo, comparando básicamente si fue o no de su agrado con las calificaciones de otros clientes, para predecir sus preferencias [10]. Un ejemplo son las recomendaciones de Amazon, donde los usuarios registran sus reseñas y el sistema realiza sugerencias de compras basadas en el perfil del cliente. Las relaciones entre clientes y artículos de acuerdo con el historial de compra se realizan mediante la construcción de matrices de similitud considerando las calificaciones o *ratings* de los clientes [9, p. 16]. Como se mencionó anteriormente, estos *ratings* pueden ser implícitos o explícitos.

2.1.4 Filtrado basado en contenido (*Content-based filtering*)

Estos métodos tienen en cuenta las características descriptivas de los ítems. En el ámbito del *retail*, se utilizan atributos propios de los artículos (marca, color, tamaño o descripciones). El objetivo es buscar similitudes entre los ítems a través de dichos atributos. Si un cliente A adquiere un artículo B, pueden explorarse otros artículos de acuerdo con los atributos analizados, y se hace una recomendación de artículos similares o relacionados a B [9, p. 17]. Es común además el utilizar algoritmos NLP (*Natural Language Processing*) para identificar palabras clave, clasificar o agrupar en vectores matemáticos términos relacionados con los ítems.

Las técnicas de filtrado basadas en contenido no requieren de datos de otros clientes para generar recomendaciones, y al mismo tiempo, están dirigidas específicamente a sus intereses, incluyendo recomendaciones de tipos especiales de

artículos. Esto constituye una ventaja para poder determinar recomendaciones de productos canjeables con puntos, además que la información disponible en Retailers S.A. comprende solamente los datos de ventas, sin un registro de comentarios o reseñas elaboradas por los clientes.

2.1.5 Estrategias de procesamiento de datos

Los modelos de aprendizaje automático pueden ver afectadas sus resultados si los datos de origen contienen un sesgo determinado en alguno de los atributos utilizados, esto eso, que pertenezcan a un grupo mayoritariamente en comparación a otros, tanto en las variables independientes como en la variable objetivo a predecir. Dos estrategias son comúnmente aplicadas para igualar el número de observaciones de acuerdo con el atributo elegido: *upsampling*, la cual se basa en incrementar el número de instancias de la clase que constituye la minoría en el *dataset*, utilizando una muestra de los datos para replicarlos y que estos se encuentren uniformemente representados, o generando datos sintéticos, y *downsampling*, que reduce los datos a utilizar basándose en una muestra de ellos, para igualar la cantidad de datos disponibles al grupo o clase minoritaria. El primer método se utiliza generalmente para hacer énfasis en datos que representen situaciones poco frecuentes o identificación de anomalías, mientras que el segundo busca reducir el *overfitting* o sobreajuste en los resultados, lo que quiere decir es que el modelo pierda la capacidad de "aprender" patrones de los datos y más bien "memorice" los resultados, obteniendo rendimientos altos en conjuntos de entrenamiento, pero con baja exactitud en datos no vistos durante este proceso.

Por otra parte, los valores atípicos también pueden afectar la capacidad de un modelo de aprender patrones ocultos en un conjunto de datos. Para un óptimo análisis, la distribución de los datos debe acercarse a la de la curva normal, evitando estos valores en caso de existir. Un modo de eliminar *outliers* en los datos es aplicando el rango intercuartil o IQR, el cual es el intervalo entre el primer y el tercer cuartil de los datos:

$$IQR = \quad Q_3 - Q_1 \tag{2.1}$$

Los valores menores al primer cuartil menos 1.5 veces el IQR, o superiores al tercer cuartil más 1.5 veces el IQR, se consideran *outliers*.

Otro método utilizado está basado en el z-score, el cual determina qué tan alejados están los valores respecto a la media de los datos, de acuerdo con la desviación estándar. Es una práctica común considerar aquellos valores mayores o menores a 3 deviaciones estándar como *outliers*, y descartarlos del conjunto de datos.

Las técnicas de normalización de datos se aplican en tareas de aprendizaje automático para definir una sola unidad de medida para todos los atributos del modelo. Esto se obtiene restando la media de los datos y dividiéndolos para la desviación estándar, de manera que reduzca el tiempo de cálculo del algoritmo y estabilice su entrenamiento. Por otra parte, en redes neuronales se aplican técnicas conocidas como métodos de regularización, los cuales buscan reducir la complejidad y el sobreajuste del modelo y mejorar su exactitud, ya que redes neuronales menos complejas son menos propensas a sobreajustar los resultados e incrementar la variabilidad en ellos. Entre estas técnicas se cuentan la de *Ridge regularization* o *L2 regularization*, conocida además como *weight decay*, la cual penaliza valores más grandes en los pesos que la función de optimización de la red actualiza durante el entrenamiento de la red, la técnica de *dropout*, para desactivar un porcentaje de las neuronas en cada capa de la red, y la de *early stopping*, para detener el entrenamiento del modelo si no hay mejoras.

En cuanto a la cantidad de datos analizada por el modelo, la técnica de *batch processing* consiste en dividir el conjunto de datos de entrenamiento en grupos más pequeños, e introducirlos en bloques al modelo, en lugar de en un solo bloque con todos los datos. Esto conlleva una reducción del tiempo y recursos computacionales utilizados al entrenar el modelo.

2.1.6 Transformación de datos categóricos

Al trabajar con modelos de aprendizaje automático, los atributos categóricos de los conjuntos de datos deben convertirse antes a valores numéricos. Por lo general, estos datos se convierten a vectores de longitud fija, a través de técnicas como Word2Vec, Doc2Vec y Node2Vec.

Las técnicas Word2Vec y Doc2Vec son utilizadas en Procesamiento de Lenguaje Natural (NLP) con el objetivo de aprender representaciones de palabras o documentos. Ambos modelos se basan en una arquitectura de red neuronal que ejecuta tareas de aprendizaje no supervisado, sin necesidad de datos etiquetados.

Los métodos Node2Vec se utilizan principalmente para representación de grafos en un contexto de datos estructurados de este modo. Utilizan técnicas NLP similares a las de modelos Word2Vec para aprender representaciones de palabras, y de este modo capturar similitudes estructurales en problemas que van más allá de textos o palabras. Esto se consigue definiendo una noción flexible de los vecinos de un nodo de la red, de modo que se aprenden representaciones que organizan los nodos basándose en los roles de la red o las comunidades a las que ellos pertenecen [11].

2.1.7 Representación de datos en grafos

Los grafos son estructuras de datos que buscan representar conjuntos de datos abstractos a través de elementos denominados nodos o vértices (*nodes*), que representan conceptos, conectados por medio de bordes o aristas (*edges*) que representan las relaciones entre nodos, y que pueden significar causas, conexiones o precedencia en el tiempo u orden. Esta arquitectura permite representar un gran número de problemas complejos, no solo en estudios científicos sino en la industria también, especialmente aquellos con información disponible en estructuras no estáticas o con un gran volumen.

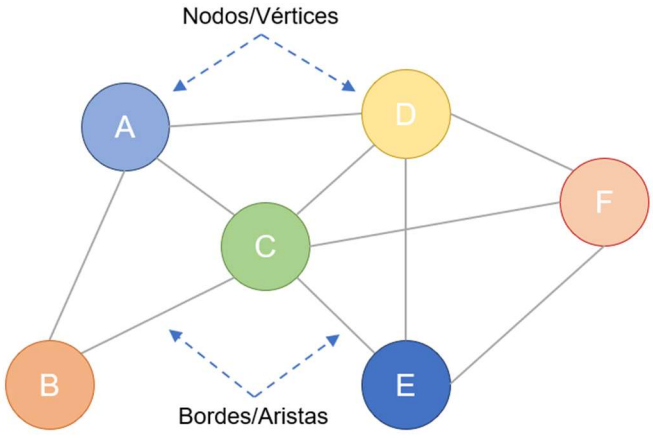
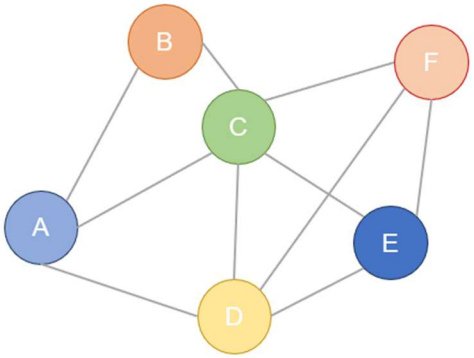


Figura 2.2. Representación de datos en un grafo

Los bordes de los grafos pueden indicar cómo se orientan las relaciones de nodo a nodo, las cuales pueden ir en una sola dirección o en ambos sentidos. Estas relaciones se definen a través de una matriz de adyacencia, cuyos elementos indican los pares de nodos conectados en el grafo. Las relaciones pueden variar en importancia, a través de la asignación de pesos ponderados a cada una de ellas.



	A	B	C	D	E	F
A	0	1	1	1	0	0
B	1	0	1	0	0	0
C	1	1	0	1	1	1
D	1	0	1	0	1	1
E	0	0	1	1	0	1
F	0	0	1	1	1	0

Figura 2.3. Representación de un grafo y su correspondiente matriz de adyacencia

. En cuanto a los nodos, si estos son de un solo tipo o no distinguibles entre ellos de acuerdo con sus atributos, constituyen un grafo homogéneo, mientras que, si son de diferentes tipos, por ejemplo, con atributos que representan a clientes y a los artículos que ellos compran, comprenden grafos heterogéneos. Finalmente, entre las diferentes clases de grafos que son de particular importancia están los grafos bipartitos, los cuales están compuestos de un grafo heterogéneo que comprende dos conjuntos diferentes de nodos, donde los bordes se conectan únicamente entre nodos de conjuntos opuestos. De manera similar, tres conjuntos de nodos dan lugar a los grafos tripartitos, y varios conjuntos, a grafos multipartitos.

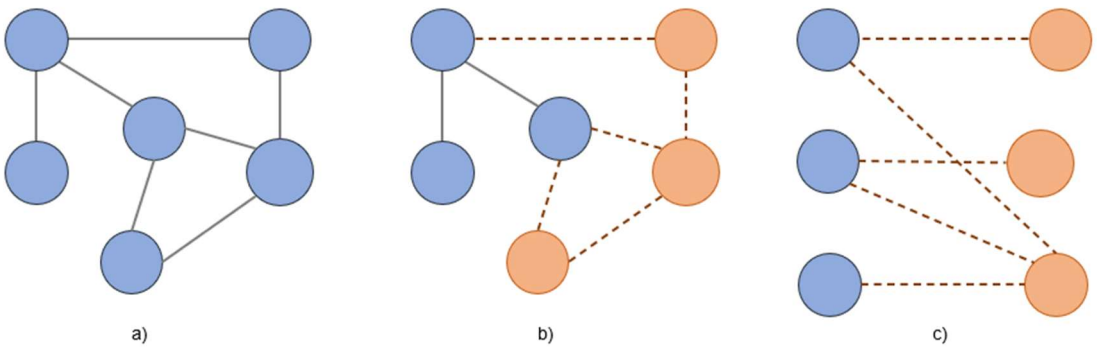


Figura 2.4. Grafo homogéneo (a), heterogéneo (b) y bipartito (c)

2.1.8 Redes neuronales

Los algoritmos de redes neuronales, también conocidas como redes neuronales artificiales, constituyen una de las técnicas más avanzadas de aprendizaje automático, donde los sistemas aprenden a realizar ciertas tareas analizando múltiples ejemplos de entrenamiento, de los cuales se extraen diferentes características, que la red aprende a reconocer para determinar patrones. De manera similar a un cerebro humano, una red neuronal está compuesta de miles o millones de neuronas, o nodos interconectados en diferentes niveles o capas, que analizan los datos ingresados y generan una salida para los nodos de la siguiente capa, ajustando los resultados obtenidos de acuerdo con los errores encontrados.

2.1.9 Redes neuronales basadas en grafos

Si bien la estructura de los grafos permite una representación adecuada de la información, el conocimiento implícito en los grafos es difícil de capturar utilizando modelos de aprendizaje automático tradicionales, como las redes neuronales convolucionales (*Convolutional Neural Networks*, CNN) aplicadas en análisis de imágenes, o las redes neuronales recurrentes (*Recurrent Neural Networks*, RNN), utilizadas en análisis de secuencias, ya que los grafos carecen de la estructura de matriz de las primeras, o el orden lineal de las segundas. Combinando el concepto de grafos con el de redes neuronales, se obtienen las denominadas redes neuronales basadas en grafos o GNN (*Graph Neural Networks*), las cuales se han convertido en uno de los métodos de vanguardia para sistemas de recomendaciones en muchos aspectos, incluyendo diferentes escenarios, objetivos y aplicaciones [7, p. 15].

Dependiendo de la arquitectura utilizada, la red neuronal se compone de una o varias capas intermedias (*hidden layers*) seguidas por una función de activación (por ejemplo, ReLU, Leaky ReLU), con una operación de *pooling*, por ejemplo, *Global average pooling*, la cual se encarga de reemplazar las capas interconectadas de una CNN clásica, para generar un mapa de características para cada categoría correspondiente de la tarea de clasificación, tomando el promedio de cada mapa de características, utilizando el vector resultante cono insumo para una capa softmax [12] que hace las veces de clasificador, que permita determinar las probabilidades de pertenencia del resultado a cada clase, como se muestra en la figura 2.5.

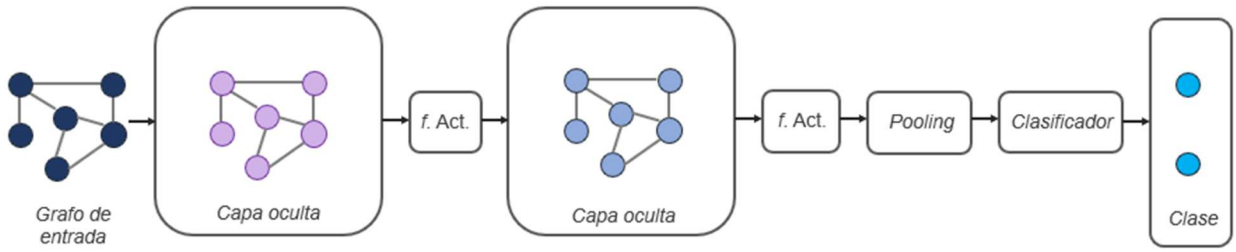


Figura 2.5. Arquitectura básica de una red neuronal de grafos para una tarea de clasificación

El objetivo de las GNNs es aprender una mejor representación de los nodos usando para ello información de su entorno. Básicamente, una red de este tipo trabaja agregando información de las características obtenidas de nodos vecinos o de los bordes adyacentes iterativamente, para actualizar la representación actual de cada nodo, a través de cada capa que compone la red. Esta actualización se conoce como intercambio de mensajes (*message passing*), utilizando una función de agregación, y, dependiendo del algoritmo utilizado, puede limitarse a una muestra o número determinado de nodos vecinos.

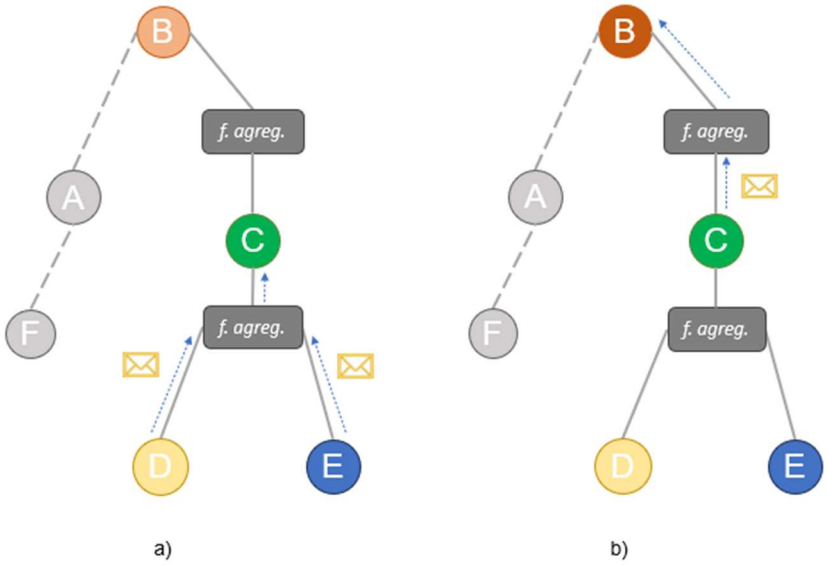


Figura 2.6. *Message passing* para actualizar la representación del nodo C en la primera capa (a), y para el nodo B en la segunda (b)

Una ventaja importante de las redes de este tipo frente a modelos tradicionales de aprendizaje automático está en que, mientras los modelos basados en árboles como

Random Forest y XGBoost necesitan un número fijo de atributos, las GNNs tienen la capacidad de manejar la información en grafos de tamaños y estructuras diversas, lo cual no sería posible con datos tabulares sin un extenso preprocesamiento o ingeniería de atributos (*feature engineering*), lo cual, en muchos casos, implica codificar múltiples variables en grupos de bits (*one-hot encoding*), generando problemas de información relevante muy espaciada a través de todo el conjunto de datos (*sparsity*), afectando la capacidad de aprendizaje del modelo.

En la industria del *retail*, los sistemas de recomendaciones pueden construirse a través de tareas de clasificación de nodos (*node classification*), predicción de enlaces (*link prediction*) o clasificación de grafos (*graph classification*). Existen múltiples algoritmos basados en GNNs, los cuales básicamente difieren en el método utilizado para aprender las representaciones de los nodos, encontrando los patrones o dependencias dentro de los grafos que buscan analizar, aplicando un enfoque de aprendizaje inductivo (*inductive learning*) o transductivo (*transductive learning*).

2.1.10 Aprendizaje inductivo y transductivo

Un algoritmo utiliza un enfoque inductivo cuando a partir de los datos etiquetados aprende a predecir datos que no han sido observados previamente, es decir, siguiendo la premisa de aprendizaje no supervisado. El modelo aprende a generalizar durante el entrenamiento, sin necesidad de observar todo el conjunto de datos.

Por el contrario, en los métodos transductivos, mientras se aprende de los datos etiquetados en el entrenamiento, se trata de predecir las etiquetas del conjunto de datos de prueba al mismo tiempo. El modelo usa la información de los datos de nodos vecinos para mejorar las predicciones. En otras palabras, ambos conjuntos de datos, de entrenamiento y de pruebas, se observan durante el entrenamiento.

Los modelos transductivos tienen la ventaja de alcanzar una precisión más alta en las predicciones, a costa de perder capacidad de generalización. Los algoritmos inductivos, son más adecuados si los datos sobre los que se harán las predicciones cambian constantemente o nuevos datos con diferentes características surgen frecuentemente.

2.1.11 Análisis RFM

Una de las técnicas más utilizadas para segmentar y clasificar clientes de acuerdo con su importancia o fidelidad hacia la marca de una empresa es el análisis RFM, el cual evalúa las compras o servicios adquiridos conforme a su recencia (*Recency*), frecuencia (*Frequency*) y valor monetario (*Monetary value*). Básicamente, se establecen diferentes categorías de clientes según el tiempo transcurrido desde su última compra, cuántas compras han realizado durante un período determinado y el monto que representan, con el objetivo de comprender sus preferencias.

Múltiples variantes del análisis RFM se utilizan en el ámbito del *retail*, las cuales utilizan parámetros adicionales para aumentar la capacidad del análisis en la identificación de características de los clientes. Por ejemplo, el modelo LRFM considera además la duración de la relación del cliente con la compañía, donde L es el tiempo transcurrido entre la primera y última compra (*Length*) [13], mientras que el modelo GRFM identifica grupos (*Groups*) de clientes leales de acuerdo a los patrones de sus compras y las características de los bienes adquiridos [14], y el modelo RFM-V agrega la variedad de los productos comprados como una cuarta variable (*Variety*), el cual es recomendado en cadenas de *retail* que cuentan con tarjetas de membresía o fidelidad con muchas cestas (*baskets*) e información de compra [15].

La clasificación de los clientes de acuerdo a los valores de cada variable puede realizarse de diferentes maneras, entre las que se cuentan técnicas de *clustering* y algoritmos *K-means*, o determinando rangos fijos dependiendo de la naturaleza del negocio, o a través del cálculo de un score para cada una de las variables, de manera que los clientes con una puntuación más alta conformen los grupos de mayor interés para el negocio, para los cuales se busca diseñar estrategias de mercadeo o soluciones personalizadas, obteniendo una mayor eficiencia en los recursos invertidos, a la vez de facilitar el análisis y evaluación de resultados de campañas específicas.

2.2 Fundamentos del problema

Considerando que existen tantos clientes como diversidad en sus intenciones de compra, es difícil predecir su comportamiento. Existen diferentes criterios para identificar

los hábitos de consumo de clientes. Algunos tienen en cuenta a características intrínsecas a la persona, como la edad, género, grupo demográfico o lugar de compra, sin embargo, en el ámbito del *retail,* las técnicas de *Basket Recommendation* (BR) buscan identificar múltiples ítems como un todo, dentro de lo que se denomina un *basket*, donde los artículos que lo componen tienen relaciones significativas entre ellos (complementarias o consistentes) [16]. Además, en los últimos años se han propuesto mejoras para esta técnica, que utilizan redes neuronales basadas en grafos, y que aprenden la intención u objetivo para el que se están incorporando los artículos al *basket*, o los patrones de múltiples intenciones de compra como ayuda para encontrar relaciones semánticas complejas de artículos [17], entre otras.

Las estrategias descritas anteriormente, a través de la aplicación de modelos de Inteligencia Artificial y Aprendizaje Profundo o *Deep Learning*, permitirán identificar los tipos de artículos canjeables que se deben ofrecer a los clientes de un modo objetivo y basado en criterios técnicos, según sus compras durante una campaña promocional. El modelo de red neuronal a utilizarse considerará las características de la compra y las relaciones entre los tipos artículos no canjeables como base para generar la recomendación, a diferencia de la promoción estática actual que se realiza a través de los diferentes canales de comunicación con el cliente.

2.3 Soluciones relacionadas de analítica y aprendizaje automático

La evolución de los sistemas de recomendaciones se puede clasificar en los siguientes grupos: Modelos analíticos basados en minería de datos (*data mining*), modelos basados en factorización de matrices, modelos neuronales y modelos de redes neuronales basadas en grafos. Sobre estos últimos, los artículos pueden diferenciarse a través de sus atributos, como precios o categorías, permitiendo inclusive la agrupación basada en estos atributos para la construcción de grafos heterogéneos.

Los sistemas de recomendaciones tienen diversas aplicaciones, entre las que destacan las orientadas a comercio electrónico o *E-commerce*, donde se busca aumentar el valor de negocio a través de diferenciar las acciones de los usuarios, como el agregar un artículo al carrito de compras en contraste con el acto de la compra en sí. Otras aplicaciones incluyen las llamadas recomendaciones de Punto de Interés (*POI,*

Point-of-Interest Recommendation), como las realizadas por Yelp, recomendaciones de noticias, o de películas, como Netflix [18], *delivery* o restaurantes, como Uber Eats [19], o música, empleos, entre otros.

A continuación, se detallan las características de algunos modelos utilizados en motores de recomendaciones desarrollados para la industria del *retail*.

2.3.1 Análisis *Market Basket* con algoritmo Apriori

El análisis de cesta de compra o *Market Basket* es una técnica de *data mining* que busca aumentar las ventas a través de la comprensión del comportamiento o historial de compra de los clientes. Se basa en buscar combinaciones de ítems que frecuentemente aparecen juntos en una venta [20], es decir, vinculando la compra de uno o más productos con otros (*itemsets*) [9, p. 23]. Por ejemplo: "Si el cliente compra jugo de naranja y frutas, ¿cuál es la probabilidad que compre pan?". Esta posibilidad es los que se conoce como Regla de asociación, y se mide a través de tres variables fundamentales:

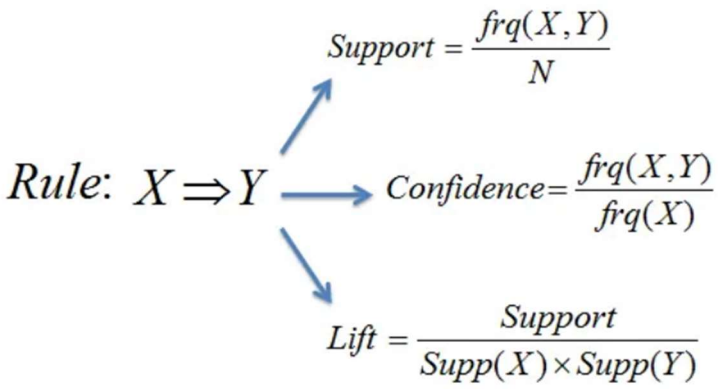


Figura 2.7. Variables para reglas de asociación en análisis *Market Basket* [20]

- ***Support:*** Es la proporción de transacciones que contienen el *itemset*.
- ***Confidence:*** Es la probabilidad de compra de un ítem posterior dado un ítem anterior.
- ***Lift:*** Es cuán frecuente la condición se cumple realmente, comparado a la estimación de que ocurra.

El algoritmo Apriori se basa en encontrar *itemsets* que se presentan con frecuencia en las transacciones, de acuerdo con un valor límite definido para la variable *support*. Luego, calcula la variable *confidence* de todas las reglas de asociación posibles. De este modo, se inicia con *itemsets* de un solo artículo, añadiendo un ítem a la vez, y descartando aquellos *itemsets* que no cumplen los umbrales definidos.

2.3.2 Descomposición en valores singulares

El método SVD (*Single Value Decomposition*) es una técnica matemática utilizada para descomponer una matriz en sus elementos [21]. A través de la factorización de matrices, o reducción de la dimensionalidad, se extraen los factores latentes para representar claramente las relaciones entre clientes y artículos.

La reducción de la dimensionalidad se consigue reduciendo el número de valores singulares a los primeros *k* valores, con lo que se obtiene una aproximación de la matriz original [9, p. 26]. Esto permite escalar eficientemente los datos, incrementando además la rapidez en los cálculos al utilizar matrices más pequeñas.

2.3.3 Redes convolucionales de grafos (GCN)

Las redes convolucionales de grafos (*Graph Convolutional Networks*) extienden el concepto de una GNN al definir una aproximación localizada de primer orden para operaciones convolucionales sobre los grafos en las diferentes capas definidas en la red, con el objetivo de propagar la información entre los nodos utilizando funciones de activación no lineales después de cada capa [22]. Están basadas en las redes neuronales convolucionales utilizadas para procesamiento de imágenes, pero llevadas al ámbito de los grafos.

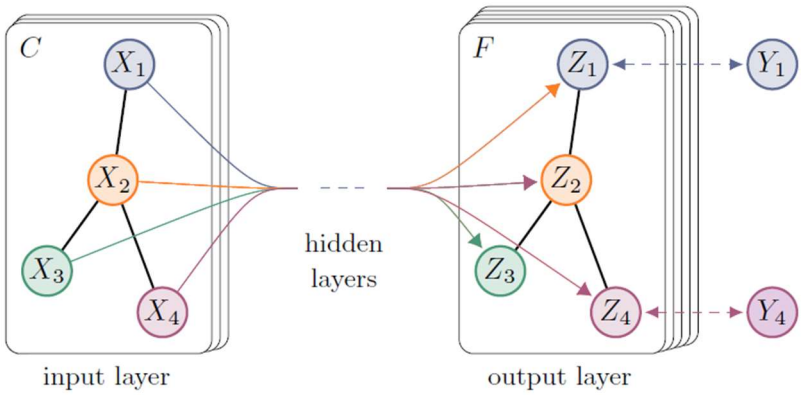


Figura 2.8. Representación esquemática de una GCN multicapa [22]

El objetivo de una GCN es generar representaciones de los nodos basándose en la información de los nodos y bordes vecinos. Como resultado, cada nodo define un grafo calculado a partir de su entorno, promediando los mensajes recibidos. Si bien estas redes admiten características no solo para los nodos sino también para los bordes, en estos últimos solo se permiten valores de una sola dimensión. Esta arquitectura es transductiva; es decir, la estructura de los grafos debería conocerse de antemano en el tiempo de entrenamiento. Por ello es útil en tareas de aprendizaje semi-supervisado, como clasificación de nodos, sin embargo, también se utiliza en aprendizaje supervisado, como en predicción de enlaces y clasificación de grafos.

2.3.4 Redes neuronales de grafos de orden más alto (*k*-GNN)

La aplicación de normalización de vecinos (*neighborhood normalization*) que realizan las GNNs por defecto, si bien contribuye a reducir la inestabilidad numérica generada por las diferencias en magnitudes de relaciones entre nodos (es decir, nodos con grados de relaciones mucho mayores que otros), trae consigo la reducción de la expresividad de la red en distinguir determinadas estructuras en los grafos. Es posible eliminar este tipo de normalización por completo, teniendo en cuenta estructuras de grafos de órdenes más altos en múltiples escalas, realizando las tareas de *message passing* directamente entre estructuras de subgrafos compuestas por *k* nodos, en lugar de nodos individuales [23]. Esta arquitectura se conoce como *k*-GNN, la cual mejora el rendimiento de las GNNs en tareas de clasificación y regresión de grafos.

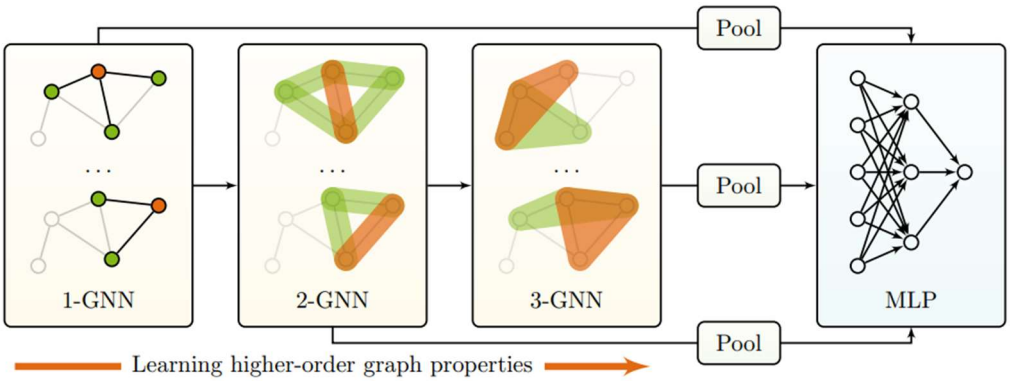


Figura 2.9. Arquitectura de red 1-2-3-GNN [23]

En la figura 2.9, para cada subgrafo de *k* nodos, se aprende una determinada característica, la cual se inicializa con las características aprendidas de todos los subgrafos de (*k*-1)-elementos.

2.3.5 Redes convolucionales en grafos con filtrado rápido espectral localizado

Este tipo de arquitectura convolucional utiliza polinomiales Chebyshev en lugar de polinomiales generales, con el objetivo de resolver las limitaciones de las capas convolucionales al momento de capturar estructuras localizadas de grafos a través de filtros eficientes, tanto en el aprendizaje como en la evaluación, manteniendo una baja complejidad computacional [24].

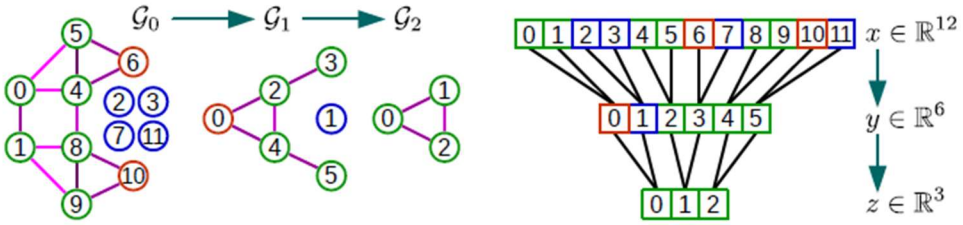


Figura 2.10. Ejemplo de agrupación y reorganización de nodos para redes basadas en polinomiales Chebyshev [24]

Esta técnica requiere obtener entornos de nodos significativos en los grafos para las tareas de *pooling* o agregación, donde los nodos similares se agrupen. Después de este proceso, el algoritmo busca crear un árbol binario balanceado y reorganizar los nodos, de modo que cada nodo tenga dos hijos, excepto en el nivel más bajo. Esta

configuración permite que las operaciones se ejecuten eficientemente, aprovechando inclusive los beneficios de procesamiento paralelo en GPUs.

2.3.6 Redes de atención de grafos (GAT)

A diferencia de las GCNs, donde cada vecino de un nodo tiene la misma importancia, en las redes de atención de grafos (*Graph Attention Networks*) se consideran a ciertos nodos más esenciales que otros, a través de un mecanismo de atención que asigna a un factor de ponderación para cada conexión entre nodos. Mediante el apilamiento de capas en las cuales los nodos pueden atender las características de sus vecinos, es posible especificar diferentes pesos a diferentes nodos sin requerir operaciones computacionalmente costosas de matrices o conocer la estructura del grafo de antemano [25].

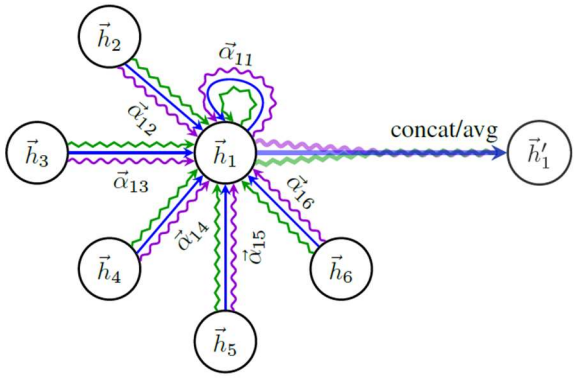


Figura 2.11. *Multi-head attention* en GATs [25]

En las GATs se utiliza el mecanismo de atención en reemplazo de la operación estáticamente normalizada de convolución. Este paso, junto a la ejecución de la función de activación y la normalización a través de una función softmax, puede replicarse más de una vez, correspondiendo cada repetición a lo que se conoce como *multi-head attention* (como muestran las flechas de la Figura 2.11), con el objetivo de promediar o concatenar los resultados.

2.3.7 Otros modelos basados en GNNs

Existen sistemas de recomendaciones en *retail* que utilizan arquitecturas como NCF o *Neural Collaborative Filtering*, la cual fue desarrollada con el objetivo de resolver

el problema de establecer los ratings implícitos entre clientes y artículos [26]; Triple2Vec, la cual se enfoca en representar en vectores los bordes de los grafos [27], o el modelo *Light Graph Convolution Network* (LightGCN), que busca resolver problemas de implementaciones basadas en redes convolucionales basadas en grafos que tienen baja contribución, o incluso degradan, el rendimiento del filtrado colaborativo, como son la transformación de características (*feature transformation*) y activación a través de funciones no lineales, simplificando el diseño de la red convolucional, volviéndola más concisa y con mejores resultados en la recomendación. Este modelo incluye solamente el componente más esencial de la red convolucional basada en grafos, que es la agregación en base a vecinos (*neighborhood aggregation*), para filtrado colaborativo [28].

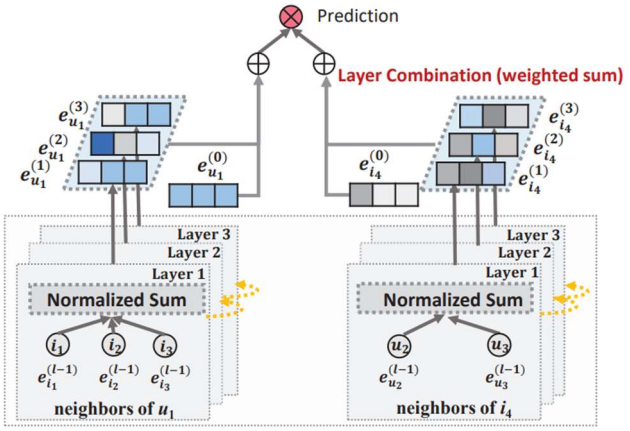


Figura 2.12. Arquitectura del modelo LightGCN [28]

Otros métodos, como el de Agregación de Interacciones Heterogéneas, parten de la premisa que las recomendaciones dentro de la cesta de compra deben relacionarse con la intención u objetivo del usuario para dicha cesta. Definiendo una entidad "cesta", para representar esa intención, la recomendación se puede modelar como una tarea de predicción de enlace entre cesta y artículo en el grafo Usuario-cesta-ítem o UBI (*User-Basket-Item graph*). Denominado como BasConv, este modelo incluye tres tipos de agregadores para tres tipos de nodos, que aprenden las representaciones tanto de su entorno como de un contexto de orden superior, utilizando un esquema de grafo tripartido entre usuario/cliente, cesta y artículo [29].

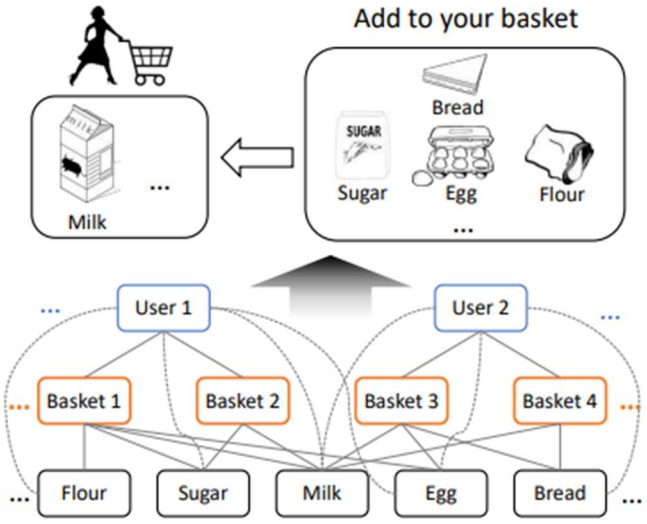


Figura 2.13. Recomendaciones de BasConv y grafo UBI de interacciones [29]

2.4 Métricas de evaluación

El objetivo más importante de un sistema de recomendación es, desde luego, evaluar las predicciones realizadas para todas las clases existentes a través de la exactitud (*accuracy*) del modelo en general, la cual está definida como:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{2.2}$$

Es decir, la proporción que existe entre las predicciones correctas y el total de predicciones. Esta métrica mide el número de veces que se ha previsto correctamente cualquier clase.

Otras medidas utilizadas en los modelos a evaluar y en el desarrollo de la solución serán la precisión (*precision*), que es la proporción de predicciones correctas en las predicciones totales de la clase positiva:

$$Precision = \frac{TP}{TP + FP} \tag{2.3}$$

Igualmente, la sensibilidad o *recall*, definida como la proporción de predicciones correctas en una clase positiva.

$$Recall = \frac{TP}{TP + FN} \tag{2.4}$$

También el *F1 Score*, que representa la media armónica de precisión y sensibilidad, y se usa como medio de balance entre las métricas que la componen.

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{2.5}$$

2.4.1 Métricas por clases

Los diferentes tipos de artículos de canje que sean recomendados por la solución propuesta pueden constituir diferentes clases objetivo, de tal manera que las métricas anteriores pueden aplicarse individualmente a cada clase, tomando en consideración los datos de la matriz de confusión generada, considerando cada clase por separado como la clase “positiva”, y el resto como “negativas”. Esto permitirá determinar cuáles son las clases que los modelos pueden predecir con mejores resultados. También es posible promediar las métricas utilizando un promedio ponderado, el cual considera el equilibrio entre las clases, de acuerdo con su representación en el *dataset*.

2.4.2 ROC y AUC

La curva ROC (*Receiver Operating Characteristic curve*) es un gráfico que presenta el rendimiento de un modelo de clasificación en todos sus umbrales de decisión. Esta curva se grafica considerando los valores de tasa de falsos positivos en el eje X y la tasa de verdaderos positivos (que es igual al *recall*) en el eje Y. Entre más cercano esté el arco de la curva hacia arriba y a la izquierda, mejor es el rendimiento del modelo. Por otra parte, AUC o *Area under the (ROC) Curve*, mide toda el área que se encuentra por debajo de la curva ROC, con valores entre 0 y 1. Entre más cercano a este último se encuentre el valor del AUC, el modelo hará mejores predicciones.

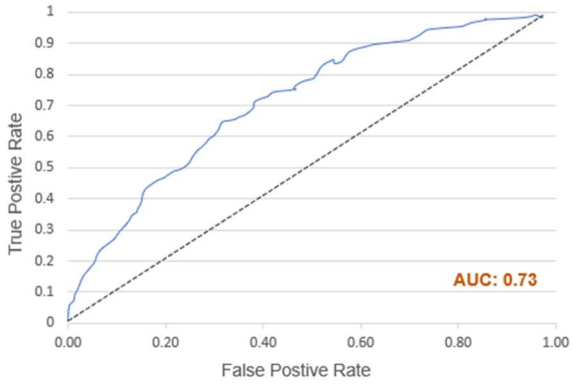


Figura 2.14. Ejemplo de una curva ROC y su correspondiente AUC

De modo similar a las métricas descritas previamente, al tratarse con predicciones de múltiples clases como salidas del modelo, se pueden determinar los valores de ROC y AUC para cada clase objetivo.

2.4.3 Tasa de conversión

En los sistemas de recomendación, una métrica de importancia corresponde a la tasa de conversión o *Conversion rate*, la cual puede medir directamente el valor para el negocio generado a partir de las recomendaciones [30]. Esta tasa permitirá identificar en qué proporción las recomendaciones que genera el sistema se traducen en ventas de canjes según las sugerencias a los clientes. En el contexto de los artículos de canje ofrecidos, la fórmula a utilizar estará definida del modo siguiente:

$$Conversion\ rate = \frac{Número\ de\ Canjes\ correctamente\ predichos}{Total\ de\ recomendaciones} \tag{2.6}$$

2.5 Limitaciones de los modelos

Los modelos descritos utilizan diferentes técnicas de analítica o inteligencia artificial para obtener recomendaciones; si bien estos son utilizados en *retail*, tienen limitaciones inherentes a su arquitectura base, la demanda de recursos tecnológicos o al cumplimiento de los objetivos de la empresa, entre las cuales se detallan:

- Por lo general, las soluciones de mayor impacto en demanda de recursos de procesamiento son aquellas basadas en redes neuronales, si bien presentan

ventajas al momento de interpretar las relaciones entre artículos dentro de una transacción modelada como grafo. Las soluciones de analítica pueden ofrecer una solución menos compleja que las de redes neuronales, sin embargo, estas últimas proporcionan un mayor nivel de aprendizaje de patrones de compra de los clientes, lo que se traduce en mejores recomendaciones.

- Aun cuando se pueden utilizar en escenarios de aprendizaje inductivo, los modelos GCN o LightGCN son inherentemente transductivos; los mejores resultados se obtienen al conocer la estructura completa de los grafos durante el entrenamiento, por lo cual se pierde la capacidad de generalizar para nuevos datos con diferentes estructuras.
- Las redes GAT requieren un procesamiento intensivo y extenso para entrenamiento, con mejores resultados que las redes GCN, aun cuando tienen cierta tendencia al sobreajuste a los datos debido al gran número de parámetros y a la falta de supervisión específica al ponderar los mecanismos de atención.
- Los modelos Chebyshev requieren una elección apropiada del número de polinomiales, al igual que el número de *attention heads* en las GATs; valores inadecuados afectan significativamente el rendimiento del modelo.
- Configuraciones más complejas o de mayor profundidad en las redes no siempre van de la mano con un mayor nivel de precisión en los resultados entregados, por el contrario, pueden producir sesgos en los resultados, o sobreajustes.

Todos los modelos detallados dentro del estado del arte pueden ofrecer recomendaciones de artículos o análisis de cestas de compra, sin embargo, la utilización con los datos de la compañía, con las características de las transacciones y sus tipos de artículos de canje es algo que deberá ser puesto a prueba, además que las recomendaciones de artículos que se busca obtener no corresponden a artículos comunes a incluir en una venta, sino resultados en función de los artículos canjeables.

2.6 Software y recursos de desarrollo

El desarrollo de la solución será realizado bajo el lenguaje de programación, Python, considerando un ambiente local para los notebooks de modelamiento en grafos

y bajo el entorno de programación Google Colab para el entrenamiento de los modelos, por cuanto estas tareas demandan recursos más potentes de servidores, tanto a nivel de CPU y RAM para las pruebas necesarias, y especialmente por la disponibilidad y ventaja de ofrecer procesamiento con GPU, lo cual es una necesidad al trabajar con modelos de redes neuronales.

Entre las librerías de Python más importantes a utilizar se incluyen:

- ***Scikit-Learn***, la cual está especializada en modelos de aprendizaje automático.
- ***Pandas***, la cual permite transformación y exploración de datos.
- ***NumPy***, para manipulación de datos y funciones matemáticas.
- ***PyTorch,*** la cual se especializa en la implementación de algoritmos de *Deep Learning*.
- ***Pytorch Geometric***, o PyG, basada en PyTorch, permite el diseño y programación de Redes Neuronales basadas en Grafos.

Los resultados a obtener de los modelos se almacenarán en una base de datos residente en un servidor de la empresa, para consumo de las diferentes áreas del negocio que la requieran. Inicialmente, la base de datos sugerida corresponde al RDBMS MSSQL Server en versión Express; con monitoreo del crecimiento mensual para determinar políticas de borrado y pase a históricos o si amerita un *upgrade* de versión, con el respectivo costo de licencias.

2.7 Visualización de resultados

Los resultados entregados por la solución permitirán establecer las recomendaciones de productos canjeables según las ventas de los usuarios. Esta información también podrá consultarse a través de un *dashboard* interactivo basado en Microsoft PowerBI, ya que actualmente la compañía ya usa esta plataforma a nivel del área de Analítica, y por tanto es familiar tanto para los usuarios de esta área como para los de alto nivel que tienen acceso a los reportes gerenciales.

CAPÍTULO 3

3. DISEÑO E IMPLEMENTACIÓN

La implementación de la solución propuesta requiere varias etapas que inician con el tratamiento de los datos originados desde el sistema de Punto de venta, luego con el diseño de los algoritmos de Aprendizaje Automático que extraerán los patrones y características de estos datos para determinar las recomendaciones de tipos de artículos canjeables a partir de todos los demás tipos de artículos adquiridos en una transacción, y finalmente, el detalle de la plataforma requerida para el funcionamiento y visualización de resultados, junto con los indicadores y métricas que se usarán para evaluar los resultados de los modelos. Este capítulo tiene como objetivo la presentación de estos aspectos.

3.1 Exploración y validación de datos

Los datos obtenidos para establecer las recomendaciones comprenden la información de las ventas que incluyan artículos canjeables, especificando la información intrínseca de cantidad, volumen y del tipo de artículo o departamento al que pertenece el artículo vendido, así como del cliente, en lo referente a su género y la tienda con mayor cantidad de compras realizadas durante el período.

3.1.1 Datos de ventas

Como se mencionó anteriormente, Los datos originales de ventas se obtuvieron a nivel nacional de 161 tiendas de la empresa, en el período comprendido entre los meses de junio a octubre del 2023, correspondiente al inicio de la última campaña de artículos canjeables. El total de ventas con al menos un artículo de canje comprende 446345 registros, pero debido a que en una transacción pueden existir diferentes SKUs (*stock keeping unit*, o códigos de artículos) para artículos altamente similares, pero que varían en atributos como la presentación o tamaño, color, medida, diseño o marca, se agruparon dichos códigos por el departamento al que pertenecen, directamente a través de vistas de base de datos Si bien los atributos mencionados constituyen información muy útil para establecer características únicas del artículo que enriquecen el análisis, en el sistema de ventas de la empresa no se existe información sobre ellos en la mayoría de registros.

Tabla 3.1 Estadísticas de atributos secundarios de artículos

Atributo	Valor
SKUs distintos vendidos	29957
Porcentaje de artículos con datos de color válidos	2.59%
Porcentaje de artículos con datos de presentación/tamaños válidos	5.53%
Porcentaje de artículos con datos de unidad de medida válidos	5.78%
Porcentaje de artículos con datos de marca válidos	6.60%

La falta de información en estos atributos implica que no se puedan utilizar como insumo para el análisis, sin embargo, con el objetivo de no perder expresividad en artículos pertenecientes al mismo departamento, las vistas de base de datos incluyen características de niveles de precio, dependiendo del cuartil al que pertenece el precio unitario del SKU. Así los artículos se agrupan en los rangos de precio bajo (desde el mínimo precio al primer cuartil de los precios para el departamento), precio medio-bajo (hasta el segundo cuartil), medio-alto (hasta el tercer cuartil) y alto (hasta el máximo precio). Para estas agrupaciones se consideró el total de unidades vendidas y su precio total.

A través de vistas de base de datos se excluyeron aquellas ventas que solo tenían artículos canjeables, ya que se necesita tanto de este tipo de artículos como de aquellos de venta normal para determinar las recomendaciones de los primeros de acuerdo con los segundos presentes en una transacción. Para definir el ámbito del análisis, se tomaron en cuenta además las ventas con un solo artículo de canje. De igual modo, no se consideran los registros de artículos que hayan sido devueltos, para analizar solamente las ventas netas y efectivas.

Para efectos de las recomendaciones, se consideran como una familia y departamento diferente a aquellos que incluyan los artículos canjeables, es decir, si un artículo es canjeable, para aquellas transacciones donde se haya adquirido de esta forma representará un departamento y familia diferente a la original.

Las ventas de artículos canjeables están orientadas solamente a clientes fidelizados de la empresa, de los cuales se consideran como atributos adicionales el género y la tienda en la que han realizado el mayor número de compras.

Finalmente, el conjunto de datos con la información agrupada por departamentos y niveles de precios comprende 291317 registros en total.

3.1.2 Exploración

Para este análisis se explorarán las características de clientes y ventas que permitan definir los algoritmos a utilizar. A continuación, se listan estadísticas básicas del conjunto de datos:

Tabla 3.2 Estadísticas del conjunto de datos

Característica	Valor
Total de registros	291317
Cantidad de transacciones/ventas	37914
Cantidad de clientes únicos	32365
Cantidad de departamentos canjeables	22
Cantidad de departamentos normales	281
Número de Tiendas con información para análisis	150

Las cantidades obtenidas en los tipos de artículos normales y canjeables únicos que existen (281 y 22, respectivamente), permitirán que las recomendaciones se basen en sugerir departamentos de artículos, y no sobre artículos individuales en sí, por lo que las relaciones entre ventas y artículos para el modelamiento se definirán de esta forma. El número de tiendas se redujo al considerar solamente las transacciones con artículos de canje y normales combinados, y aquellas con más de un artículo canjeado.

Las campañas de canjes que realiza la compañía son evaluadas semana a semana, considerando especialmente el número de ventas de canjes por día, como se observa en la figura 3.1, que contiene un subconjunto de los datos originales.

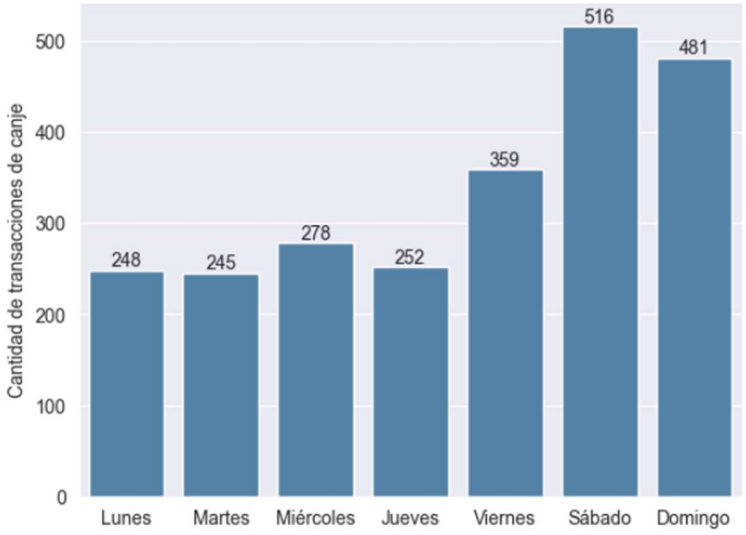


Figura 3.1. Ventas de artículos canjeables por día de la semana

La muestra obtenida corresponde a la última semana del conjunto de datos. Se observa que la mayor cantidad de canjes se realizan los sábados y domingos, en ese orden (516 y 481, respectivamente), con una reducción del 48% en los lunes y martes, con un incremento desde el miércoles (4992) hasta el viernes (6243).

El conjunto de datos incluye también información del género del cliente. Se puede comprobar si los valores presentes de esta característica se encuentran uniformemente distribuidos entre las ventas de artículos canjeables.

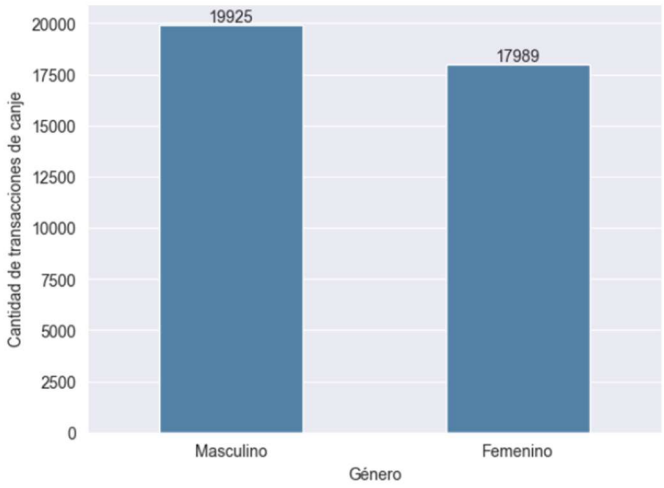


Figura 3.2. Transacciones por género del cliente

En la figura 3.2 se puede observar que, del total de canjes en todo el conjunto de datos, 19925, es decir, el 52.55% es realizada por clientes de género masculino, mientras que el 47.45% corresponde al género femenino. La diferencia es de solo el 5.10%, por lo que se considera que no existe un desbalance significativo.

3.1.3 Balanceo de datos y eliminación de *outliers*

Es importante que los datos obtenidos no se encuentren desbalanceados, siendo este un problema muy frecuente en *datasets* reales. Si bien las redes neuronales son un método muy utilizado en clasificar datos no balanceados, a menudo esto afecta negativamente al modelo [31]. La mayoría de las tiendas de la empresa se encuentran en la ciudad de Guayaquil, por consiguiente, se debe verificar si efectivamente, en esta ciudad se realiza el mayor número de canjes. Debido a que no necesariamente los hábitos de consumo de un cliente de Guayaquil son iguales a los de clientes de otras ciudades, las recomendaciones que se obtengan para estos últimos no deben estar sesgadas con datos de los primeros.

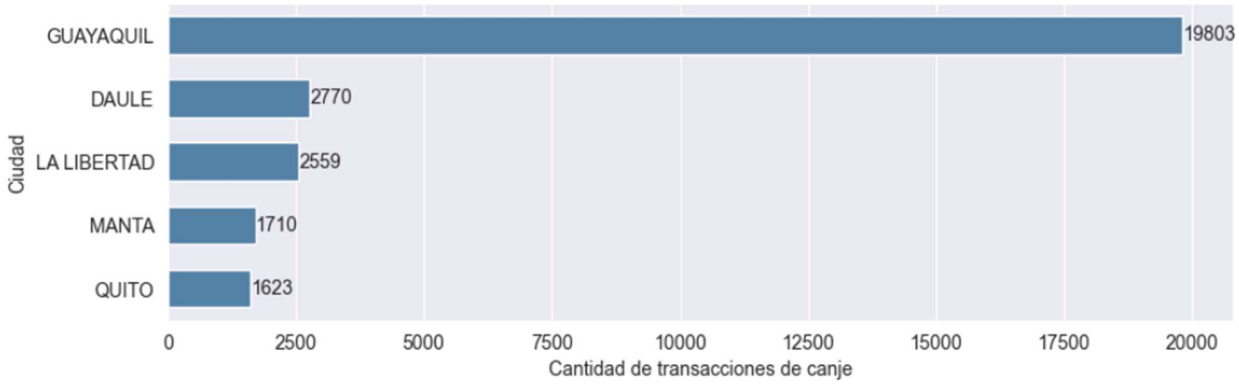


Figura 3.3 Ciudades con mayor cantidad de transacciones de canje

La figura 3.3 muestra las cinco ciudades con mayor número de transacciones de canje; como se observa, las transacciones de canje en Guayaquil comprenden más del 50% de todo el conjunto de datos, mientras la ciudad siguiente representa el 7% del total. Considerando la diferencia en cantidad, el análisis se limitará a la información obtenida de Guayaquil al ser la más representativa del conjunto de datos.

El objetivo del sistema de recomendaciones será determinar los tipos de artículos de canje para sugerir al cliente, de modo que el departamento correspondiente al artículo

constituirá la variable objetivo del modelo de clasificación. Analizando las transacciones se puede determinar la cantidad de facturas para cada departamento:

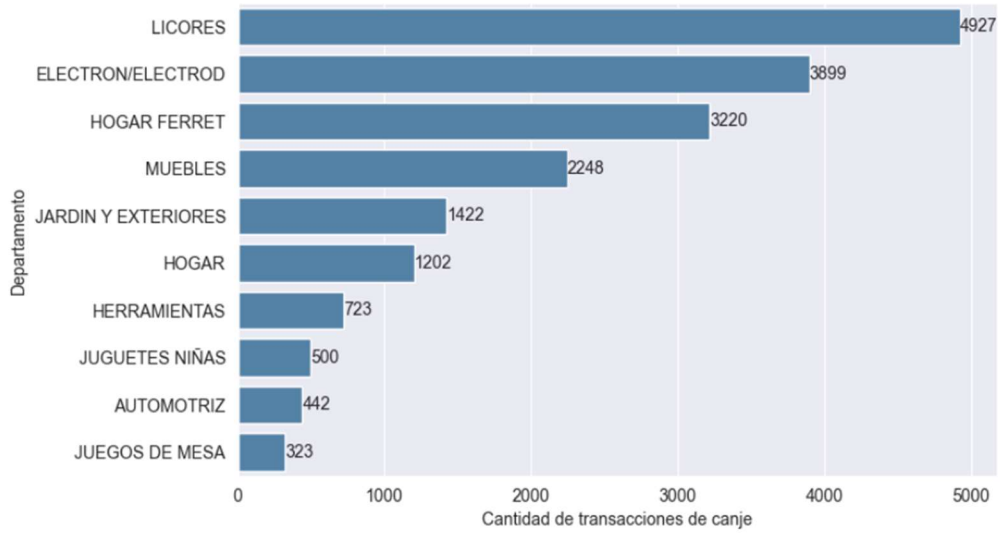


Figura 3.4 Departamentos con mayor número de ventas de canje en Guayaquil

La figura 3.4 muestra los diez primeros departamentos en cantidad de transacciones de artículos canjeables para Guayaquil, siendo, en este orden, los de licores, electrónicos/electrodomésticos, hogar de ferretería, muebles y jardín/exteriores, representando el 79.36% del total. Sin embargo, la diferencia en cantidad del primero con el quinto es de más del triple. Por consiguiente, para trabajar con clases objetivos balanceadas se considera un aspecto técnico, que es el de basar el análisis en las muestras más representativas, y el aspecto de interés del negocio, tomando en cuenta aquellas clases donde es más importante que los clientes incrementen sus compras.

Para el área comercial de la empresa, promover las ventas de canjes en los departamentos de licores (por tratarse de un producto perecible), electrónicos y hogar de ferretería (por el grado de obsolescencia que rápidamente pueden adquirir ante nuevas versiones con más o distintas funcionalidades) tienen mayor prioridad sobre los departamentos como muebles y jardín, de los cuales además dispone de un número menor de existencias. Los departamentos objetivo para este análisis serán por tanto los mencionados, considerando que, a futuro, con un período mayor de datos para la campaña de canjes se incluya un mayor número de departamentos.

En una tienda de *retail*, con múltiples secciones pertenecientes a varias líneas de negocio (supermercado, juguetes, ferretería, ropa), donde los clientes pueden adquirir sus productos en cualquier punto de venta del local, la cantidad de artículos de diferentes departamentos entre dos transacciones puede ser significativa, y por tanto afectar también el análisis. Las relaciones que se determinen en una venta de pocos artículos serán muy diferentes a aquellas en una venta con decenas de ellos, aun con la agrupación por departamentos.

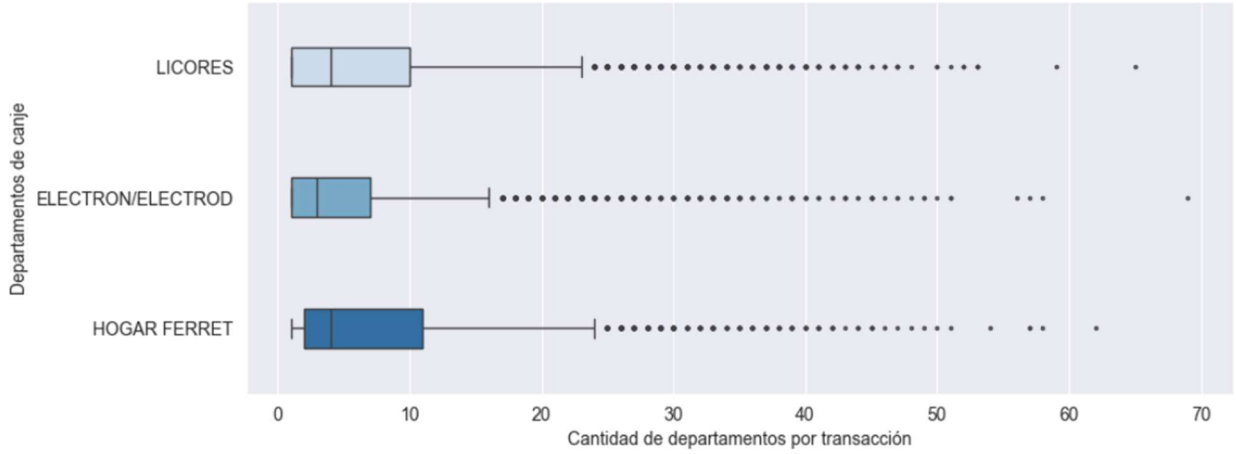


Figura 3.5 Diagrama de caja de departamentos normales por grupo de canje

En la figura 3.5 se observa la variabilidad en la cantidad de artículos para las diferentes transacciones, con una alta presencia de valores que deben excluirse del modelamiento. Para evitar este inconveniente, se aplicó una selección basada en el rango intercuartil o IQR, con el objetivo de eliminar las transacciones con *outliers* en el número de ítems y garantizar la estabilidad del modelo.

Sin embargo, aun con esta consideración, hay una diferencia importante en la cantidad de transacciones para cada clase objetivo. Para resolver este inconveniente, se aplicó una estrategia de *downsampling*, tomando en cuenta el número de transacciones de la menor de las clases, y utilizando esa cantidad para el análisis de todas las demás.

3.2 Definición de atributos

El conjunto de datos disponible contiene datos específicos de la venta (ID, fecha), relativos al lugar de la compra (tienda, ciudad), relativos al cliente (género, tienda en la

que frecuentemente compra), y relacionados con los artículos (categorías de precios, departamentos, cantidad y monto). Estos atributos pueden utilizarse en diferentes configuraciones de grafos y de los modelos en sí.

Como fue indicado en párrafos anteriores, el lugar de la compra se limitará a la ciudad de Guayaquil. Con respecto a los datos del cliente, estos son muy utilizados en sistemas de recomendaciones actuales para construir grafos heterogéneos o bipartitos. Sin embargo, un número importante de características del cliente más allá del género, como su edad, grupo demográfico o cantidad de puntos disponible es recomendable para realizar un análisis considerando el impacto de esas variables sobre las compras de artículos de canje. Por otra parte, los datos de la venta son útiles en un período mayor a una campaña de canje, o de considerarse obtener recomendaciones para diferentes temporadas. En ambos casos, sería necesario un número mayor de observaciones que cubran varios meses o años, en los cuales se debe considerar además los cambios en los artículos pertenecientes a las campañas.

El objetivo del sistema de recomendaciones es determinar qué departamentos sugerir considerando los artículos normales presentes en la transacción. Es decir, para una transacción específica, las relaciones ocultas entre tipos de artículos determinarán la clase de departamento que se aplicó como canje. Esto implica que los datos propios de la transacción, relacionados a los artículos vendidos, se utilizarán para el modelo de red neuronal, mientras que el valor correspondiente del ID del cliente se utilizará para relacionarlo con las recomendaciones obtenidas y almacenadas en base de datos. Por lo tanto, se considerarán como atributos los niveles de precio definidos desde las vistas de base de datos, junto con la cantidad vendida.

Sin embargo, también es necesario considerar los datos categóricos que constituyen los nombres de departamentos como representaciones (*embeddings*) de vectores numéricos que pueda interpretar el modelo de red neuronal. Para ello, se utilizará el algoritmo Node2vec, transformando estas descripciones en vectores de cinco elementos, utilizando la configuración predeterminada de este algoritmo.

Tabla 3.3 Ejemplos de valores vectorizados de nombres de departamentos

Departamento	*Embeddings* generados por node2vec				
ACEITES Y GRASAS	-0.041127	-0.021407	-0.045548	-0.119437	-0.059314
AIRE LIBRE	-0.181932	0.121849	-0.081555	-0.100302	-0.049237
ARTICULOS DE OFICINA	-0.167562	0.041395	-0.038856	-0.061794	-0.032309
SNACKS	0.023875	0.006561	0.085867	0.031220	-0.030537
UTILES ESCOLARES	-0.175601	-0.029528	-0.078964	-0.076857	-0.115927

3.3 Modelamiento de grafos

Las recomendaciones de artículos de canje pueden modelarse como una tarea de clasificación de grafos, donde cada grafo representa una transacción de venta, y donde los nodos están compuestos únicamente por los departamentos normales vendidos. La relación entre dos nodos, la cual constituye un borde del grafo, irá en ambos sentidos, y cada nodo está relacionado con todos los demás. Lo que se busca predecir para cada grafo es la clase a la que pertenece, representada por el departamento del artículo de canje que fue adquirido en la venta, el cual constituye la variable objetivo, el cual se reemplaza por un valor secuencial entre 0 y 2. De este modo, se construye un grafo homogéneo para que la red neuronal aprenda las relaciones entre los diferentes tipos de artículos y pueda determinar a qué departamento de canje pertenece dicho grafo o transacción.

Para definir los grafos a analizar por la red neuronal se deben diferenciar los atributos que constituirán los nodos y bordes. En estos últimos, se pueden definir atributos dependiendo del algoritmo de red neuronal utilizado. Como atributos de los grafos se definirán los niveles de precio y los vectores de nombres de departamento, mientras que la cantidad de ítems será un atributo para otorgar pesos (*weights*) a los bordes.

Las matrices de adyacencia están construidas a partir de los índices de cada departamento normal presente en la venta; al estar relacionados todos los artículos entre sí, se generarán todas las posibles combinaciones entre nodos a través de la función *itertools.permutations* de Python. La figura 3.6 muestra un ejemplo de una transacción del conjunto de datos modelada como un grafo.

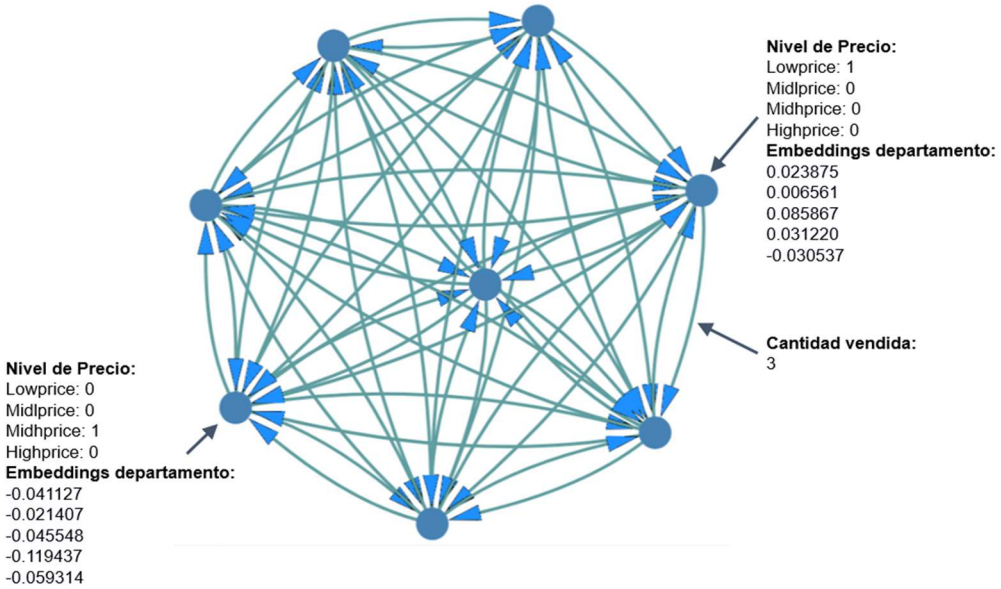


Figura 3.6 Transacción de venta modelada como grafo

3.4 Segmentación del conjunto de datos

En tareas de aprendizaje automático, es importante la división del conjunto de datos en los correspondientes subconjuntos de entrenamiento, validación y test, utilizando los dos primeros para prueba y mejoramiento del modelo. Para ello, se realizó una separación aleatoria de los tres conjuntos, utilizando la función *train_test_split* del paquete scikit-learn de Python, respetando la proporción de clases objetivo en cada uno (haciendo uso de la propiedad *stratify*), de modo que en ninguna etapa del proceso se presente un desbalance de clases que afecte a la capacidad del modelo de aprender de los datos, reduciendo el riesgo que aprenda las relaciones de tipos de artículos de una clase en detrimento de otras. Los valores definidos para separación fueron del 70% de los grafos para entrenamiento, 15% para validación y 15% para prueba.

A través de *Dataloaders* proporcionados por PyTorch, la carga de datos para análisis del modelo se realizó utilizando *batches* de grafos, de modo que el procesamiento se ejecute de forma más ágil al procesar grupos de *n* grafos a la vez. Diferentes tamaños de *batches* se evaluaron para encontrar el valor más adecuado en cada modelo.

3.5 Diseño de algoritmos y modelos

Para la elección del algoritmo a utilizar se consideraron cinco arquitecturas de redes neuronales de grafos implementadas como capas convolucionales en PyTorch Geometric: GCN a través de GCNConv, k-GNN mediante GraphConv, GAT utilizando GATConv, modelos Chebyshev con ChebConv y finalmente, la arquitectura GraphSAGE implementada en PyG a través del operador SAGEConv, para los cuales se evaluaron las métricas de rendimiento detalladas en el capítulo 2.

3.5.1 Experimentos

Para el entrenamiento de los diferentes modelos se realizaron múltiples experimentos con diferentes valores de hiperparámetros, entre los que se consideraron los valores que se muestran en la tabla 3.4:

Tabla 3.4 Valores de hiperparámetros utilizados en entrenamiento

Hiperparámetro	Valores de prueba
Número de neuronas	16, 32, 64, 128
Learning rate	0.1, 0.05, 0.01, 0.005, 0.001, 0.0005, 0.0001
Número de capas	2, 3, 4, 5, 6
Tamaño del *batch*	1, 4, 8, 16, 32, 64
Regularización L2	1e-5, *None*
Porcentaje de *dropout*	0.1, 0.15, 0.2, 0.25, 0.3, 0.5, 0.6
*Multi-attention heads**	2, 4, 8
Polinomiales**	2, 4
Número de épocas	1000, 1500, 2000, 2500, 3000, 5000

* Para arquitectura GATConv

** Para arquitectura ChebConv

En lo referente a la configuración interna de las redes, las configuraciones de *dropout* utilizadas se aplicaron para cada capa convolucional, con el método de *Global average pooling* previo a una capa lineal totalmente interconectada como capa de salida del modelo. Adicionalmente, se aplicó normalización de los datos de atributos de los nodos y los bordes de los grafos, para cada *batch* procesado en cada época de entrenamiento.

Se utilizaron además las funciones Adam (*Adaptative moment estimation*) para optimización, incluyendo el valor mencionado como regularización L2, y *CrossEntropyLoss* como función de pérdida, la cual en PyTorch internamente ejecuta una función softmax que permitirá establecer las probabilidades de pertenencia del grafo a cada clase objetivo. Pruebas iniciales con SGD (*Stochastic gradient descent*), otra de las funciones de activación utilizadas con frecuencia en redes neuronales, arrojaron resultados con rendimientos considerablemente más bajos en comparación con la función Adam. La figura 3.7 muestra el *pipeline* de entrenamiento utilizado en los experimentos.

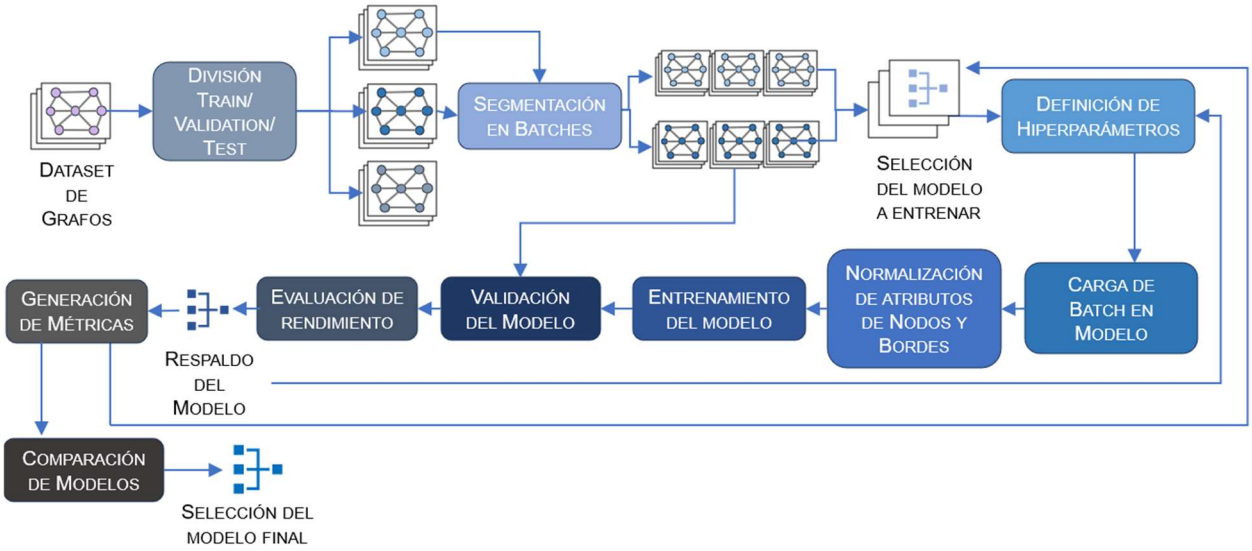


Figura 3.7 Esquema general del pipeline de entrenamiento

3.5.1.1 Modelos GCNConv, GraphConv, GATConv y ChebConv

Las cuatro arquitecturas indicadas tienen en común que permiten trabajar con atributos en los bordes de los grafos, incorporándolos como una característica más al modelo para analizar las relaciones entre los nodos. Los modelos GATConv y ChebConv tienen además dos hiperparámetros de interés que no poseen GCNConv y GraphConv: Para GATConv, el número de *multi-attention heads* que representan subredes separadas ejecutándose en paralelo dentro del modelo, y, para ChebConv, el número de polinomiales que determina el tamaño de los vecinos de los nodos que serán considerados por cada convolución. A diferencia de los otros modelos, se seleccionó LeakyReLU con el parámetro de *negative_slope* en 0.2 como función de activación en el modelo GATConv de acuerdo con la documentación del modelo original [25, p. 3], la cual

asegura que todas las neuronas de la red (exceptuando las inactivas por *dropout*) contribuyan al resultado, aun si sus entradas son negativas.

3.5.1.2 Modelo SAGEConv

Además de los modelos descritos anteriormente, se evaluó la arquitectura GraphSAGE. La fortaleza de este modelo inductivo se basa en obtener atributos de nodos para generar eficientemente representaciones de nodos para datos no vistos con anterioridad, mediante el aprendizaje de una función de muestreo y agregación de atributos de los vecinos locales de un nodo determinado, en lugar de entrenar representaciones individuales para cada nodo. Contrario a los métodos de representación de nodos basados en factorización de matrices como GCN, en este modelo se incorporan características del nodo en el algoritmo para aprender simultáneamente la estructura topológica de los vecinos de cada nodo como la distribución de las características de los nodos entre sus vecinos [32]. Esta arquitectura es usada como base en sistemas de recomendación de Uber Eats [19], o Pinterest [33], quienes aprovechan la escalabilidad a miles de millones de nodos que permite el modelo.

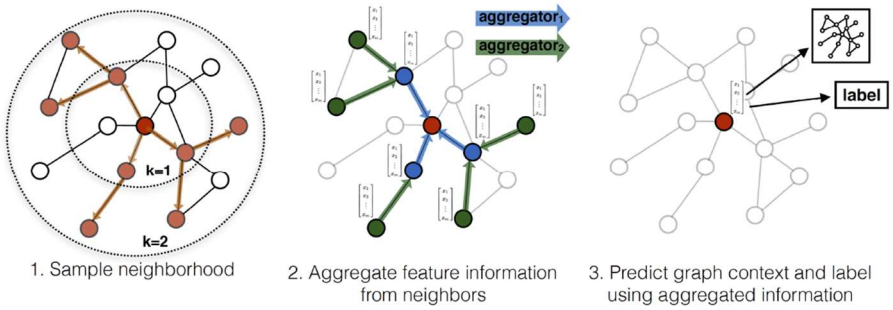


Figura 3.8 Ilustración de la técnica de muestreo y agregación de GraphSAGE [32]

PyG implementa GraphSAGE a través de una variante llamada SAGEConv, la cual mejora la arquitectura original a través del uso de un operador convolucional más expresivo. A diferencia de GraphSAGE, SAGEConv utiliza el promedio de la representación de los vecinos de los nodos, normalizado por el grado de cada vecino, como función de agregación, lo cual permite la identificación de características más complejas en los nodos y capturar información con mayor granularidad sobre la estructura del grafo [34].

A diferencia de modelos anteriores, las capas convolucionales SAGEConv no trabajan con atributos en los bordes de los grafos, por consiguiente, la cantidad de ítems normales vendidos que se utilizó para asignar pesos a los bordes se incorpora como un atributo más de los nodos, y de este modo conservar dicha información para el análisis.

3.6 Selección del modelo

Una vez definidos los algoritmos a evaluar, se procede a entrenar los modelos. Los mejores rendimientos obtenidos para cada hiperparámetro de acuerdo con la arquitectura utilizada se detallan en la de la tabla 3.5.

Tabla 3.5 Valores de hiperparámetros con mejor rendimiento por modelo

Hiperparametro	GCNConv	GraphConv	GATConv	ChebConv	SAGEConv
Número de neuronas	32	64	64	64	64
Learning rate	0.001	0.001	0.001	0.001	0.001
Número de capas	3	3	3	3	3
Tamaño del *batch*	16	16	8	8	8
Regularización L2	1e-5	1e-5	1e-5	1e-5	1e-5
Porcentaje de *dropout*	0.15	0.15	0.25	0.25	0.25
*Multi-attention heads**	N/A	N/A	8	N/A	N/A
Polinomiales**	N/A	N/A	N/A	2	N/A
Número de épocas	3000	3000	3000	3000	3000

Se determinó que algunos hiperparámetros no tienen mayor variación entre modelos, como son el número de capas, épocas y de regularización L2. El número de neuronas y de *dropout* coincide en la mayoría de los modelos, por lo que se deduce que estos se benefician de una arquitectura de baja complejidad, sin necesidad de una gran profundidad en capas o un número elevado de neuronas.

Los valores obtenidos por el algoritmo SAGEConv obtuvieron los rendimientos más altos durante el proceso de entrenamiento y validación, seguido por el método GATConv. Los métodos ChebConv, GCN y GraphConv tuvieron rendimientos significativamente más bajos. Esto se explica principalmente por la naturaleza inductiva de los dos primeros métodos, los cuales están diseñados especialmente para generalizar las relaciones sin conocer de antemano la estructura de los grafos, y realizar mejores

predicciones en datos no observados. Si bien la arquitectura GATConv incluye una mejora sobre SAGEConv en tanto se calculan coeficientes de atención para determinar cuán importante es un nodo respecto a otro, en los grafos modelados con los datos disponibles todos los nodos se relacionan entre sí, y por tanto contribuyen del mismo modo. En este método, solo las cantidades vendidas otorgan peso a los bordes, pero en el modelo SAGEConv, al considerarse la cantidad se considera como un atributo más del nodo, el modelo enfocándose enteramente en ellos y agregar las características de los vecinos obtenidos como muestra para obtener las representaciones, de modo que adquieren más información a incorporar más significativamente en el aprendizaje.

Si bien entre el método SAGEConv y GATConv la diferencia en exactitud fue del 8.75%, otro factor considerado fue el tiempo y recursos utilizados en el entrenamiento del modelo. Mientras SAGEConv tardó 2.7 horas en entrenarse con a GATConv le tomó 1.8 días, lo cual constituye un inconveniente a medida que se incrementen los datos disponibles en la empresa. Por consiguiente, la arquitectura SAGEConv se eligió como la más adecuada para el motor de recomendaciones. La figura 3.9 muestra un esquema de la arquitectura del modelo aplicado a los datos de la empresa.

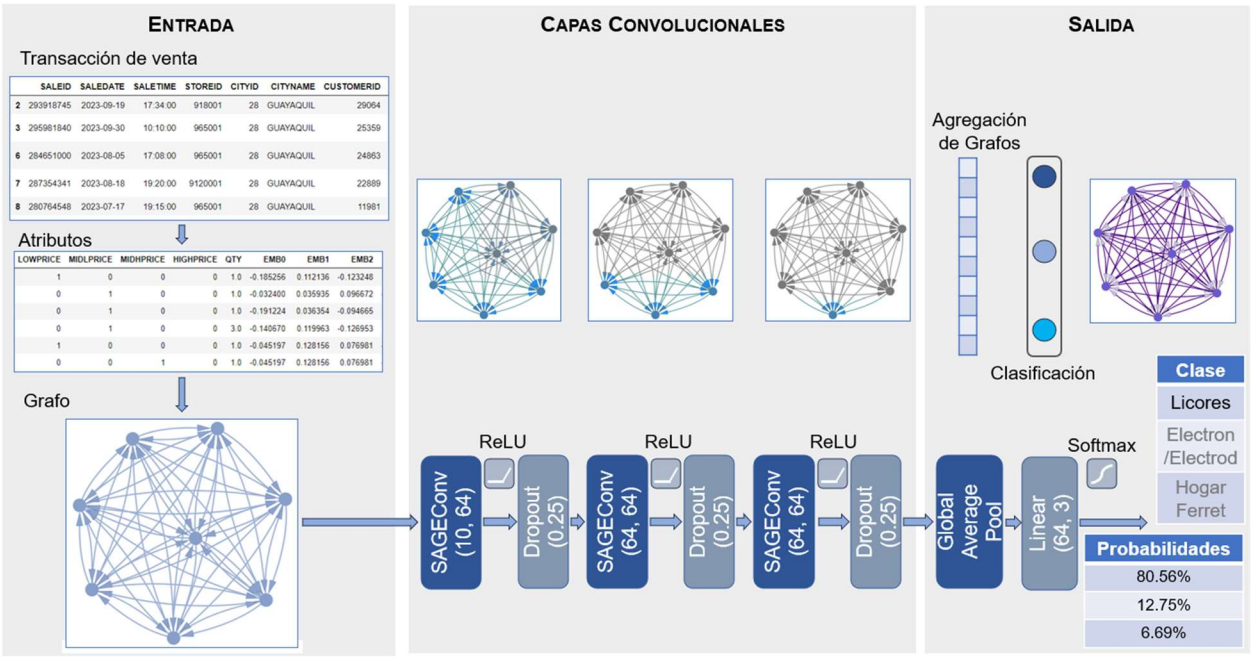


Figura 3.9. Esquema de la arquitectura de red neuronal seleccionada para el motor de recomendaciones

3.7 Métricas y resultados

Como se mencionó en el capítulo anterior, las métricas para evaluar los resultados entregados por el entrenamiento del modelo utilizado en la solución propuesta incluyen la exactitud (*accuracy*), precisión (*precision*), sensibilidad (*recall*) y *F1 Score*, las cuales analizan la cantidad de predicciones correctas (positivas o negativas) al mismo tiempo que los falsos positivos o negativos. La tabla 3.6 muestra los resultados obtenidos luego del entrenamiento, correspondiente al conjunto de datos de validación, mientras que la figura 3.10 muestra las curvas de *accuracy* a través de las diferentes épocas.

Tabla 3.6 Resultados del entrenamiento de modelos evaluados

Modelo	Accuracy	Precision	Recall	F1 Score
GCNConv	0.6268	0.6278	0.6268	0.6263
GraphConv	0.6385	0.6500	0.6385	0.6325
ChebConv	0.7668	0.7735	0.7668	0.7641
GATConv	0.8163	0.8185	0.8163	0.8156
SAGEConv	**0.9038**	**0.9055**	**0.9037**	**0.9034**

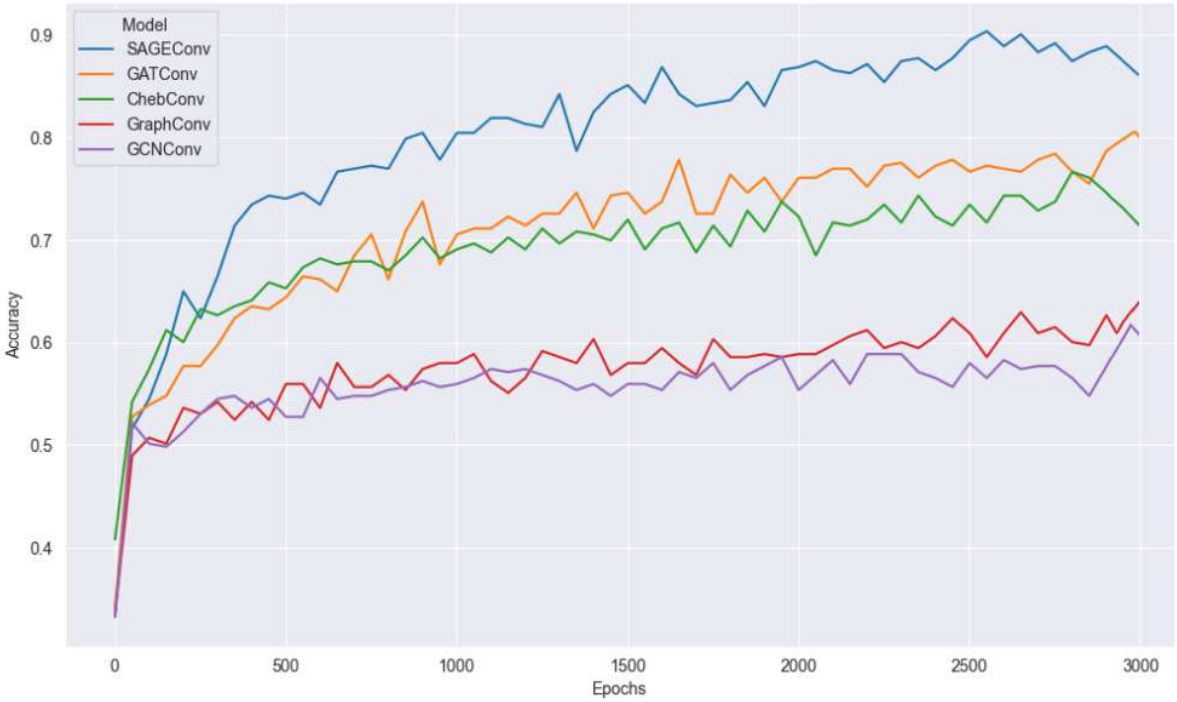


Figura 3.10. *Accuracy* de los modelos evaluados

Para el modelo utilizado, una predicción correcta positiva se define si la compra efectuada por el cliente incluye el tipo de artículo de canje entregado como resultado por

el modelo. Un falso positivo implica que en la compra se incluyó un tipo de artículo de canje distinto a la clase que predijo el modelo, y un falso negativo corresponde a tipos de artículos de canje que se determinaron diferentes a la clase evaluada, pero que en realidad sí se adquirieron. Los sistemas de recomendación son altamente sensibles a los casos de falsos positivos, por cuanto representan recomendaciones comprobadas como no relevantes [35]. Desde el punto de vista del negocio, no es recomendable un número alto de falsos positivos, pues esto indicaría que el modelo no está realizando buenas recomendaciones de interés para los clientes. A través de la matriz de confusión se pueden verificar estos valores:

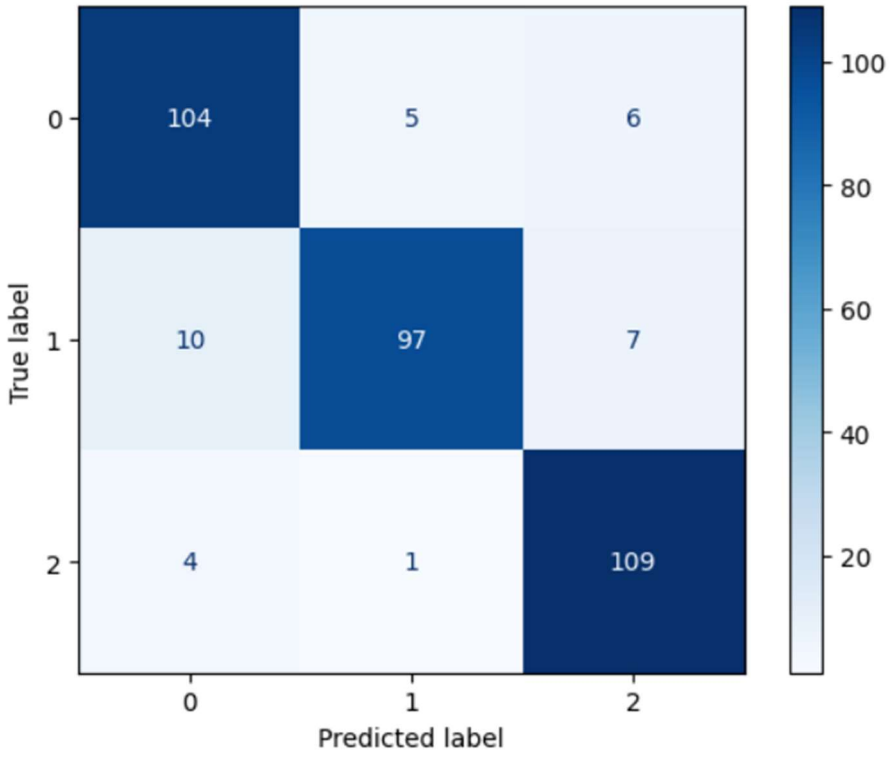


Figura 3.11 Matriz de confusión del modelo SAGEConv

Considerando los valores para las diferentes clases en la matriz de confusión, se determina que, en total, 33 observaciones son clasificadas como falsos positivos, el 9.62% del total de observaciones del conjunto de validación, lo cual no se considera un valor que pueda preocupar a la empresa. A partir de los datos de la matriz, se pueden determinar además los valores de las métricas para cada una de las clases, como muestra la tabla 3.7, donde, si bien no existe una diferencia considerable en los valores de *accuracy*, se observa que el modelo tiene una precisión superior en el número de

transacciones pertenecientes a las clases de electrónicos/electrodomésticos y hogar ferretería, con un resultado ligeramente menor para la clase de licores.

Tabla 3.7 Métricas por clase en el modelo SAGEConv

Clase	Accuracy	Precision	Recall	F1 Score
0: Licores	0.9329	0.8813	0.9043	0.8927
1: Electrónicos/Electrodomésticos	0.9475	0.9417	0.8508	0.8940
2: Hogar Ferretería	0.9271	0.8934	0.9561	0.9237

Finalmente, se pueden graficar las curvas ROC para cada clase, con su correspondiente valor AUC, como se indica en la figura 3.12.

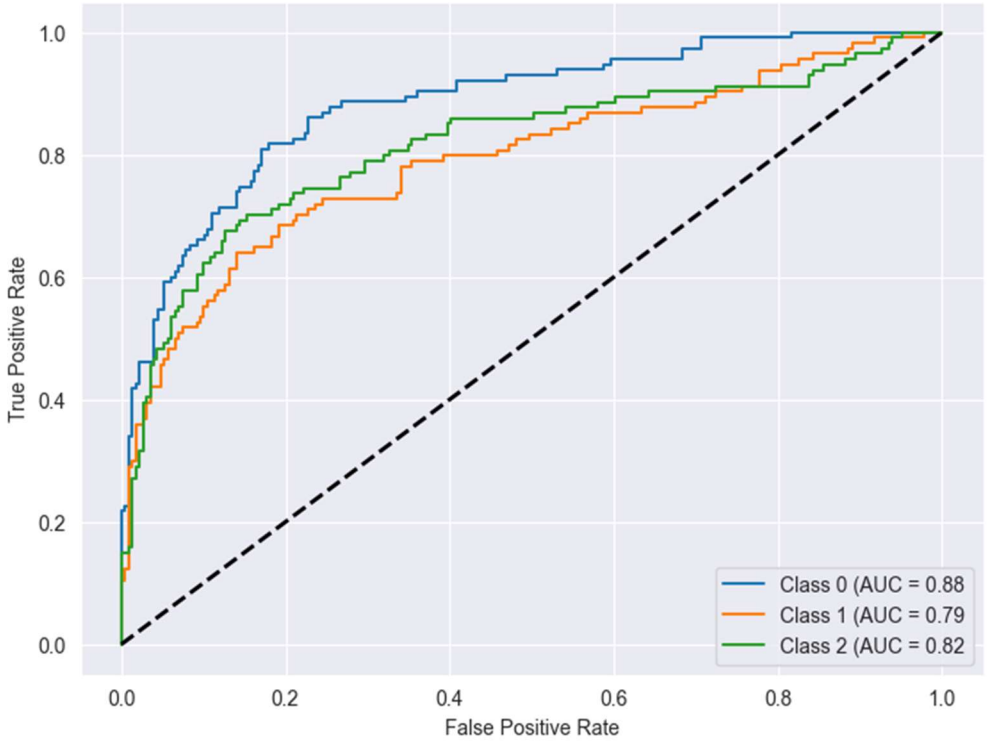


Figura 3.12 Curvas ROC y AUC para las clases del modelo SAGEConv

3.8 Módulos del sistema

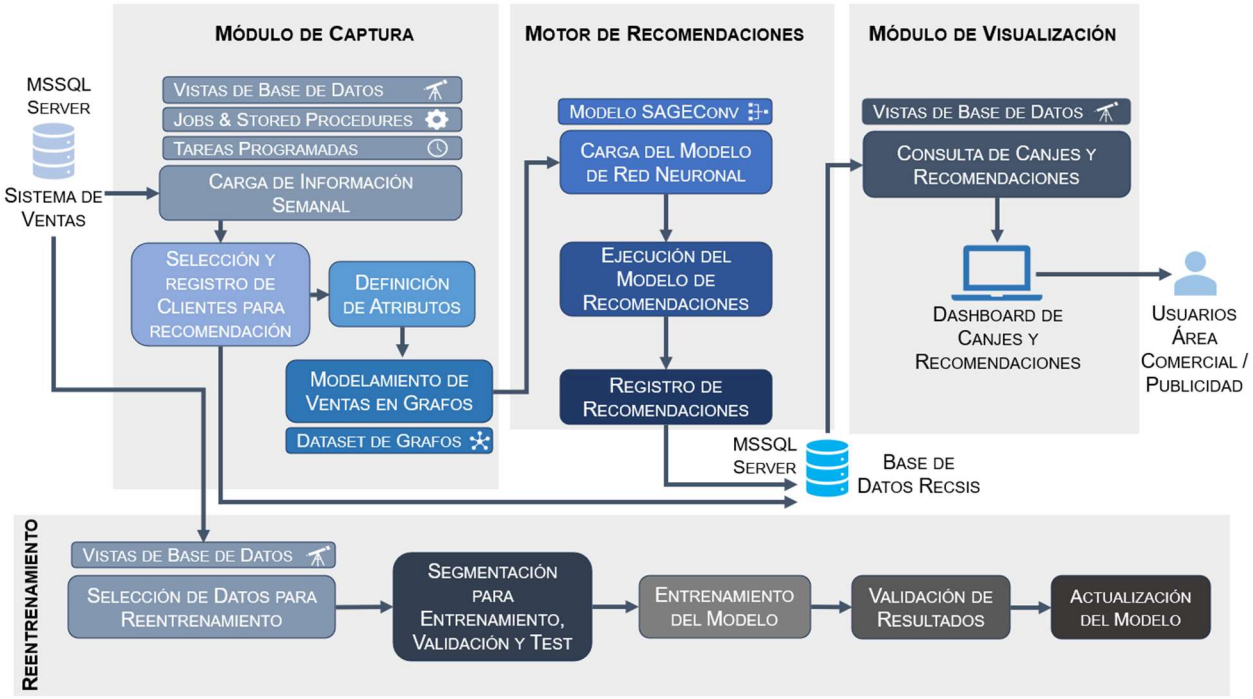


Figura 3.13 Arquitectura del sistema de recomendaciones

Una vez entrenado el modelo, se debe utilizar para generar recomendaciones sobre ventas subsiguientes. Como se mencionó en el primer capítulo, el Sistema de Recomendaciones propuesto se compone de tres módulos, según se observa en la figura 3.13, y que se detallan a continuación:

3.8.1 Módulo de captura

Obtiene los datos de la base de datos de ventas del sistema transaccional de punto de venta de la empresa, a través de un *job* de base de datos configurado en el motor RDBMS MSSQL Server, el cual ejecuta un *Stored Procedure* que extrae la información de las tablas de la base de datos central de ventas y las almacena en tablas de preprocesamiento en una base de datos creada especialmente para la solución y denominada RECSIS. Esta base contendrá tanto las tablas con los datos de origen, como aquellas con los datos de resultados del análisis realizado por los modelos.

La ejecución de las operaciones de carga se realizará al inicio de cada semana laboral, extrayendo los datos del período semanal anterior, los cuales se identificarán a

través de un ID y fecha de proceso, para poder relacionarse posteriormente con las recomendaciones generadas.

Durante la fase piloto de implementación, las ventas extraídas para análisis incluirán los artículos de canje para confirmar las predicciones realizadas por el modelo, Luego de este período, las ventas que se carguen en el sistema no tendrán artículos de canje. Al igual que en el tratamiento de datos previo al entrenamiento del modelo, la información obtenida se validará para no incluir artículos devueltos.

3.8.2 Motor de recomendaciones

Su función es ejecutar semanalmente los scripts desarrollados en Python que toman la información de las tablas alimentadas por el módulo de captura, filtrando todas las ventas del último proceso de carga ejecutado, y realizando las transformaciones de datos a grafos requeridos por el algoritmo SAGEConv y ejecutar el modelo entrenado para determinar los artículos recomendados. Las probabilidades entregadas por el modelo se almacenarán en estructuras de resultado creadas en la base RECSIS, donde se incluye además la información del cliente y la fecha en la que se generan las recomendaciones.

La ejecución de estos scripts se realizará mediante llamadas a línea de comandos mediante un archivo de procesamiento por lotes (.bat), y programadas a través de la herramienta *Task scheduler* del servidor Windows que albergará las aplicaciones.

3.8.3 Módulo de visualización

Comprende el *dashboard* definido en PowerBI al que tendrán acceso los usuarios de las áreas Comercial y de Publicidad, donde podrán visualizar los resultados semanales de las ventas con canjes, y cuáles son las recomendaciones entregadas por el motor para el desarrollo de las campañas publicitarias, o modificaciones en la estrategia de mercadeo de los artículos canjeables o en la lista de ellos, sean códigos, puntos, o descuentos otorgados.

El *dashboard* interactivo estará configurado para conectarse directamente a la base de datos RECSIS y las recomendaciones a mostrar por defecto se visualizarán para los clientes a los que no se han generado recomendaciones en el último mes.

3.9 Infraestructura para procesamiento y almacenamiento

La solución propuesta requiere de tres componentes básicos para su funcionamiento, como son un motor de base de datos para almacenamiento de la información extraída de ventas, un servidor de aplicaciones para las librerías Python y el modelo entrenado y un servidor para la ejecución del *dashboard* de recomendaciones. Todos los componentes de hardware corresponden a equipos o servicios que ya posee o a los que tiene acceso la empresa, por ende, solo se requiere la definición de las bases de datos en el primero la instalación del ambiente de ejecución Python en el segundo y la publicación del *dashboard* en el PowerBI Service del tercero.

El RDBMS Microsoft SQL Server se eligió para la solución por cuanto es el motor utilizado en el sistema de ventas de la empresa, además de garantizar compatibilidad en la conexión desde PowerBI Service para facilitar la extracción de datos para el dashboard. Por consiguiente, la base de datos RECSIS se creará en ese mismo servidor, teniendo la ventaja que la instancia instalada cuenta con la versión Standard, por lo tanto, no existen limitaciones de tamaño máximo de la base.

El servidor de aplicaciones requiere la instalación de las librerías Python, Scikit-Learn, Pandas, NumPy, PyTorch y PyTorch Geometric, además de la configuración de la tarea semanal de análisis de la información de ventas. Este servidor contará con los permisos de firewall y políticas de acceso a la red necesarias para conectarse desde y hasta el servidor de base de datos. El PowerBI Service que actualmente posee la empresa permitirá la publicación del dashboard de recomendaciones; del mismo modo, se solicitará al área de redes la habilitación de los permisos de acceso al servidor de base de datos.

El detalle de las características de hardware y software requeridos por la solución se lista en la tabla 3.8.

Tabla 3.8 Características de hardware y software

Función del Servidor	Hardware	Software
Base de Datos	Intel Xeon CPU – 2.8 GHz 24 GB RAM Partición de datos: 1.6 TB SSD Partición de log: 550 GB SSD	Windows Server 2019 Datacenter Microsoft SQL Server 2014 SP3
Aplicaciones	Intel Xeon CPU – 2.8 GHz 16 GB RAM 500 GB SSD	Windows Server 2019 Datacenter Python 3.10 (Incluye librería NumPy 1.23.5) PyTorch 2.0.1 PyTorch Geometric 2.0.2 (Incluye librerías Scikit-learn 1.2.2 y Pandas 1.5.3)
Visualización	N/A	PowerBI Service (SaaS)

3.10 Plataforma de visualización

A través del servicio de PowerBI se podrá acceder al Módulo de Visualización compuesto por el *dashboard* interactivo, el cual mostrará las ventas realizadas a través de la modalidad de canje de puntos, así como las recomendaciones de artículos de canje para conjuntos específicos de artículos vendidos.

Los gráficos implementados permitirán elegir filtros específicos de rango de fechas, grupo de artículos y tiendas, los cuales se aplicarán a cada una de las siguientes opciones:

3.10.1 Vista resumen

Figura 3.14 Vista Resumen del *dashboard* de visualizaciones

Presenta información general sobre las ventas con canjes realizadas por formato de tienda y artículos durante un período determinado, así como la cantidad de recomendaciones realizadas, los valores de ticket promedio por línea del negocio y estadísticas de canjes totales durante la última semana.

3.10.2 Vista análisis de ventas

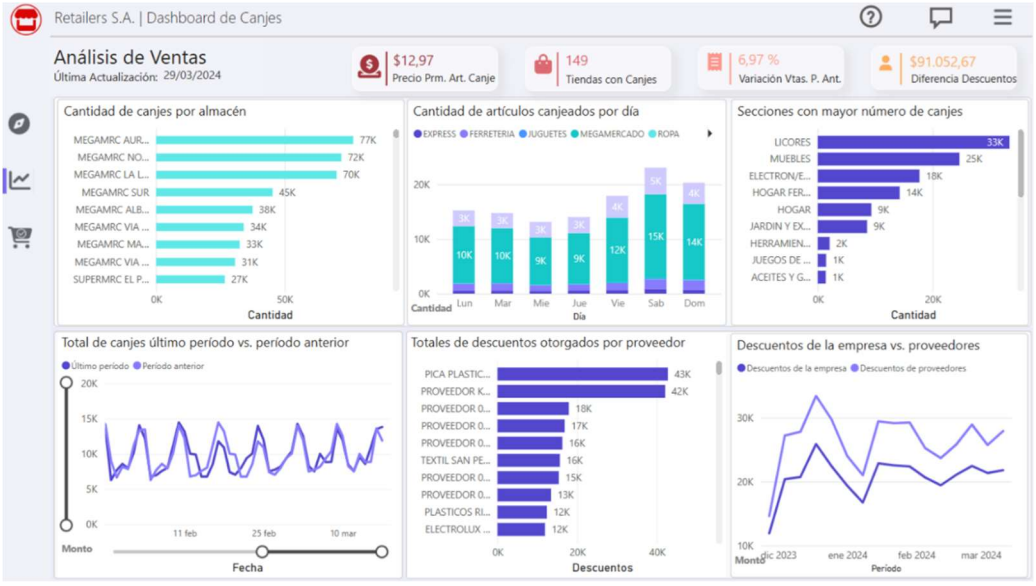


Figura 3.15 Vista de Análisis de Ventas en *dashboard*

Permite observar el comportamiento de las ventas de artículos canjeables por almacén, día o sección, así como los valores de descuentos otorgados por la empresa y los proveedores de los artículos.

3.10.3 Vista recomendaciones

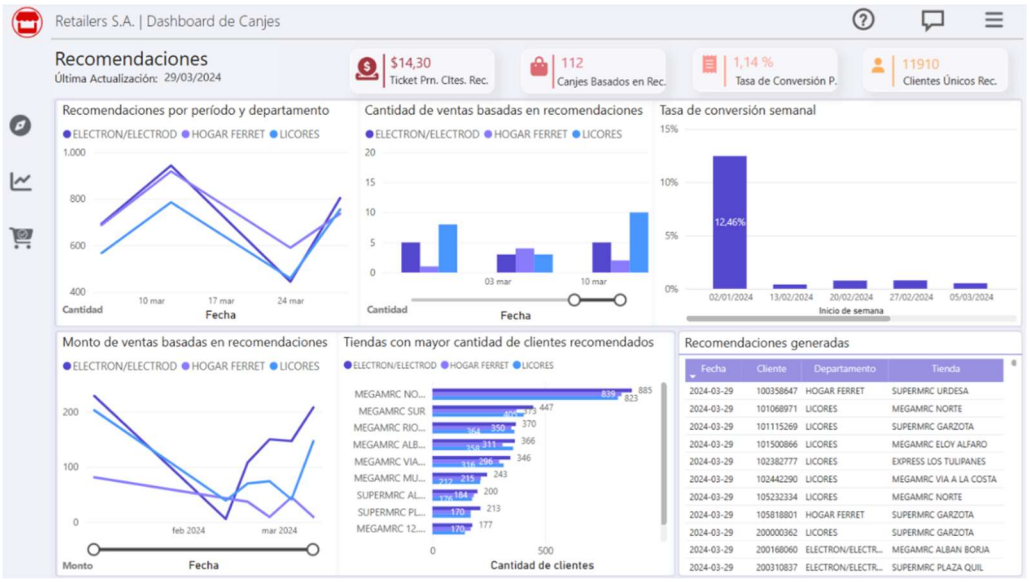


Figura 3.16. Vista Recomendaciones en *dashboard*

Muestra los departamentos más recomendados para los diferentes transacciones y clientes, así como las cantidades y montos de compras realizadas sobre la base de las recomendaciones de acuerdo con la tasa de conversión obtenida para un período determinado.

CAPÍTULO 4

4. ANÁLISIS DE RESULTADOS

Los resultados obtenidos por la solución propuesta se analizan en este capítulo, a partir de las estrategias utilizadas para validarlos, los beneficios de la implementación y las técnicas de recolección para datos futuros.

4.1 Estrategias de validación

Se consideraron dos fases para validación, correspondientes a la implementación de la solución en un ambiente pre-productivo de la empresa, y al proceso de evaluación de los resultados. Para la primera fase, la solución se implementó en modo de piloto, accesible únicamente por el usuario encargado de la revisión de ventas de artículos canjeables para la línea de *retail*. El objetivo de este piloto fue la validación del funcionamiento correcto de la solución, sus tiempos de proceso y resultados entregados, por consiguiente, se contó con acceso a datos extraídos semanalmente del sistema de ventas.

El usuario final estuvo encargado de revisar el progreso de la campaña de artículos canjeables semanalmente, y verificar las recomendaciones generadas por el modelo a través de la interfaz compuesta por el *dashboard* de recomendaciones. La información que se muestra a través de esta herramienta incluye las visualizaciones y estadísticas que forman parte de los reportes que se consolidan manualmente en la actualidad a partir de la información total de canjes generada por el área de Sistemas, y que sirven de insumo para la revisión semanal de gerencia de compras sobre los artículos que deben mantenerse o no en las campañas de canje.

Por otra parte, las recomendaciones que se generan desde el modelo de red neuronal permitieron determinar los departamentos sugeridos para compra de los clientes. Si bien para el usuario no es posible realizar una comparación directa con sistemas existentes al no contarse en la actualidad con una solución dinámica de recomendaciones, sí es posible consultar los departamentos más recomendados a través de diferentes criterios, ya sea por tienda o rango de fechas.

La segunda fase de la evaluación se divide a su vez en dos etapas, compuestas por un período de validación *offline* basado en el segmento de datos de test del conjunto de datos original, sobre el que se calcularon las métricas utilizadas en la etapa de análisis de resultados del entrenamiento del modelo, y otra etapa de validación *online* compuesta por un período de captura de datos de compras sin artículos canjeables, en el cual se determinó un grupo especial de clientes para el cual se generaron recomendaciones por un tiempo determinado, seguido de un período de verificación de su comportamiento de compra para confirmar el cumplimiento de las predicciones generadas.

4.2 Criterios de evaluación

Se utilizaron los siguientes criterios para determinar el cumplimiento de los objetivos de la solución:

- Integración de la arquitectura propuesta con los datos de origen del sistema de punto de venta, para la extracción y tratamiento de ellos.
- Tiempos de ejecución, tanto en lo referente a la ejecución del modelo como a la entrega y visualización de resultados de canjes.
- Comparación de los resultados de canjes con los reportes manuales que se utilizan en la actualidad.
- Facilidad de uso y selección dinámica de parámetros y filtros en gráficos de ventas de la interfaz.
- Registro de resultados de recomendaciones en base de datos.

Los tiempos de ejecución se evaluaron conforme a lo que requiere la solución para el procesamiento de los datos de venta a través del modelo de red neuronal, el cual se realiza en horas de la madrugada para no afectar los procesos de contabilización de la empresa ni el registro en línea de transacciones desde el punto de venta. El *dashboard* de recomendaciones debe a su vez garantizar una consulta rápida a los datos, especialmente durante la aplicación de los filtros configurados.

A través de los filtros por rango de fechas, la información visualizada en el *dashboard* puede compararse con los reportes generados manualmente por los usuarios del área de Publicidad para garantizar que el análisis se lleva a cabo con la información

del período de ventas correspondiente de acuerdo con los criterios de selección que actualmente se utilizan.

4.3 Recolección de datos

La información extraída del sistema de punto de venta para la primera fase de evaluación estuvo basada en el subconjunto de test correspondiente al 15% de las ventas originales, el cual no fue utilizado durante la fase de entrenamiento del modelo. A partir de los grafos generados para cada transacción se alimentó el modelo de red neuronal para generar las recomendaciones a evaluar.

En la segunda fase de evaluación se obtuvo la información de ventas realizadas a clientes durante el mes de diciembre del 2023. A diferencia de la fase de entrenamiento del modelo, los datos extraídos en este proceso incluyeron las ventas a clientes fidelizados sin artículos canjeables, sin embargo, para garantizar la similitud de los datos respecto a los utilizados durante el entrenamiento, se filtraron las ventas realizadas únicamente en las tiendas pertenecientes a la ciudad de Guayaquil. El *pipeline* de extracción definido a través de *stored procedures* y vistas de la base de datos de ventas, junto al tratamiento realizado a través del código Python para el modelamiento de las transacciones en grafos y la generación de recomendaciones se detalla en el diagrama de la figura 4.1.

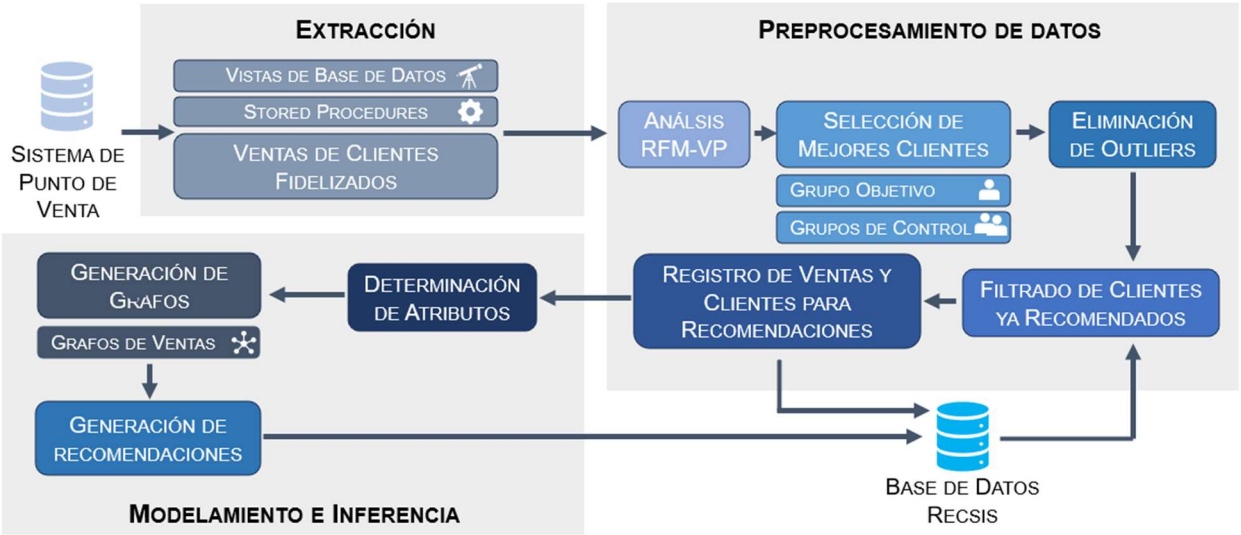


Figura 4.1 *Pipeline* de recolección y procesamiento de datos para evaluación

4.3.1 Grupo objetivo de clientes

Con el propósito de enfocar el análisis de las recomendaciones generadas por la solución, se consideraron las ventas de clientes preferentes con mayor sentido de pertenencia hacia la marca de la empresa, el cual es un grupo que al negocio interesa mantener. Para este fin se utilizó la metodología RFM-V *(Recency, Frequency, Monetary value, Variety*) como base para determinar el grupo de mejores clientes de la compañía durante el período de recolección, por cuanto considera, además de las ventajas del análisis RFM tradicional, la variedad de artículos, o en este caso, diferentes departamentos, que intervienen en una transacción, partiendo del principio que la compra de variedad de bienes adquiridos es uno de los parámetros principales para determinar el comportamiento de los clientes [36], y de este modo reflejar de un modo más representativo los diferentes escenarios de compras que ocurren en las diferentes tiendas.

Adicionalmente, la cantidad de puntos con la que cuenta un cliente es importante, ya que un valor considerable de puntos disponibles le facilitará adquirir aquellos productos que representen una ganancia mayor, a la vez que no es posible realizar recomendaciones de productos a clientes que no cuenten con un valor suficiente por haberlos utilizado al final de período de captura; al considerar este valor como una quinta variable P (*Points*) en conjunto con aquellas definidas por el modelo RFM-V, se obtiene un análisis específico para la empresa, y permite determinar con mayor precisión la importancia de los clientes para el negocio.

Para el cálculo de la recencia (R) se consideró el valor del día del período de evaluación, siendo 1 el día más antiguo y 30 el más reciente. La frecuencia (F) estuvo determinada por el número de transacciones efectuadas por un cliente en las diferentes tiendas, y el valor monetario (M) comprende el total de sus compras en el período. Como ya se indicó previamente, la variedad (V) se definió de acuerdo con el número de departamentos diferentes presentes en la venta, y los puntos (P) corresponden a la cantidad disponible al final del período de captura de datos. La tabla 4.1. muestra las estadísticas descriptivas para cada una de estas variables.

Tabla 4.1 Estadísticas de variables para determinar clientes objetivo

Estadística	Recency	Frequency	Monetary	Variety	Points
Mínimo	1	1	0.07	1	0
Primer Quintil	11	1	9.69	12	420
Segundo Quintil	18	2	24.99	14	1630
Mediana	21	2	37.27	15	3900
Media	19	3	54.49	17	5720
Tercer Quintil	23	3	84.78	18	7750
Cuarto Quintil	27	5	122.89	22	8600
Máximo	29	174	21364.05	136	24692

A través del método de dividir los valores en quintiles, se consigue una separación clara entre los diferentes niveles de clientes, y se identifica claramente a aquellos con los valores más altos en cada variable. Para ello se asignó un *score* entre 1 y 5, con 1 para aquellos valores en el primer quintil, y 5 para aquellos en el quinto quintil. Las recomendaciones generadas para análisis fueron consideradas para aquellos clientes con *score* de 5 en todas las variables. De este modo se evita afectar el análisis seleccionando clientes con un valor significativo en una variable en particular, por ejemplo, clientes con múltiples compras de poco valor o con una sola compra de un valor considerable. Del total de 526 691 clientes fidelizados en las ventas del período, 3985 representaron a aquellos con el score más alto. Con ello se pudo obtener el tamaño de la muestra recomendada de clientes para los que se generaron las recomendaciones, a través de la calculadora del sitio web *calculator.net*, con la que se determinó que 351 clientes eran necesarios como muestra significativa de la población, con un nivel de confianza del 95%. Estos clientes fueron elegidos aleatoriamente a través del uso de una vista de base de datos.

En cuanto a los datos del período de evaluación de las ventas con artículos de canje, se obtuvo un total de 13264 transacciones de venta con 10729 clientes únicos, de las cuales 705 corresponden a los clientes dentro del grupo objetivo con mayor *score*.

4.4 Evaluación

Se delimitaron además las transacciones del grupo de clientes objetivo para excluir las ventas con un número de departamentos correspondiente a valores

considerados como *outliers* en el conjunto de datos. Las estadísticas de los datos a evaluar se detallan en la tabla 4.2, los cuales fueron almacenados en estructuras indexadas de la base de datos RECSIS para consulta a través de vistas desde los scripts Python, e identificados bajo un grupo determinado por el año y semana de procesamiento, para evitar que clientes analizados en posteriores ejecuciones semanales del proceso dentro del mismo mes sean considerados nuevamente para la generación de recomendaciones, con el objetivo de cumplir con uno de los principios de los sistemas de este tipo, el cual es evitar los problemas de sobrecarga de información para los clientes [37], lo cual impacta de manera negativa en su predisposición para adquirir los bienes ofrecidos. Finalmente, se extrajeron los atributos de interés para el modelo, generando los grafos que utiliza la red neuronal para realizar las predicciones.

Tabla 4.2 Estadísticas del conjunto de datos para generación de recomendaciones

Característica	Valor
Total de registros	30753
Cantidad de transacciones/ventas	1384
Cantidad de clientes únicos	351
Cantidad de departamentos diferentes	268
Número de Tiendas con información para análisis	46

Un método válido para evaluar sistemas de recomendaciones es a través de un análisis *offline* a manera de simulación o *backtesting*, esto es, utilizar los datos históricos para estimar cómo un usuario pudo reaccionar a recomendaciones ofrecidas a partir del conocimiento de sus hábitos de consumo posteriores, buscando resultados predictivos del rendimiento *online* de los sistemas [38], con la ventaja de requerir menos tiempo y riesgos respecto a los métodos *online*. Si bien este método es adecuado para validar el subconjunto de test [39], no es posible reflejar el comportamiento real de los usuarios una vez que se generan las recomendaciones al grupo seleccionado de clientes, y, por consiguiente, no se puede determinar con seguridad su grado de incidencia y cuánto influyen en el proceso de decisión de compra del cliente. Por otra parte, el método de validación *online* o de *A/B test*, permite obtener resultados más cercanos a la realidad, reflejando el grado de influencia de la recomendación ofrecida en compras subsiguientes, como lo muestran diferentes experimentos y soluciones [40], convirtiéndolo en el más conveniente para los datos generados durante el período de

captura y análisis posterior, ofreciendo las recomendaciones a través de una de las opciones de la app de fidelización en reemplazo de la información estática actual, seleccionando aleatoriamente los artículos del departamento sugerido de acuerdo con los puntos disponibles en la cuenta del cliente durante un período de prueba de una semana, para luego analizar las compras de artículos canjeables realizadas por los clientes seleccionados durante las cuatro últimas semanas del mes de enero.

Las transacciones representadas a través de grafos en el segmento de test del conjunto de datos original, que representa la fase de validación *offline*, así como el detalle de las ventas generada durante el período de captura, que a su vez representa la fase *online*, se cargaron en el modelo entrenado SAGEConv para establecer las recomendaciones de acuerdo con la probabilidad que la estructura del grafo o transacción pertenezca a una de las clases objetivo.

En el caso de las transacciones del análisis *online*, los resultados entregados por el modelo, sus probabilidades y la identificación del cliente al que pertenecen se registraron en las tablas resultado de RECSIS, identificando la clase con la probabilidad más alta, que finalmente representa el departamento a recomendar. Para aquellos clientes con más de una transacción en el conjunto de datos, la recomendación determinada fue aquella con la probabilidad más alta entre sus transacciones. Luego del proceso de generación se obtuvieron los siguientes resultados de recomendaciones por departamento:

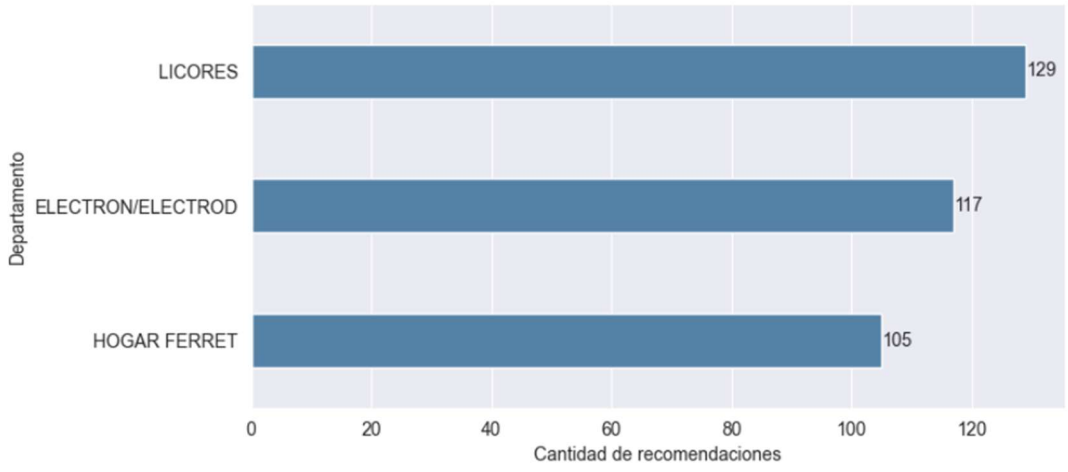


Figura 4.2 Recomendaciones por departamento

Como se observa, el departamento de Licores es el que mayor número de recomendaciones alcanzó en el período de análisis, lo cual indica que las transacciones de venta del grupo de clientes seleccionado tienden a asemejarse a las transacciones históricas de canjes de licores con las que fue entrenado el modelo SAGEConv.

4.4.1 Validación de resultados

Las métricas detalladas en los capítulos 2 y 3, es decir, *precision*, *recall*, *F1 Score* y ROC/AUC se utilizaron en primera instancia contra el conjunto de datos de test para determinar si las predicciones realizadas por el modelo SAGEConv sobre datos no observados en el entrenamiento guardan relación con los valores de rendimiento reportados.

Tabla 4.3 Métricas en el conjunto de test

Clase	Accuracy	Precision	Recall	F1 Score
0: Licores	0.9360	0.8965	0.9123	0.9043
1: Electrónicos/Electrodomésticos	0.9302	0.9043	0.9043	0.9043
2: Hogar ferretería	0.9360	0.9026	0.8870	0.8947
Rendimiento general	0.9012	0.9012	0.9012	0.9011

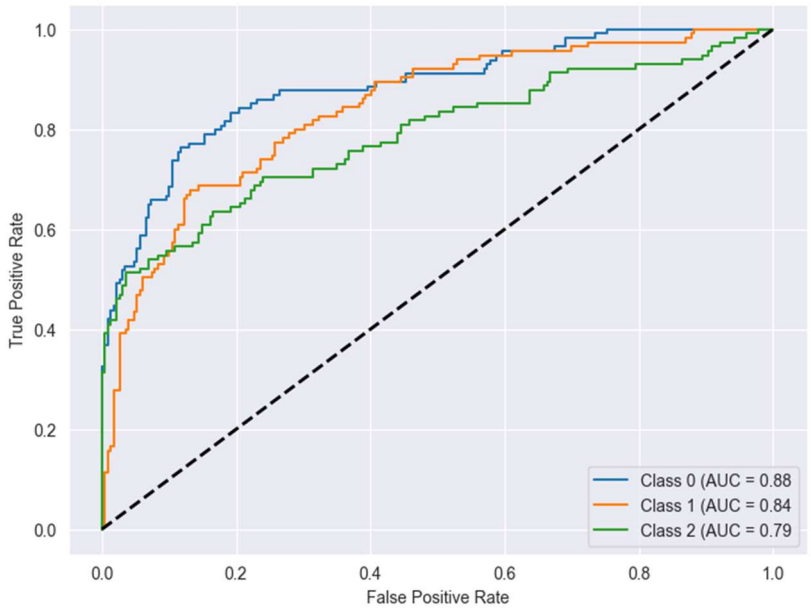


Figura 4.3 Curvas ROC y AUC para el conjunto de test

Los valores obtenidos para las métricas detalladas en la tabla 4.3, así como la gráfica de la figura 4.2, muestran que los resultados se asemejan a lo obtenido al final del entrenamiento, lo cual quiere decir que el modelo está generalizando correctamente la información de los grafos para establecer las recomendaciones.

Para la etapa de validación *online*, se compararon las compras posteriores al período de recomendación realizadas por los clientes seleccionados con las transacciones del resto de clientes del mismo grupo objetivo, segmentándolos aleatoriamente en diferentes grupos de control para considerar las compras de artículos canjeables que los clientes pueden realizar sin la influencia de recomendaciones recibidas. Esta segmentación también se hizo a través de extracción de datos a través de vistas de la base RECSIS. Para evaluar el valor que aporta al negocio el modelo de recomendaciones, se utilizó el *Conversion rate*, comparando los canjes que se predijeron correctamente contra el total de recomendaciones, y así cuantificar qué tanto se siguen las recomendaciones al hacer la venta de artículos de canje. Además, se verificaron otras transacciones de los clientes con canjes de departamentos diferentes a los predichos. La tabla 4.4 muestra los resultados de la evaluación de compras de clientes de acuerdo con las recomendaciones realizadas:

Tabla 4.4 Compras de clientes en función de las recomendaciones

Característica	Valor
Cantidad de clientes con recomendaciones	351
Clientes sin compras de artículos de canje	256
Clientes con compras de artículos de canje	95
Clientes con compras de otros departamentos de canje	36
Clientes con compras de departamentos diferentes a recomendados	13
Clientes con compras iguales a recomendaciones	44
Conversion rate	12.54%

Al finalizar el período de evaluación de cuatro semanas, se encontró que 95 clientes de los 351 que recibieron las recomendaciones efectuaron compras de artículos de canje, de los cuales, 36 adquirieron artículos de departamentos distintos a los grupos de Hogar ferretería, Electrónicos/Electrodomésticos y Licores, mientras 57 sí adquirieron artículos de estos grupos, de los cuales, 13 no coincidieron con los departamentos determinados en la recomendación. Un total de 44 clientes sí compraron exactamente

artículos del departamento recomendado, representando un *conversion rate* del 12.54%, lo que se traduce en una concreción de venta importante e indicativa de la fiabilidad de la solución.

Luego de separar los 351 clientes del grupo objetivo de los 3985 mejores clientes, se establecieron 10 grupos de control, 9 de ellos con 351 clientes y un grupo de 124, se compararon los resultados de los clientes que recibieron las recomendaciones contra el resto, comprobando cuántos realizaron canjes de los departamentos evaluados, y determinar si efectivamente, al recibir recomendaciones se produce un incremento sobre lo que normalmente compran los mejores clientes, sin recomendaciones de por medio. La figura 4.4 muestra los clientes que canjearon artículos de los departamentos recomendados en los grupos de control y en el grupo objetivo.

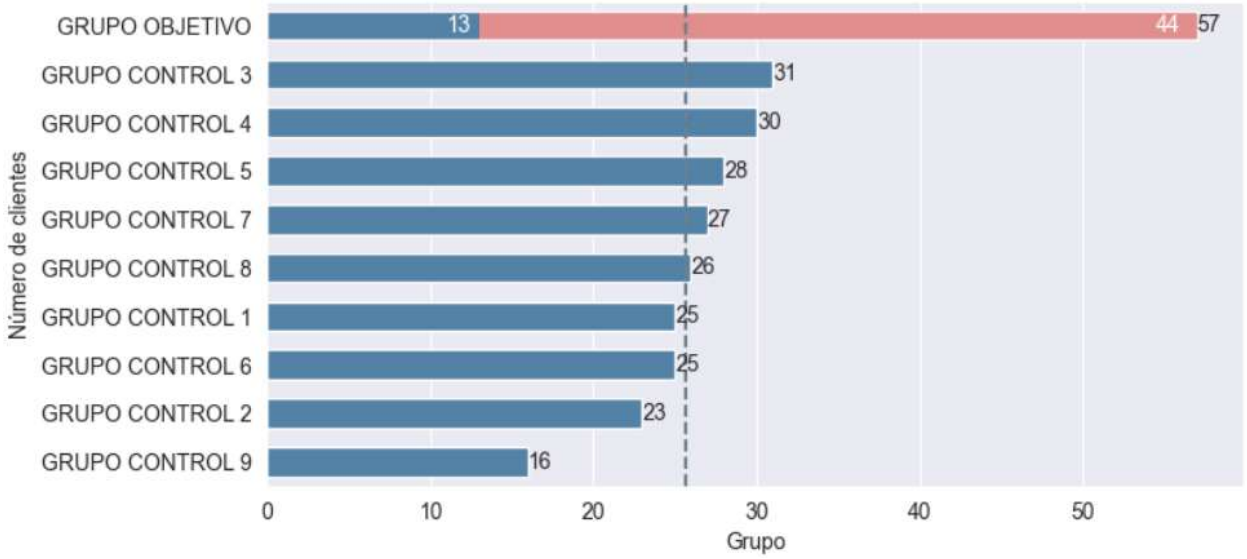


Figura 4.4 Compras de departamentos recomendados en grupos de clientes

Como se indicó anteriormente, de los 57 clientes con compras de los departamentos de canje recomendados, 44 realizaron compras de los mismos departamentos sugeridos, lo cual representó un incremento del 71.40% sobre el promedio de 25.67 clientes entre los demás grupos. Sobre el grupo inmediato inferior, que tuvo 31 clientes, las recomendaciones efectivas representaron a su vez un incremento del 41.94%. De este modo, es posible concluir que ofrecer las recomendaciones de canje sí tiene un efecto sobre las compras de los clientes.

4.5 Puesta en marcha y funcionamiento

Una vez configurados los procesos *batch* de extracción, carga e inferencia sobre los datos de ventas, se determinaron los tiempos de ejecución, en cumplimiento de lo detallado al inicio de este capítulo y de acuerdo con las siguientes tareas:

- Se definió un *job* de extracción de datos en la base RECSIS durante la madrugada para obtener las ventas de cada semana anterior mediante un *stored procedure*, y de este modo generar las recomendaciones para la siguiente semana. Para ello se determinó el tiempo que generaba el proceso, el cual, en el período de evaluación tardó cerca de dieciocho minutos. Es de resaltar que esto corresponde a una extracción completa de ventas de cuatro semanas.
- Un segundo *stored procedure* definido en el mismo *job* realizó el preprocesamiento de datos y la generación de grafos, con un tiempo de tres minutos y dieciséis segundos adicionales, hasta el registro correspondiente en tablas intermedias.
- Sobre la base de los tiempos obtenidos, se definió la hora de ejecución del proceso de generación de recomendaciones a través de una tarea programada del servidor para 15 minutos después de la ejecución del *job* anterior.
- Se observó adicionalmente que la ejecución del *job* y la tarea ocasionó un impacto moderado en el procesamiento y uso de memoria RAM del servidor (65% y 83%, respectivamente), lo cual se enmarca en lo tolerable para un proceso ejecutado durante un período de nula transaccionalidad.

4.6 Pruebas de funcionalidad

El objetivo de estas pruebas fue obtener información relevante de las transacciones de artículos de canje, recomendaciones y su visualización. La información obtenida a través del *dashborard* PowerBI permitió la selección intuitiva de diferentes criterios de filtrado y atributos de las ventas de departamentos canjeables, lo cual contribuyó a la rápida comprensión de su funcionamiento por parte del usuario final.

El usuario del área comercial con acceso al *dashboard* validó que la información necesaria para su análisis semanal comprendía las gráficas utilizadas para el

seguimiento de las campañas de artículos canjeables, destacando el grado de interacción de la interfaz. De igual modo, el tiempo de espera para obtener la información necesaria para los análisis semanales se reduce a cero, ya que la ejecución de los procesos en la madrugada le permite contar con los insumos necesarios a primera hora del día, en cumplimiento de la eficiencia en los procedimientos que busca el negocio al evitarse la solicitud de información a las áreas de Sistemas y Analítica.

4.7 Beneficios

Los resultados obtenidos permitieron determinar los beneficios contrastados a los costos actuales y al costo de implementación y mantenimiento de la solución. Los valores calculados corresponden al período regular en el que será utilizada la solución, el cual es de una semana.

Tabla 4.5 Costos actuales

Rubro	Cantidad	Valor / hora	Total
Operación			
Usuario de área Comercial	6 horas	$ 7.50	$ 45.00
Operador de Sistemas	1.5 horas	$ 12.50	$ 18.75
Operador de Analítica	1.5 horas	$ 11.25	$ 16.88
Total			**$ 80.63**

Los valores detallados en la tabla 4.5 corresponden a los costos por hora actuales en los que incurren los operadores de las áreas de Sistemas y Analítica para preparar la información semanal de seguimiento de artículos canjeables, y en el tiempo que el usuario final del área comercial invierte en analizar la información recibida y generar la información y gráficos relevantes que utiliza la gerencia.

La tabla 4.6 detalla los rubros en los que la empresa debería invertir para desarrollar una solución similar considerando los costos promedio del mercado, en un tiempo estimado de desarrollo de 6 semanas, incluyendo la programación, pruebas e implementación del sistema.

Tabla 4.6 Costos de la solución por desarrollo externo

Rubro	Cantidad	Valor
Desarrollo		
Analista de Datos/Programador	1	$ 1200.00
Científico de Datos/Jefe de Proyecto	1	$ 1900.00
Total		**$ 3100.00**

A diferencia de los valores actuales y de desarrollo externo indicados, para la solución implementada solo debe considerarse el costo semanal de operación, compuesto por tiempo designado para supervisión de Operador de Monitoreo nocturno, cuya función es el monitoreo general de sistemas en ejecución *batch* dentro del negocio. La tabla 4.7 muestra los valores comparativos de cada opción.

Tabla 4.7 Comparación de costos

Rubro	Valor
Costos operativos semanales actuales	$ 80.63
Costos de desarrollo	$ 3100.00
Costos de la solución (Operador nocturno, 1 hora/semana)	**$ 6.50**

En cuanto a los resultados obtenidos mediante las recomendaciones generadas, es posible cuantificar los beneficios del sistema de acuerdo con el valor total de las transacciones que fueron iguales a las recomendaciones durante el período de evaluación. Considerando que el valor de canje promedio entre los grupos de control fue de $ 16.47, y el promedio de clientes que realizaron compras sin recibir las recomendaciones, es posible determinar los valores que muestra la tabla 4.8.

Tabla 4.8 Ingresos por conversiones

Grupo de clientes	Valor Canje	Cantidad de clientes	Total
Compras iguales a recomendaciones	N/A	44	$ 794.27
Promedio en grupos de control	$ 16.47	25.67	$ 422.78
Diferencia clientes con y sin recomendaciones			**$ 371.49**

Cabe mencionar que estos valores corresponden al período acumulado de 4 semanas de evaluación, por consiguiente, se puede calcular el costo de operación del

sistema en el mismo período para contrastarlo ante los ingresos obtenidos, y así estimar un beneficio mensual de la solución.

Tabla 4.9 Beneficio mensual de la solución

Rubro	Valor
Diferencia ingresos por recomendaciones	$ 371.49
Costo operativo solución (4 semanas)	$ 26.00
Beneficio	**$ 345.49**

4.8 Estrategia de recolección de datos futuros

Para el análisis de los períodos subsiguientes, se tendrán en cuenta los siguientes aspectos al momento de la recolección de datos:

- Los clientes con una recomendación previa deben marcarse a nivel de la base de datos para ser excluidos del envío de comunicaciones personalizadas. Serán considerados nuevamente en el mes siguiente.
- Se establecerá semanalmente el grupo objetivo de clientes para recomendaciones, aplicando los métodos definidos en la etapa de evaluación.
- Para un despliegue de la solución posterior al piloto, se aplicará un esquema de seguridad para varios usuarios en el *dashboard* de recomendaciones, implementado a través de las opciones proporcionadas por PowerBI.
- Los cambios que se definan para departamentos de canje nuevos deberán evaluarse por parte del personal del área comercial; se acordó que un cambio mayor al 10% del total de departamentos para canje, si bien poco frecuente, requerirá un reentrenamiento del modelo. De igual manera, se considera un incremento del 5% en el número de tiendas en la ciudad de Guayaquil como umbral para reentrenamiento.
- El período para reentrenamiento se establece en seis meses, aun si no se presentan las variaciones descritas, sin embargo, la Gerencia comercial y el área de Publicidad pueden solicitar una revisión de la conveniencia de un reentrenamiento sin necesidad de llegar a los niveles definidos.

CONCLUSIONES

5. CONSIDERACIONES FINALES

La empresa considera de importancia el análisis de ventas realizadas por concepto de canjes cada semana, y las recomendaciones que aporte la solución, tanto para optimizar las tareas actuales de análisis como para el incremento de dichas ventas. En este sentido, y conforme a lo detallado en documento presente, se pueden definir algunas conclusiones sobre el proceso de desarrollo de la solución y recomendaciones a implementar a futuro.

5.1 Proceso de desarrollo

- Los sistemas de recomendaciones basados en la implementación de técnicas de aprendizaje automático constituyen herramientas cada vez más usadas por las empresas, especialmente en el ámbito del *retail*, para determinar más efectivamente el comportamiento de compra de sus clientes.
- La cantidad de artículos presentes en los conjuntos de datos de entrenamiento y evaluación, tanto de tipos canjeables como normales, hizo necesaria la agrupación por departamentos y la selección de aquellos más importantes para la generación de recomendaciones.
- La utilización del algoritmo Node2Vec permitió una representación numérica de las descripciones de los departamentos en una cantidad reducida de variables, a diferencia de las que hubieran sido necesarias mediante la aplicación de técnicas *one-hot encoding*.
- Las transacciones de ventas con y sin artículos de canje fueron representadas adecuadamente a través del modelamiento y clasificación de grafos. Al no disponerse de un número significativo de atributos de clientes en el conjunto de datos para entrenamiento, el análisis se enfocó en las características intrínsecas a la venta, como los departamentos a los que pertenecen los artículos modelados a través de nodos.
- De igual manera, las redes neuronales de grafos se constituyeron en el modelo idóneo para el descubrimiento de las características ocultas en las diferentes

transacciones de venta, al no poseer estas una longitud fija de variables o elementos.

- Se consideraron para el modelamiento las transacciones de la ciudad de Guayaquil, debido a su cantidad considerablemente mayor en comparación a otras ciudades, al igual que los tres departamentos de canje con mayor demanda y más importantes para el negocio.
- Entre los algoritmos evaluados, SAGEConv, en combinación con la normalización de atributos, regularización L2, procesamiento en *batches* e inclusión de capas *dropout*, permitió clasificar mejor los tipos de transacciones de canje de acuerdo con las métricas de *accuracy*, *precision*, recall y *F1 score* elegidas.
- Se definió un *pipeline* de extracción de datos del sistema de ventas en un ambiente pre-productivo para la evaluación de resultados, a través de la implementación de tareas programadas, vistas y *stored procedures* en una base de datos definida para almacenar los resultados de las recomendaciones generadas, con el objetivo que estén disponibles para consumo de otras soluciones, y facilitar el filtrado de clientes ya recomendados.
- Mediante la aplicación del modelo RFM-V, al cual se agregó una variable adicional P basada en los puntos (*points*) del cliente, se obtuvo un *score* para determinar a los mejores clientes, de los cuales se seleccionó una muestra estadísticamente representativa como el grupo objetivo que recibiría las recomendaciones, para compararse posteriormente con grupos de control.
- Luego de la validación de resultados, se determinó que ofrecer las recomendaciones a clientes sí tuvo un impacto significativo en las compras de artículos de canje realizadas por los clientes del grupo objetivo.
- El procesamiento y visualización de resultados, tanto de las ventas con canjes, como de las recomendaciones a través del *dashboard* interactivo basado en PowerBI permitió una consulta flexible e intuitiva de los datos, a la vez de reducir los tiempos dedicados a la extracción y manipulación actuales en las que interviene personal de las áreas de Sistemas, Analítica y Publicidad.

5.2 Recomendaciones a futuro

- Para incorporar un número mayor de departamentos de canje a recomendar, es conveniente analizar un conjunto de datos más extenso para entrenamiento, disponiendo de un mayor volumen de estas ventas y considerando además variaciones a través de diferentes períodos. Es posible que esto requiera generar otros modelos basados también en el algoritmo SAGEConv, pero específicos para aquellos departamentos, sin embargo, esto permitirá un análisis más preciso de las transacciones para estos canjes.
- Del mismo modo, para incluir en el análisis la información de ventas otras ciudades, se recomienda la generación de otros modelos diferentes al evaluado, debido a que clientes con una cultura y hábitos diferentes de compras podrían no ajustarse correctamente a las características encontradas por la red neuronal definida en la solución.
- Si bien la identificación de los mejores clientes que se requiere para la generación de recomendaciones puede utilizarse también en el desarrollo de estrategias promocionales del área de Marketing, es posible realizar para este fin una categorización de grupos de clientes a través de técnicas de *clustering*, según evalúen las áreas de Gerencia comercial y Analítica.
- La solución fue desarrollada bajo el concepto de omnicanalidad en la comunicación de los resultados a través del registro en base de datos, por tanto, es recomendable que otras herramientas implementadas por el negocio, tales como la generación de correos electrónicos para los clientes o el portal web accedan a las recomendaciones generadas.
- El alcance de los modelos de recomendaciones puede extenderse a ventas distintas a canjes, de tal manera que se utilicen como base de otras soluciones aplicables al *retail*, tales como asistentes virtuales interactivos en *apps*, recomendaciones personalizadas en sitios de *E-commerce*, o carritos de compra (*shopping carts*) inteligentes, en los que las sugerencias al cliente se efectúen en tiempo real, a medida que elige los artículos a adquirir, fomentando estrategias de *upselling* y *cross-selling*. Por consiguiente, se recomienda a la empresa la creación de un departamento específico de *Machine Learning Operations* (MLOps) que lidere la ejecución de proyectos de este tipo u otros donde se aplique Inteligencia artificial.

GLOSARIO

A/B Testing. Técnica utilizada para validar dos versiones diferentes de un sitio web o solución informática, con el objetivo de determinar cuál genera un mejor resultado en dos o más grupos seleccionados aleatoriamente.

Adaptive Moment Estimation (ADAM). Algoritmo de optimización utilizado comúnmente en *machine learning* para actualizar los pesos de una red neuronal durante el entrenamiento, es una extensión del algoritmo *Stochastic Gradient Descent* (SGD), diseñado para mejorar tiempos de entrenamiento.

Area Under the Curve (AUC). Métrica que evalúa el área bidimensional debajo de la curva *Receiver Operating Characteristic* (ROC), para determinar el rendimiento de un algoritmo al distinguir entre clases.

Artificial Neural Network. Conocida también como red neuronal artificial, modelo de inteligencia artificial compuesto de un conjunto de neuronas agrupadas en capas, que simula los procesos de aprendizaje efectuados por el cerebro humano.

Backtesting. Método de validar las predicciones generadas por un modelo utilizando datos históricos.

Basket Recommendation. Técnica en la que se busca recomendar ítems para la siguiente cesta (*basket*) de una compra, basándose en cestas de compra examinadas anteriormente.

Basket. Conjunto de artículos o ítems presentes en una compra o en un subconjunto de esta, que guardan alguna relación entre sí.

Batch processing. Método utilizado para procesamiento de datos de manera automática o en segmentos para optimizar los recursos computacionales disponibles.

Clustering. Método utilizado para agrupar datos u objetos en grupos de acuerdo con su similitud, de tal manera que compartan atributos o características comunes dentro de cada grupo al que pertenecen.

Collaborative filtering. Técnica utilizada en sistemas de recomendaciones, basada en identificar los ítems de interés para un usuario de acuerdo con las reacciones de otros usuarios similares.

Content-based filtering. Método que considera las características de los ítems para recomendar otros similares a los usuarios, analizando sus elecciones previas conforme a sus preferencias.

Conversion rate. Porcentaje de usuarios que realizaron una acción determinada.

Convolutional Neural Network (CNN). Modelo de *deep learning* basado en una red neuronal especializada en procesar datos estructurados en arreglos, aprendiendo las características de los atributos a través de filtros.

Cross-selling. Estrategia de venta que consiste en sugerir a los clientes artículos adicionales o complementarios a los que desea adquirir.

Dashboard. Herramienta de visualización de datos obtenidos desde un sistema informático a través de resúmenes o gráficos estadísticos.

Data mining. Proceso de analizar y extraer información de grandes volúmenes de datos con el objetivo de identificar patrones o relaciones de interés para el negocio.

Dataloader. Método o clase que permite cargar datos de forma eficiente en un modelo de aprendizaje automático.

Dataset. Conjunto de datos almacenados en o extraídos desde sistemas de información.

Deep learning. Conocido también como aprendizaje profundo, clase de métodos de *machine learning* que utilizan redes neuronales, generalmente con múltiples capas para extracción, aprendizaje e identificación de características de los datos.

Downsampling. Proceso para reducir el tamaño de una muestra de datos.

Dropout. En redes neuronales, técnica que consiste en descartar un número determinado de nodos o neuronas en las capas de la red para evitar el sobreajuste de resultados.

Early stopping. Técnica utilizada para evaluar la capacidad de generalizar de un modelo y detener su entrenamiento cuando esta capacidad tiende a decrecer.

Embedding. Técnica de *natural language processing* para representar palabras en forma de vectores matemáticos.

Feature engineering. Manipulación de los datos de acuerdo con sus características para mejorar el entrenamiento y predicciones de modelos de *machine learning*.

Feature transformation. Técnica de *machine learning* utilizada para transformar una determinada característica a través de una función matemática, con el fin de mejorar el rendimiento de un modelo.

Global Average Pooling. Operación de *pooling* que sustituye capas totalmente conectadas (*fully connected*) de las CNNs, calculando una salida promedio de cada mapa de características, reduciéndolo a un solo valor para cada categoría correspondiente a la tarea de clasificación que se desea ejecutar.

Graph Neural Network (GNN). Tipo de red neuronal artificial en la cual los datos a procesar están representados como grafos.

Grafo. Tipo de datos abstracto compuesto por nodos o vértices y las relaciones entre estos nodos, llamadas bordes o aristas.

Graph Attention Network (GAT). Arquitectura de red neuronal de grafos que se basa en aprender la importancia de la relación entre nodos especificando diferentes pesos para diferentes nodos vecinos.

Graph classification. Tarea de clasificación utilizada en datos estructurados como grafos, en la cual se busca la clase a la que pertenece un grafo en particular.

Graph Convolutional Networks (GCN). Variante de redes neuronales convolucionales utilizada para datos representados en grafos, que aprende representaciones en las capas ocultas (*hidden layers*) que codifican la estructura del grafos y atributos de sus nodos.

Hidden layer. Capa intermedia entre las capas de entrada y salida en una arquitectura de red neuronal, donde las neuronas que la componen utilizan pesos como entradas y producen salidas de acuerdo con una función de activación.

Inductive learning. Técnica de aprendizaje donde se observan casos de entrenamiento para establecer reglas generales, aplicables luego a datos de prueba.

Interquartile Range (IQR). Medida de dispersión de los datos, basada en la división de los datos en cuartiles, que comprende el rango entre el primer y el tercer cuartil.

Itemset. Conjunto de artículos que forman parte de una cesta (*basket*) de compra.

Job. Tarea automática programada en un sistema informático o motor de base de datos.

K-means. Algoritmo utilizado para particionar los datos en k grupos (*clusters*), con cada observación perteneciente al *cluster* con la media más cercana.

L2 regularization. Conocida también como *Ridge regression, Ridge regularization*, o *Weight decay*, técnica que agrega un factor de penalización a la suma de los cuadrados de los pesos en la función de pérdida de una red neuronal, para prevenir el sobreajuste.

LeakyReLU. Función de activación utilizada en redes neuronales, similar a la función ReLU, pero con una pendiente pequeña para valores negativos en lugar de una pendiente cero, ocasionando que la función no se detenga al evaluar valores negativos.

Learning rate. Hiperparámetro de ajuste en un algoritmo de optimización que perite determinar qué tanto se actualizan los pesos en cada iteración y controlar cuánto debe cambiar el modelo en respuesta al error estimado.

Link prediction. Tarea en análisis de redes y grafos que consiste en predecir enlaces faltantes o conexiones futuras entre dos nodos.

Machine learning. Campo de la inteligencia artificial que estudia los métodos y técnicas necesarias para que las computadoras identifiquen patrones en datos y elaboren predicciones.

Message passing. Intercambio de mensajes entre programas o nodos de una red.

Multi-Head Attention. Módulo de capas paralelas para mecanismos de atención en el que se procesan independientemente y a la vez las secuencias de entrada y de salida.

Natural Language Processing (NLP). Tecnología de Inteligencia artificial que permite a las computadoras interpretar el lenguaje humano, a través de algoritmos de *machine* y *deep learning*.

Neighborhood aggregation. Técnica utilizada en redes neuronales de grafos donde la representación matemática de un nodo se actualiza a través de las representaciones de sus nodos vecinos.

Node classification. Tarea comúnmente ejecutada en modelos de *machine learning* basados en grafos, que consiste en asignar etiquetas a los grafos de acuerdo con sus propiedades o relaciones.

One-hot encoding. Técnica mediante la cual se convierten las variables categóricas a conjuntos de números expresados como *bits*, donde solo un valor es '1' y el resto son ceros, para mejorar el rendimiento de algoritmos de *machine learning*.

Outlier. Conocido también como valor atípico, se refiere a una observación cuyo valor es numéricamente distante, muy por encima o debajo de la mayoría de las observaciones.

Overfitting. En aprendizaje automático, se produce cuando el modelo se ajusta exactamente a los datos observados durante el entrenamiento, producto de un excesivo proceso de este tipo, ocasionando que el algoritmo pierda capacidad de generalización en las predicciones.

Pipeline. Conjunto de procesos, entidades o actividades que intervienen organizada o secuencialmente en la recopilación, tratamiento, análisis y presentación de resultados de un proyecto determinado.

Point-of-Interest (POI) Recommendation. Recomendaciones generadas con el objetivo que el usuario visite lugares nuevos basándose en el historial de sus consumos o actividades previas.

Pooling. Técnicas aplicadas en *deep learning* que buscan reducir o simplificar la representación de atributos en mapas de características.

Rating. En sistemas de recomendaciones, valor o categoría que representa la retroalimentación o *feedback* recibido de los usuarios de acuerdo con alguna opinión o preferencia.

Relational Database Management System (RDBMS). O Sistema de administración de bases de datos relacionales, se utiliza para crear, mantener y gestionar bases de datos relacionales, compuestas por tablas a las que se accede por medio de sentencias SQL.

Receiver Operating Characteristic Curve (ROC Curve). Gráfico que muestra el rendimiento de un algoritmo clasificador a través de sus diferentes umbrales de discriminación.

Rectified Lineal Unit (ReLU). Función de activación utilizada en redes neuronales artificiales, que transmite la información generada por la combinación lineal de los pesos y entradas a través de las neuronas.

RFM score. Valor numérico resultante de un análisis RFM o similar que se asigna a cada cliente y permite establecer su categoría o importancia para el negocio.

Recurrent Neural Network (RNN). Modelo de *deep learning* que recibe y procesa datos de entrada secuenciales, como series de tiempo o secuencias de palabras y voz, transformándolos en una salida secuencial específica.

Spasity. En *machine learning*, se refiere a un conjunto de datos, secuencia de números o matrices con un alto número de valores en cero, los cuales tendrán poca o ninguna relevancia en los cálculos o inferencias.

Structured Query Language (SQL). O Lenguaje de consulta estructurada, es un lenguaje de programación con sentencias especiales para extraer o manipular datos de una base de datos relacional.

Stochastic gradient descent SGD. Algoritmo iterativo de optimización, el cual empieza en un punto aleatorio de la función y va bajando hasta alcanzar el punto más bajo de la misma.

Stock Keeping Unit SKU (SKU). Código de referencia utilizado para identificar o mantener un artículo dentro de una tienda o en el sistema que esta utilice.

Stored procedure. Conjunto de sentencias SQL agrupadas en forma de un programa con un propósito específico, almacenado en un motor relacional de base de datos.

Task scheduler. Componente del sistema operativo Microsoft Windows que permite ejecutar programas automáticamente de acuerdo con una programación definida.

Transductive learning. Técnicas de *machine learning* en las que se observan todos los datos de antemano durante el entrenamiento del modelo, aplicando los resultados a casos de prueba específicos.

Upgrade. Mejora o actualización aplicable a un determinado hardware o software.

Upsampling. Método que se usa para aumentar el tamaño de la muestra u observaciones en un conjunto de datos para análisis, generalmente cuando existen pocas observaciones de una clase en particular.

Upselling. Táctica de venta que consiste en persuadir al cliente para que realice la compra de un producto o servicio de un valor superior al solicitado originalmente.

Validación Offline. Técnica de evaluación de sistemas de recomendaciones que consiste en utilizar datos históricos para estimar las reacciones de los usuarios a partir de un punto determinado, basándose en su comportamiento posterior al mismo.

Validación Online. Método de validación de sistemas de recomendaciones que consiste en seleccionar una muestra aleatoria de usuarios para analizar sus datos en un entorno real de interacción.

Weight. En una red neuronal artificial, es el valor numérico que representa la intensidad y dirección de la influencia que ejerce una neurona sobre otra.

BIBLIOGRAFÍA

[1] J. Han, M. Kamber y J. Pei, «Mining Frequent Patterns, Associations, and Correlations: Basic Concepts and Methods,» de *Data mining: concepts and techniques*, Tercera ed., Waltham, MA: Elsevier, 2012, p. 244.

[2] R. Shen, «Prediction Method of User Behavior Label Based on the BP Neural Network,» *Scientific Programming,* vol. 2022, nº 7241956.

[3] A. Esmael y L. Adane, «Book Recommendation Using Collaborative Filtering Algorithm,» *Applied Computational Intelligence and Soft Computing,* vol. 2023, nº 1514801.

[4] E. Yıldız, C. Güngör Şen and E. Işık, "A Hyper-Personalized Product Recommendation System Focused on Customer Segmentation: An Application in the Fashion Retail Industry," *J. Theor. Appl. Electron. Commer. Res.,* no. 18, pp. 571-596, 2023.

[5] Z. Shao, S. Wang, Q. Zhang, W. Lu, Z. Li y X. Peng, «A systematical evaluation for next-basket recommendation algorithms,» de *2022 IEEE 9th International Conference on Data Science and Advanced Analytics DSAA'2022*, Shenzhen, China, 2022.

[6] D. Le, H. Lauw y Y. Fang, «Basket-Sensitive Personalized Item Recommendation,» de *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, 2017.

[7] C. Gao, Y. Zheng, N. Li, Y. Li, Y. Qin, J. Piao, Y. Quan, J. Chang, D. Jin, X. He y Y. Li, «A Survey of Graph Neural Networks for Recommender Systems: Challenges, Methods, and Directions,» *ACM Trans. Recomm. Syst.,* vol. 1, nº 1, pp. 1-51, 2023.

[8] D. Jannach, P. Pu, F. Ricci y M. Zanker, «Recommender systems: Past, present, future,» *AI Magazine,* nº 42, p. 3–6, 2021.

[9] A. Rytt, G. Rodríguez y F. Young, «Investigación e implementación de sistemas de recomendación para e–commerce de ropa,» Universidad ORT Uruguay, Facultad de Ingeniería, 2021.

[10] S. Kangas, «Collaborative filtering and recommendation systems,» *VTT information technology,* pp. 11-20, 2002.

[11] A. Grover y J. Leskovec, «node2vec: Scalable feature learning for networks,» de *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, August 2016.

[12] M. Lin, Q. Chen y S. Yan, «Network in network,» arXiv preprint arXiv:1312.4400, 2013.

[13] R. Ait daoud, A. Amine, B. Belaid y R. Lbibb, «Clustering Prediction Techniques in Defining and Predicting Customers Defection: The Case of E-Commerce Context,» *International Journal of Electrical and Computer Engineering,* vol. 8, pp. 2367-2383, 2018.

[14] H.-C. Chang y H.-P. Tsai, «Group RFM analysis as a novel framework to discover better customer consumption behavior,» *Expert Syst. Appl.,* nº 38, pp. 14499-14513, 2011.

[15] P. Ozkan y I. Deveci-Kocakoc, «A customer segmentation model proposal for retailers: RFM-V,» *Advances in global services and retail management,* pp. 1-12, 2021.

[16] M. Li, X. Bao, L. Chang y G. Tianlong, «Modeling personalized representation for within-basket recommendation based on deep learning,» *Expert Systems with Applications,* vol. 192, nº 116383, 2022.

[17] Z. Liu, L. Xiaohan, F. Ziwei, S. Guo, K. Achan y P. Yu, «Basket Recommendation with Multi-Intent Translation Graph Neural Network,» de *IEEE International Conference on Big Data (Big Data)*, Atlanta, GA, USA, 2020.

[18] R. M. Bell y Y. Koren, «Lessons from the Netflix Prize Challenge,» *Acm Sigkdd Explorations Newsletter,* vol. 9, nº 2, p. 75–79, 2007.

[19] A. Jain, I. Liu, A. Sarda y P. Molino, «Food Discovery with Uber Eats: Using Graph Learning to Power Recommendations,» https://www.uber.com/en-EC/blog/uber-eats-graph-learning/, 2019.

[20] S. Li, «A Gentle Introduction on Market Basket Analysis – Association Rules,» Towards Data Science. https://towardsdatascience.com/a-gentle-introduction-on-market-basket-analysis-association-rules-fa4b986a40ce, 2017.

[21] Interactive Chaos, «Descomposición en Valores Singulares (SVD),» https://interactivechaos.com/es/wiki/descomposicion-en-valores-singulares-svd.

[22] T. N. Kipf y M. Welling, «Semi-Supervised Classification with Graph Convolutional Networks,» doi: 10.48550/arXiv.1609.02907, 2017.

[23] C. Morris, M. Ritzert, M. Fey, W. L. Hamilton, J. E. Lenssen, G. Rattan y M. Grohe, «Weisfeiler and Leman Go Neural: Higher-order Graph Neural Networks,» arXiv e-prints, 2018.

[24] M. Defferrard, X. Bresson y P. Vandergheynst, «Convolutional neural networks on graphs with fast localized spectral filtering,» de *30th Conference on Neural Information Processing Systems (NIPS 2016)*, Barcelona, Spain, 2016.

[25] P. Veličković, G. Cucurull, A. Casanova y A. Romero, «Graph attention networks,» arXiv preprint arXiv:1710.10903, 2017.

[26] X. He, L. Liao, H. Zhang, L. Nie, X. Hu y T. S. Chua, «Neural collaborative filtering,» de *Proceedings of the 26th international conference on world wide web*, 2017.

[27] V. Fionda y G. Pirrò, «Triple2Vec: Learning Triple Embeddings from Knowledge Graphs,» 2019.

[28] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang y M. Wang, «LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation,» de *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20)*, New York, NY, USA, 2020.

[29] Z. Liu, M. Wan, S. Guo, K. Achan y P. Yu, «BasConv: Aggregating Heterogeneous Interactions for Basket Recommendation with Graph Convolutional Neural Network,» 10.1137/1.9781611976236.8, 2020.

[30] E. Zangerle y C. Bauer, «Evaluating Recommender Systems: Survey and Framework,» *ACM Comput,* vol. 55, nº 8, 2023.

[31] Z. a. Huang, Y. Sang, Y. Sun y L. Jiancheng, «A neural network learning algorithm for highly imbalanced data classification,» *Information Sciences,* vol. 612, pp. 496-513, 2022.

[32] W. Hamilton, Z. Ying y J. Leskovec, «Inductive representation learning on large graphs,» *Advances in neural information processing systems,* vol. 30, 2017.

[33] Pinterest Engineering, «PinSage: A new graph convolutional neural network for web-scale recommender systems,» 2018. [En línea]. Available: https://medium.com/pinterest-engineering/pinsage-a-new-graph-convolutional-neural-network-for-web-scale-recommender-systems-88795a107f48.

[34] S. Sheikh, «Exploring SageConv: A Powerful Graph Neural Network Architecture,» 2023. [En línea]. Available: https://medium.com/@sheikh.sahil12299/exploring-sageconv-a-powerful-graph-neural-network-architecture-44b7974b1fe0.

[35] F. Ortega, A. Hernando y J. Bobadilla, «Extended Precision Quality Measure for Recommender Systems,» *Advances in Artificial Intelligence,* vol. 7023, 2011.

[36] s. Moghaddam, N. Abdolvand y S. Rajaei Harandi, «A RFMV model and customer segmentation based on variety of products,» *Journal of Information Systems and Telecommunication,* vol. 5, pp. 155-161, 2017.

[37] F. Isinkaye, Y. Folajimi y B. Ojokoh, «Recommendation systems: Principles, methods and evaluation,» *Egyptian Informatics Journal,* vol. 16, nº 3, pp. 261-273, 2015.

[38] P. Castells y A. Moffat, «Offline recommender system evaluation: Challenges and new directions,» *AI Magazine,* vol. 43, nº 225–38, 2022.

[39] R. Cañamares, P. Castells y A. Moffat, «Offline evaluation options for recommender systems,» *Information Retrieval Journal,* vol. 23.4, pp. 387-410, 2020.

[40] D. Jannach y M. Jugovac, «Measuring the Business Value of Recommender Systems,» *ACM Trans. Manag. Inform. Syst.,* vol. 10, nº 4, 2019.