PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

FACULTAD DE HÁBITAT, INFRAESTRUCTURA Y CREATIVIDAD

SISTEMAS DE INFORMACIÓN



TEMA:

Prototipo de una herramienta de visualización interactiva con asistente conversacional para la toma de decisiones comerciales en PYMES: Caso de estudio Arte Color.

AUTOR: Alisson Verónica Maldonado Pazmiño

QUITO DM, JULIO DE 2025

DEDICATORIA

A todas las personas que han sido parte fundamental de mi crecimiento personal y profesional. A mis padres, cuyo amor, sacrificio y guía han sido el pilar de mi formación, brindándome la fortaleza para enfrentar cada desafío. A mis abuelos y tíos, por su apoyo incondicional, sus enseñanzas y su ejemplo de vida, que han dejado una huella imborrable en mi camino. A mi hermano, mi compañero de vida, cuya determinación y fortaleza me inspiran a superarme cada día. A mis amigas, quienes se convirtieron en mi segunda familia, compartiendo alegrías, desafíos y aprendizajes que enriquecieron esta travesía.

A todos ustedes, con gratitud y cariño, dedico mi trabajo de titulación.

AGRADECIMIENTO

A quienes hicieron posible la culminación de mi trabajo de titulación y han sido pilares esenciales en mi formación. A mis padres, por su amor incondicional, su esfuerzo y su fe en mí, que han sido el motor que me impulsa a seguir adelante. A mi familia por su apoyo constante y sus valiosas enseñanzas, que me han guiado en cada paso. A mi tutor, Dr. Charles Edisson Escobar Terán. A cada docente que, con paciencia y dedicación, han contribuido en mi formación, impulsándome a crecer profesionalmente. A la empresa Arte Color, por brindarme un espacio de aprendizaje durante el desarrollo de este proyecto, lo cual fue fundamental para contextualizar y aplicar este trabajo en un entorno real. A todos quienes, de diversas formas, me brindaron su respaldo, inspiración y aliento, contribuyendo significativamente a este logro.

RESUMEN

Este trabajo presenta un prototipo de herramienta de visualización interactiva con asistente conversacional inteligente, enfocado en mejorar la toma de decisiones comerciales en la empresa Arte Color, una PYME ecuatoriana dedicada a la comercialización de productos de acabados para la construcción. La propuesta parte de la necesidad de transformar los datos históricos de ventas en información estratégica que permita identificar patrones, evaluar el desempeño comercial y apoyar decisiones basadas en evidencia.

La herramienta permite realizar consultas en lenguaje natural sobre un conjunto de datos estructurado, facilitando el acceso a indicadores clave como productos más vendidos, clientes frecuentes, evolución de ingresos, comparativas por sucursal y análisis de marcas. Para ello, se utilizaron modelos de lenguaje generativo (LLMs) que procesan preguntas realizadas por el usuario y entregan respuestas precisas, alineadas con las necesidades de análisis comercial.

El componente visual se basa en un dashboard interactivo construido con bibliotecas de Python como Streamlit y Plotly, los cuales permiten explorar los datos desde múltiples perspectivas. Por otro lado, el desempeño del asistente conversacional fue evaluado mediante la métrica F1 Score, aplicada sobre un conjunto de preguntas base y sus variaciones, lo que permitió medir la precisión y consistencia de las respuestas generadas por los modelos de lenguaje.

Este enfoque demuestra cómo la combinación de inteligencia artificial y visualización de datos puede contribuir significativamente a la toma de decisiones en pequeñas y medianas empresas. El proyecto ofrece una solución accesible, práctica y adaptable, sin requerir conocimientos técnicos avanzados por parte del usuario final. Finalmente, representa una base sólida para futuras aplicaciones que integren capacidades predictivas, automatización de decisiones o conexión con fuentes de datos sincronizadas en el momento.

TABLA DE CONTENIDOS

CAPÍTULO I 11
1. INTRODUCCIÓN 11
1.1 Justificación 11
1.2 PLANTEAMIENTO DEL PROBLEMA 12
1.2.1 Problema principal 13
1.2.2 Problemas secundarios 13
1.3 OBJETIVOS 14
1.3.1 Objetivo General 14
1.3.2 Objetivos Específicos 14
1.4 ALCANCE 14
1.5 Limitaciones 15
CAPÍTULO II 16
2 MARCO TEÓRICO Y CONCEPTUAL 16
2.1 Asistentes conversacionales en entornos empresariales 16
2.2 Procesamiento de lenguaje natural (PLN) 17
2.3 Modelos de lenguaje de gran escala (LLM) 18
2.4 Evaluación del LLM 18
2.4.1 Métrica F1 Score 19
2.5 Visualización de datos para la toma de decisiones 19
2.6 Buenas prácticas para diseño de dashboards interactivos 20
2.6.1 Teoría del color 20
2.6.2 Ubicación de gráficos 20
2.6.3 Interacción dinámica 20
2.7 Indicadores clave de desempeño (KPI) en ventas 21
2.8 Metodología CRISP-DM 21
2.9 HERRAMIENTAS TECNOLÓGICAS 22
2.9.1 Python 22
2.9.2 Streamlit 23
2.9.3 Google Colab 23

2.9.4 Stack AI 23
2.9.5 GitHub 23
CAPÍTULO III 24
3 METODOLOGÍA 24
3.1 Investigación Científica 24
3.2 Investigación Técnica 25
3.2.1 Fases del Proceso del Prototipo 25
3.2.2 Visualización de datos 26
CAPÍTULO IV 27
4 PROTOTIPO: HERRAMIENTA VISUAL INTERACTIVA CON ASISTENTE CONVERSACIONAL INTELIGENTE 27
4.1 PREPARACIÓN DE LOS DATOS 27
4.1.1 Comprensión del negocio 28
4.1.2 Comprensión de los Datos 30
4.1.3 Preparación adecuada de los datos 35
4.2 MODELADO DEL ASISTENTE CONVERSACIONAL 45
4.2.1 Selección de los Modelos de lenguaje (LLM) 45
4.2.2 Configuración de Stack AI 47
4.2.3 Definición del Prompt e instrucciones de respuesta 48
4.2.4 Interacción del Usuario con el asistente 51
4.3 EVALUACIÓN DE LOS MODELOS APLICADOS AL ASISTENTE CONVERSACIONAL 55
4.3.1 Modelos de Lenguaje Evaluados 55
4.3.2 Configuración de hiperparámetros 56
4.3.3 Preguntas Base para la Evaluación 56
4.3.4 Generación de respuestas esperadas (Ground Truth) 58
4.3.5 Generación de respuestas con Stack AI 60
4.3.6 Métrica de evaluación seleccionada: F1 Score 60
4.3.7 Evaluación Automática de los LLM 61
4.3.8 Resultados de la evaluación: 63
4.3.9 Análisis de resultados 64
4.4 DISEÑO DE LA CAPA DE VISUALIZACIÓN INTERACTIVA PARA LA TOMA DE DECISIONES EN ARTE COLOR 66

4.4.1 Herramienta Seleccionada para la Capa de Visualización 66
4.4.2 Selección de indicadores clave (KPI) 67
4.4.3 Diseño de la interfaz 67
4.4.4 Visualizaciones implementadas 69
4.4.5 Flujo de construcción del prototipo de visualización 73
CAPITULO V 76
5 CONCLUSIONES Y RECOMENDACIONES 76
5.1 CONCLUSIONES 76
5.2 RECOMENDACIONES 78
MATERIAL DE REFERENCIA 79
ANEXOS 81

ÍNDICE DE TABLAS

Tabla 1 Comparativa entre GPT-4o, Claude 3.5 Sonnet y Gemini 1.5 Pro en criterios clave. 46
Tabla 2 Selección de preguntas base para la evaluación 57
Tabla 3 Evaluación de los modelos de lenguaje grande 63

ÍNDICE DE FIGURAS

Figura 1 Archivo csv inicial extraído del sistema de facturación de Arte Color 31
Figura 2 Distribución de registros de ventas por año. 33
Figura 3 Distribución de registros por sucursal 34
Figura 4 Distribución de productos más vendidos 34
Figura 5 Dataset de ventas inicial unificado 37
Figura 6 Fragmento de código en Python para renombrar y seleccionar columnas 37
Figura 7 Dataset de columnas estandarizadas 37
Figura 8 Fragmento de código en Python para conversión 38
Figura 9 Fragmento de código en Python para eliminación datos inválidos 40
Figura 10 Eliminación de espacios en blanco 41
Figura 11 Fragmento de código en Python para validación de rangos y eliminación de valores atípicos 42
Figura 12 Fragmento de código en Python para la asignación de las marcas en el dataset 43
Figura 13 Fragmento de código en Python para creación de columnas: marcas, sucursales .. 43
Figura 14 Creación de ID Únicos 44
Figura 15 Creación de columnas temporales para el análisis de tendencias comerciales 44
Figura 16 Vista previa del archivo final de ventas tras la preparación del dataset 45
Figura 17 Creación del chat en Stack AI 47
Figura 18 Instrucciones de respuesta para el LLM 49
Figura 19 Configuración de instrucciones conversacionales en Stack AI para el asistente de Arte Color 51
Figura 20 Flujo de consulta en Stack AI 52
Figura 21 Ejemplo de consulta y respuesta generada por Claude 3.5 Sonnet 53
Figura 22 Ejemplo de consulta y respuesta generada por GPT-4o 54
Figura 23 Ejemplo de consulta y respuesta generada por Gemini 1.5 Pro 54
Figura 24 Fragmento de código en Python para la generación de respuestas esperadas 59

Figura 25 Tokenización de respuestas y cálculo de la métrica F1 Score para comparar las respuestas generadas por los modelos con las respuestas esperadas. 62
Figura 26 Comparación del rendimiento promedio (F1 Score) por modelo y temperatura: ... 64
Figura 27 Interfaz del asistente conversacional ArteColor BOT integrado en el prototipo 69
Figura 28 Evolución de las ventas totales por mes 70
Figura 29 Evolución de las ventas totales..... 71
Figura 30 Clientes con mayor monto comprado 71
Figura 31 Clientes más frecuentes por cantidad comprada 72
Figura 32 Top 10 productos por cantidad vendida..... 72
Figura 33 Top de 10 productos por valor de venta 73
Figura 34 Estructura del repositorio de la aplicación Arte Color en GitHub 74
Figura 35 Dashboard Comercial de Arte Color filtrado por el mes de enero del 2024..... 75

CAPÍTULO I

1. INTRODUCCIÓN

TEMA:

Prototipo de una herramienta de visualización interactiva con asistente conversacional para la toma de decisiones comerciales en PYMES: Caso de estudio Arte Color.

1.1 Justificación

En un entorno empresarial cada vez más digitalizado y competitivo, la capacidad de tomar decisiones basadas en datos se ha convertido en un factor determinante para el éxito organizacional (Salhab, 2024). Las herramientas de visualización de datos y la inteligencia artificial (IA) emergen como soluciones clave para mejorar la toma de decisiones en las PYMES (Barón Ramírez, Barón, & Chacón, 2021).

Considerando que las PYMES son un pilar clave en la economía por su aporte al empleo y al Producto Interno Bruto (PIB), resulta relevante implementar herramientas que fortalezcan su sostenibilidad y crecimiento. Sin embargo, estas pequeñas y medianas empresas enfrentan desafíos significativos para adoptar estas tecnologías, debido a limitaciones de recursos y conocimientos técnicos, lo que dificulta la optimización de su gestión y competitividad (Viteri-Cevallos & Murillo-Párraga, 2021).

La empresa Arte Color, dedicada a la comercialización de productos de acabados de construcción, ejemplifica esta situación. A pesar de contar con datos históricos de ventas, la toma de decisiones comerciales se basa principalmente en la experiencia e intuición de sus administradores, sin el respaldo de herramientas analíticas que transformen estos datos en información útil y estructurada.

Ante esta necesidad, el presente proyecto propone un prototipo de herramienta de visualización interactiva con un asistente conversacional, que permita a Arte Color analizar sus datos históricos de ventas y mejorar la toma de decisiones comerciales.

Con esta propuesta, se espera que Arte Color obtenga una visión más estructurada y analítica de su desempeño comercial, facilitando decisiones más informadas, oportunas y estratégicas. Asimismo, la inclusión de un asistente conversacional permitirá una interacción más ágil y flexible con los datos, mediante consultas en lenguaje natural que agilicen el acceso a información clave.

1.2 PLANTEAMIENTO DEL PROBLEMA

En el contexto actual, caracterizado por la digitalización y la necesidad de tomar decisiones rápidas y basadas en datos, muchas pequeñas y medianas empresas (PYMES) enfrentan desafíos significativos para acceder y analizar su propia información comercial de forma eficiente. A pesar de contar con sistemas de facturación o archivos con historial de ventas, gran parte de estas empresas aún toman decisiones basadas en la intuición, la experiencia previa o en datos poco estructurados.

Este es el caso de Arte Color, una empresa familiar dedicada a la venta de productos de acabados de construcción, la cual ha crecido en los últimos años y actualmente opera en tres sucursales. Sin embargo, sus decisiones comerciales como qué productos promover, qué clientes fidelizar o qué marcas priorizar se fundamentan principalmente en la experiencia de sus administradores, sin el respaldo de herramientas analíticas que transformen los datos disponibles en información clara, estructurada y útil.

La empresa cuenta con información histórica proveniente de su sistema de facturación. No obstante, esta información no es transformada en conocimiento útil para la toma de decisiones, lo que dificulta la identificación de patrones de consumo, productos más rentables o diferencias en el rendimiento entre sucursales.

Por otro lado, si bien existen herramientas como Power BI, Google Looker o Tableau, su adopción en empresas pequeñas está limitada por el costo, la curva de aprendizaje o la falta de personal capacitado. Asimismo, el uso de chatbots o asistentes conversacionales, aunque han ganado popularidad, no cubren completamente las necesidades del usuario gerencial, quien requiere una visión integral, visual e interactiva de los datos para tomar decisiones fundamentadas.

Actualmente, la empresa no cuenta con una solución que combine una herramienta visual interactiva con un asistente conversacional, por lo que los gerentes de Arte Color no disponen de un medio que les permita consultar información clave de manera rápida y flexible. Esto los obliga a revisar manualmente registros históricos, dificultando el acceso a respuestas concretas. Consultas como: ¿Cuál fue el cliente con mayores compras en marzo? o ¿Qué producto fue el más vendido el último trimestre? no pueden resolverse de forma ágil ni eficiente con los recursos actuales.

De este análisis se desprenden el siguiente problema principal y los problemas secundarios:

1.2.1 Problema principal

¿Cómo desarrollar un prototipo de herramienta visual interactiva con un asistente conversacional que permita mejorar la toma de decisiones comerciales en Arte Color, a partir del análisis de sus datos históricos de ventas?

1.2.2 Problemas secundarios

- Problema secundario 1: ¿Cómo estructurar y transformar los datos históricos de ventas para facilitar su análisis en un entorno visual e interactivo?
- Problema secundario 2: ¿Cómo entrenar un asistente conversacional que permita interpretar consultas en lenguaje natural sobre los datos visualizados?
- Problema secundario 3: ¿Cómo evaluar la precisión de los modelos del asistente conversacional aplicados a los datos de Arte Color?
- Problema secundario 4: ¿Qué indicadores clave de desempeño deben ser incluidos en el dashboard para apoyar la toma de decisiones comerciales en Arte Color?

1.3 OBJETIVOS

1.3.1 Objetivo General

Desarrollar un prototipo de herramienta visual interactiva con asistente conversacional inteligente, para la toma de decisiones comerciales mediante el análisis de datos históricos de ventas en Arte Color.

1.3.2 Objetivos Específicos

- Preparar los datos de ventas de Arte Color para su uso en los modelos de aprendizaje automático y en el entorno de visualización interactiva.
- Entrenar un asistente conversacional inteligente que permita interpretar consultas en lenguaje natural sobre los datos de ventas de Arte Color.
- Evaluar la precisión de los modelos de entrenamiento del asistente conversacional aplicados a los datos de Arte Color.
- Diseñar una capa de visualización interactiva que muestre los indicadores clave de ventas para apoyar la toma de decisiones comerciales de Arte Color.

1.4 ALCANCE

El proyecto contempla el desarrollo de un prototipo funcional que abarcará las siguientes características:

- Preparación y estructuración de datos históricos de ventas provenientes del sistema de facturación de Arte Color.
- Entrenamiento de los modelos para el asistente conversacional, con el fin de que interprete consultas en lenguaje natural relacionados con los datos de la empresa.
- Evaluación del desempeño de los modelos entrenados, enfocada en medir la precisión de las respuestas del asistente conversacional, con el propósito de seleccionar el más adecuado para el prototipo.

- Diseño y construcción de un tablero interactivo que permita visualizar los indicadores clave de ventas, facilitando el acceso a información relevante para la toma de decisiones.

1.5 Limitaciones

El proyecto presenta las siguientes restricciones:

- El prototipo se limita al análisis de datos internos de Arte Color, específicamente información derivada de su sistema de facturación, sin considerar fuentes externas de mercado o competencia.
- La evaluación del prototipo se enfoca únicamente en la precisión técnica del asistente conversacional, sin validar la utilidad práctica, experiencia de usuario o impacto real en la toma de decisiones de la empresa.
- La capa de visualización permite mostrar indicadores clave y filtrar por año, mes y sucursal, usando datos estructurados. Aunque incorpora gráficos interactivos y KPIs relevantes, no incluye personalización avanzada ni conexión con múltiples fuentes de datos. Aun así, ofrece una solución funcional y accesible para apoyar decisiones comerciales.

CAPÍTULO II

2 MARCO TEÓRICO Y CONCEPTUAL

Este capítulo reúne los fundamentos conceptuales, metodológicos y tecnológicos que respaldan el desarrollo del prototipo de herramienta de visualización interactiva con asistente conversacional para apoyar la toma de decisiones comerciales en la empresa Arte Color. Se abordan los conceptos clave relacionados con visualización de datos, diseño de dashboards, procesamiento de lenguaje natural y evaluación de modelos conversacionales. Asimismo, se describe la metodología CRISP-DM aplicada al tratamiento de los datos, y finalmente, se presentan las herramientas empleadas en el desarrollo del prototipo.

2.1 Asistentes conversacionales en entornos empresariales

Los asistentes conversacionales representan una evolución significativa dentro del campo de la inteligencia artificial aplicada a entornos empresariales, al permitir la interacción entre usuarios y sistemas informáticos mediante el uso de lenguaje natural. A través de modelos conversacionales, estos asistentes simulan un diálogo fluido y humano, lo que facilita el acceso a información estructurada o compleja sin requerir conocimientos técnicos por parte del usuario (Varela-Tapia et al., 2022).

Dentro del ámbito organizacional, los asistentes conversacionales han sido adoptados para optimizar procesos internos, mejorar la atención al cliente y facilitar la toma de decisiones. Según Ramos Díaz et al. (2019) una de sus principales ventajas es la eficiencia en la recuperación de información, ya que permiten realizar consultas directas y obtener respuestas inmediatas desde bases de datos o sistemas empresariales.

Esta funcionalidad resulta particularmente útil para pequeñas y medianas empresas (PYMES), donde los recursos humanos y tecnológicos suelen ser limitados, pero existe una necesidad creciente de herramientas accesibles para consultar indicadores de gestión, ventas, inventario, entre otros.

Además de responder preguntas básicas, los asistentes conversacionales pueden interpretar intenciones del usuario, adaptarse al contexto de la consulta y proporcionar información relevante y personalizada. Esta capacidad de contextualización les permite integrarse con herramientas como sistemas de gestión (ERP, CRM) o plataformas de visualización de datos, actuando como un puente entre el usuario y los sistemas de información

2.2 Procesamiento de lenguaje natural (PLN)

El Procesamiento de Lenguaje Natural (PLN) es una rama de la inteligencia artificial que permite a las computadoras interpretar, procesar y generar lenguaje humano (Fernández et al., 2019). Su aplicación en asistentes conversacionales facilita que los usuarios puedan interactuar en lenguaje cotidiano con los sistemas, formulando preguntas abiertas o específicas sin necesidad de utilizar comandos rígidos.

En el contexto del presente proyecto, el PLN permite que los responsables de Arte Color realicen consultas directas sobre datos de ventas, mejorando la accesibilidad a la información almacenada en el Dashboard interactivo.

Además, el Procesamiento de Lenguaje Natural (PLN) contribuye a reducir la brecha entre los datos estructurados y el lenguaje natural, ya que traduce automáticamente las preguntas de los usuarios en consultas comprensibles para los sistemas. Esta funcionalidad es especialmente valiosa en entornos de pequeñas y medianas empresas, donde el personal operativo o comercial puede no tener formación técnica en bases de datos, pero sí requiere acceso ágil y comprensible a indicadores clave para tomar decisiones informadas.

2.3 Modelos de lenguaje de gran escala (LLM)

Los Modelos de Lenguaje de Gran Escala (Large Language Models, LLM) son sistemas basados en redes neuronales profundas entrenados con grandes volúmenes de datos textuales, que les permiten comprender, generar y responder en lenguaje natural. Estos modelos, utilizan arquitecturas tipo transformer que les permiten identificar patrones complejos y relaciones contextuales en secuencias de texto, generando respuestas coherentes, contextualizadas y adaptadas a distintas tareas lingüísticas (Atkinson-Abutridy, 2023).

En entornos empresariales, los LLM se han consolidado como herramientas clave para automatizar procesos comunicativos, responder consultas frecuentes, redactar contenidos, analizar documentos o asistir en la toma de decisiones mediante respuestas precisas a preguntas específicas. Su capacidad para comprender lenguaje natural los convierte en una solución accesible para usuarios no técnicos, eliminando la necesidad de estructurar consultas mediante comandos rígidos o lenguajes especializados como SQL. Además, permiten personalizar su comportamiento mediante ajustes de temperatura o entrenamiento con ejemplos específicos del dominio (Burgos et al., 2025).

2.4 Evaluación del LLM

La evaluación de LLM es fundamental para garantizar que las respuestas generadas sean precisas, relevantes y confiables. Según Kulkarni et al., 2025 Las capacidades avanzadas de los LLM no garantizan respuestas correctas, pueden producir alucinaciones (información inventada) o respuestas vagas. La evaluación robusta con métricas automatizadas permite medir con objetividad su desempeño en casos de uso específicos, como preguntas sobre ventas o previsiones comerciales.

2.4.1 Métrica F1 Score

El F1 Score equilibra precisión y exhaustividad, penalizando respuestas parcialmente correctas. En el prototipo de Arte Color, se utilizó esta métrica calculando el número de consultas correctamente respondidas (TP), frente a las incorrectas (FP) o faltantes (FN). Un alto F1 refleja un buen entendimiento del modelo sobre las consultas empresariales y asegura que el asistente sea confiable para apoyar decisiones (Kaur & Kaur Sandhu, 2024)

2.5 Visualización de datos para la toma de decisiones

La visualización de datos consiste en representar gráficamente grandes volúmenes de información con el objetivo de facilitar su interpretación y análisis, apoyando así procesos de toma de decisiones informadas (Tacuna Rubio, 2024).

En el entorno de las pequeñas y medianas empresas (PYMES), la utilización de dashboards interactivos permite a los gerentes y responsables comerciales acceder rápidamente a indicadores clave de desempeño (KPI), identificar patrones de ventas y detectar áreas de mejora.

Un dashboard bien diseñado no solo organiza datos, sino que transforma información dispersa en elementos visuales que facilitan la comprensión inmediata, contribuyendo a decisiones más rápidas y fundamentadas (Murillo Noblecilla, 2024). Además, permite detectar oportunidades y riesgos con mayor agilidad, apoyando a los responsables comerciales en la definición de estrategias, asignación de recursos y evaluación de resultados. Su utilidad es especialmente evidente en entornos como las PYMES, donde se requiere tomar decisiones precisas con información accesible y sintetizada.

2.6 Buenas prácticas para diseño de dashboards interactivos

El diseño de dashboards debe seguir principios que optimicen la percepción y comprensión de los datos por parte de los usuarios, según Córdoba Viera, Martínez Borrego y Córdoba Viera (2021) a continuación:

2.6.1 Teoría del color

La selección adecuada de colores facilita la diferenciación de categorías, la percepción de tendencias y la identificación de valores atípicos. Paletas cromáticas armoniosas mejoran la experiencia visual y reducen la fatiga cognitiva

2.6.2 Ubicación de gráficos

Los elementos visuales deben ubicarse siguiendo una jerarquía lógica que guíe la atención del usuario desde los datos más relevantes hacia detalles secundarios. La disposición eficiente facilita una lectura rápida y evita la sobrecarga visual.

2.6.3 Interacción dinámica

Los dashboards interactivos deben permitir a los usuarios filtrar, explorar y consultar información específica mediante mecanismos como menús desplegables, selecciones de periodo de tiempo o integración de consultas en lenguaje natural, aumentando la utilidad y adaptabilidad de la herramienta.

Un dashboard bien estructurado, apoyado en estas buenas prácticas, facilita la toma de decisiones ágiles y estratégicas.

2.7 Indicadores clave de desempeño (KPI) en ventas

Los KPI son métricas cuantificables que reflejan el rendimiento comercial. En PYMES, ayudan a evaluar el progreso y tomar decisiones estratégicas. Indicadores típicos incluyen total de ventas, ticket promedio y el número de transacciones realizadas. En la tesis se seleccionaron los más relevantes para Arte Color, garantizando alineación con sus objetivos organizacionales.

La correcta selección y monitoreo de los KPI permite a los responsables comerciales identificar tendencias de consumo, evaluar la efectividad de campañas promocionales, detectar productos de alta o baja rotación y anticipar cambios en la demanda. Según Yadavalli et al. (2020) explica que los KPI deben estar vinculados a los objetivos estratégicos de la empresa y facilitar una lectura clara del desempeño en áreas clave, lo cual resulta fundamental para mejorar la eficiencia operativa y la competitividad. En el contexto de esta investigación, los KPI funcionan como base tanto para la visualización de datos como para las recomendaciones generadas por el asistente conversacional.

2.8 Metodología CRISP-DM

CRISP-DM (Cross-Industry Standard Process for Data Mining) es una metodología ampliamente utilizada para el desarrollo de proyectos de análisis de datos (Shimaoka et al., 2024).

Consiste en seis fases principales:

1. **Comprensión del negocio:** Identificar los objetivos organizacionales y definir las necesidades de análisis.

2. **Comprensión de los datos:** Explorar y analizar los datos disponibles para evaluar su calidad y relevancia.

3. **Preparación de los datos:** Seleccionar, limpiar y transformar los datos necesarios para el modelado.
4. **Modelado:** Aplicar técnicas analíticas para construir modelos que resuelvan los problemas planteados.
5. **Evaluación:** Validar que los modelos cumplen con los objetivos de negocio.
6. **Despliegue:** Planificar o realizar la entrega de los resultados en el entorno operativo.

En el presente proyecto, CRISP-DM proporciona una estructura clara para preparar los datos históricos de ventas, entrenar el asistente conversacional y construir el dashboard interactivo, asegurando que cada etapa esté alineada con los objetivos de Arte Color.

2.9 HERRAMIENTAS TECNOLÓGICAS

Para la implementación del prototipo de visualización interactiva con asistente conversacional, se seleccionaron herramientas tecnológicas accesibles y especializadas en análisis de datos, visualización y procesamiento de lenguaje natural. Estas herramientas permitieron construir una solución funcional y orientada a la toma de decisiones en el contexto de una pequeña empresa como Arte Color, sin requerir infraestructura compleja ni conocimientos avanzados por parte de los usuarios finales.

2.9.1 Python

Entorno de programación elegido por su versatilidad en ciencia de datos. Se empleó principalmente en tareas como limpieza, alineación e indexación temporal mediante la biblioteca Pandas y para la generación de gráficos interactivos con Plotly. Además, permitió calcular métricas de evaluación como el F1 Score, asegurando la calidad de las respuestas del asistente conversacional.

2.9.2 Streamlit

Plataforma de desarrollo que convierte scripts de Python en apps web interactivas de forma rápida y sencilla, Soporta integración con librerías como Pandas y Plotly para visualizar datos en tiempo real.

2.9.3 Google Colab

Entorno colaborativo alojado en la nube que permite ejecutar y compartir notebooks con código, texto y visualizaciones. Ofrece recursos como GPUs, integración con Google Drive y funciones de edición simultánea.

2.9.4 Stack AI

Plataforma no-code especializada en crear flujos conversacionales con LLMs. Se utilizó para configurar el ciclo: captura de consultas en lenguaje natural, conexión con modelo conversacional, y generación de respuestas basadas en datos de ventas.

2.9.5 GitHub

Servicio de alojamiento de código que permitió publicar el prototipo de dashboard en Streamlit Community Cloud, facilitando su acceso a través de la web mediante un repositorio conectado.

Estas herramientas fueron seleccionadas por su accesibilidad y compatibilidad con el enfoque práctico del proyecto. En conjunto, permitieron desarrollar un prototipo funcional, interactivo y alineado con las necesidades reales de gestión comercial de una PYME como Arte Color.

CAPÍTULO III

3 METODOLOGÍA

3.1 Investigación Científica

La presente investigación se enmarca dentro del enfoque cuantitativo, ya que parte del análisis de datos históricos para diseñar un prototipo de solución tecnológica orientada a la toma de decisiones comerciales en la empresa Arte Color.

Se trata de una investigación aplicada, dado que busca resolver un problema concreto mediante el desarrollo de un prototipo funcional, aunque sin llegar a su implementación definitiva dentro de la organización.

Desde una perspectiva metodológica, la investigación combina un enfoque descriptivo y exploratorio. Es descriptiva porque analiza el proceso actual de gestión de ventas de la empresa, sus datos históricos y su estructura comercial. Es exploratoria porque integra tecnologías emergentes, como dashboards interactivos y asistentes conversacionales, que no han sido previamente utilizadas en Arte Color.

Las etapas consideradas incluyen la preparación y análisis de los datos históricos de ventas, el diseño y construcción de un prototipo que combine visualización interactiva e interacción conversacional, y una evaluación técnica centrada en medir la precisión del asistente conversacional frente a consultas en lenguaje natural.

3.2 Investigación Técnica

Tomando como referencia el modelo CRISP-DM (Cross Industry Standard Process for Data Mining), se han definido las fases que estructuran el proceso de desarrollo del prototipo, centrado en la consulta de datos históricos de ventas a través de un asistente conversacional integrado a un entorno visual. Este enfoque metodológico permite organizar las actividades técnicas desde la definición de necesidades de negocio hasta la validación del modelo.

3.2.1 Fases del Proceso del Prototipo

Las fases del proceso que estructuran el desarrollo del prototipo centrado en datos históricos son:

- **Comprensión del negocio:** Identificación de los objetivos organizacionales de Arte Color y definición de las necesidades de análisis comercial que justifican el desarrollo de una herramienta visual con asistente conversacional.
- **Comprensión de los datos:** Exploración de los datos históricos provenientes del sistema de facturación, evaluación de su calidad, estructura, y relevancia para el análisis posterior.
- **Preparación de los datos:** Incluye la recolección, limpieza y transformación de los datos históricos de ventas, asegurando su consistencia y adecuación para el análisis posterior.
- **Modelado:** Entrenamiento del asistente conversacional utilizando técnicas de procesamiento de lenguaje natural, con base en el dataset estructurado.
- **Evaluación:** Comparación de los resultados obtenidos por los modelos del asistente, midiendo la precisión y la capacidad de respuesta frente a las consultas planteadas, con el fin de seleccionar la mejor alternativa.

3.2.2 Visualización de datos

La visualización de datos en el prototipo se diseñó con base en buenas prácticas de claridad, simplicidad e interactividad, enfocadas en facilitar el análisis comercial para usuarios no técnicos. No se buscó desarrollar componentes visuales avanzados, sino representar de forma clara los indicadores clave mediante gráficos comprensibles y filtros dinámicos.

Se integraron elementos básicos como:

- Uso adecuado del color y disposición visual: Se emplearon colores suaves y una estructura modular que permite identificar fácilmente los principales indicadores sin sobrecargar la interfaz.
- Filtros interactivos: El usuario puede seleccionar año, mes y sucursal, lo que permite ajustar las visualizaciones según su necesidad específica.
- Asistente conversacional integrado: La inteligencia artificial se usa como apoyo para responder preguntas en lenguaje natural sobre los datos de Arte Color, lo cual refuerza la accesibilidad a la información.

El enfoque visual del prototipo se orienta a mejorar la toma de decisiones mediante una visualización clara, sin complejidad técnica, y con herramientas prácticas de exploración de datos adaptadas a las necesidades de una PYME.

CAPÍTULO IV

4 PROTOTIPO: HERRAMIENTA VISUAL INTERACTIVA CON ASISTENTE CONVERSACIONAL INTELIGENTE

Este capítulo presenta el prototipo de una herramienta visual interactiva con asistente conversacional inteligente, orientado a facilitar la toma de decisiones comerciales en la empresa Arte Color a partir del análisis de datos históricos de ventas.

La construcción del prototipo se organizó en cuatro etapas, conforme a los objetivos específicos del proyecto: preparación y limpieza de los datos, entrenamiento del asistente conversacional, evaluación de su precisión, y diseño de la capa de visualización interactiva. Cada sección describe de forma clara el enfoque adoptado, las herramientas empleadas y los principales resultados obtenidos, evidenciando el potencial de este tipo de soluciones para apoyar la gestión comercial en pequeñas y medianas empresas.

4.1 PREPARACIÓN DE LOS DATOS

La etapa de preparación de los datos incluyó previamente el entendimiento del negocio y la comprensión de los datos, tomando como referencia la metodología CRISP-DM. Estas fases permitieron contextualizar el entorno comercial de Arte Color y analizar las características del dataset. Con base en ello, se realizó la preparación de los datos para que puedan ser utilizados en las siguientes etapas del prototipo.

4.1.1 Comprensión del negocio

Arte Color es una empresa ecuatoriana dedicada a la comercialización de productos de acabados para la construcción, tales como pinturas, cerámicas y porcelanatos. Se posiciona como distribuidor de prestigiosas marcas nacionales e importadas, ofreciendo a sus clientes un portafolio diverso de productos de alta calidad. Su propuesta de valor se basa en tres pilares fundamentales:

- Asesoría personalizada: cuenta con expertos en diseño y color que acompañan a los clientes en sus decisiones.
- Precios competitivos: al ser distribuidor directo, puede ofrecer promociones y precios atractivos.
- Calidad y garantía: ofrece productos confiables, respaldados por su trayectoria y atención responsable.

Estos valores guían la operación de la empresa y sustentan su compromiso con el cliente, tanto a nivel comercial como en su atención postventa.

Objetivos del negocio

El objetivo principal de Arte Color es mejorar la eficiencia y calidad de sus decisiones comerciales, particularmente en áreas como planificación de promociones, gestión de precios, selección de productos prioritarios, análisis de desempeño de clientes y sucursales, entre otros. Actualmente, dichas decisiones se basan en la experiencia de los administradores, sin herramientas analíticas estructuradas que aprovechen los datos históricos de ventas disponibles.

Por tanto, el objetivo del negocio en el contexto de este proyecto es:

Fortalecer la toma de decisiones comerciales mediante el uso de herramientas tecnológicas que permitan transformar los datos históricos de ventas en información útil, accesible y accionable.

Evaluación de la situación actual

Arte Color presenta limitaciones significativas en su gestión de información. La empresa carece de una herramienta que permita visualizar sus datos de manera clara e interactiva, así como de mecanismos que faciliten la consulta de información en lenguaje natural. Esto genera una dependencia de los administradores y dificulta la planificación comercial estratégica.

Además, la empresa se enfrenta a los siguientes problemas:

- Falta de estructura en el análisis de datos comerciales.
- Ausencia de planificación promocional basada en análisis de rendimiento.
- Poca sistematización de la información sobre clientes y campañas.
- Dificultades para identificar tendencias, oportunidades de mejora y patrones de consumo.
- Alta dependencia operativa de sus administradores.
- Falta de herramientas analíticas para evaluar productos, clientes y tendencias.
- Promociones sin planificación estructurada, basadas solo en descuentos de proveedores.
- Ausencia de registros organizados de clientes y campañas publicitarias.

Estas condiciones afectan su capacidad para competir con empresas más grandes y limitan su crecimiento sostenido.

Plan preliminar del proyecto

Para alcanzar los objetivos planteados, el proyecto se desarrollará bajo la estructura de la metodología CRISP-DM, considerando las siguientes fases:

- Comprensión del negocio: Análisis del contexto de Arte Color y sus necesidades estratégicas.
- Comprensión de los datos: Exploración de la información disponible en el sistema de facturación.
- Preparación de los datos: Limpieza y transformación de los datos para su análisis.

- Modelado: Entrenamiento de modelos de lenguaje grande para consultas sobre los datos de la empresa.
- Evaluación: Medición de la precisión del asistente en la interpretación de consultas.
- Visualización: Construcción de un dashboard interactivo con integración conversacional.

4.1.2 Comprensión de los Datos

Tras definir las necesidades del negocio, se procedió a explorar la información disponible para el análisis. Arte Color cuenta con registros históricos de ventas provenientes de su sistema de facturación, lo que constituye la base para alimentar tanto al asistente conversacional como al dashboard interactivo.

Recolección de los datos

Para este proyecto se recopilaron datos históricos de ventas provenientes del sistema interno de facturación de la empresa Arte Color, los cuales contenían registros detallados de las ventas realizadas en sus dos sucursales: Matriz (Pinturas) y Sucursal Cerámica. En esta etapa se trabajó principalmente con tres archivos CSV:

- DetalleVentas_Producto_Ceramica.csv
- DetalleVentas_Producto_Matriz.csv

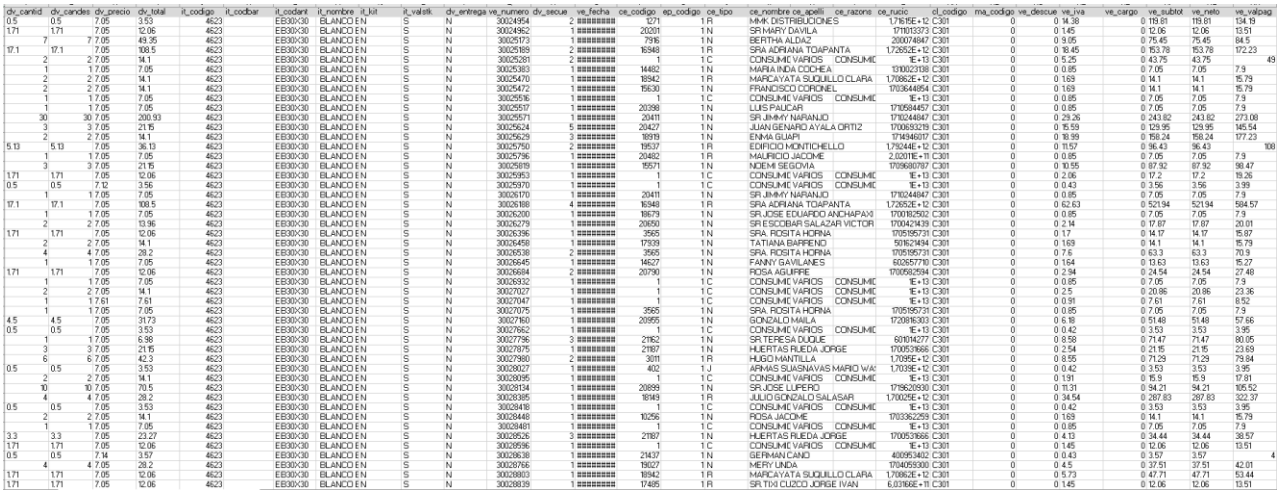
Ambos contienen información detallada de las ventas realizadas en dos sucursales distintas: Cerámica y Matriz.

Estos archivos fueron combinados en un único conjunto de datos con un total de: 122.346 registros (filas) y 34 columnas (variables).

- ResumenVentas_Producto_Marca.csv

Este archivo contiene información sobre las marcas comercializadas. De este conjunto se extrajeron únicamente las marcas únicas (sin duplicados) para luego poder unir o relacionar con el archivo de detalle de ventas, mediante el código de la marca.

Los datos contenidos en los archivos de ventas corresponden al período comprendido entre enero de 2023 y diciembre de 2024, abarcando dos años completos de actividad comercial. Este rango permite analizar comportamientos estacionales, comparar el rendimiento anual y evaluar tendencias relevantes en el tiempo.



dv_cantid	dv_candes	dv_precio	dv_total	it_codigo	it_codbar	it_codant	it_nombre	it_lot	it_valstk	dv_entrega	ve_numero	dv_secue	ve_fecha	ce_codigo	ep_codigo	ce_tipo	ce_nombre	ce_apelli	ce_razons	ce_rucio	cl_codigo	ma_codigo	ve_descue	ve_iva	ve_cargo	ve_subtot	ve_neto	ve_valpag
0.5	0.5	7.05	3.53	4623		EB30X30	BLANCO EN		S	N	30024954	2	########	1271	1	R	MHK DISTRIBUCIONES			1,71E+12	C301	0	0	14.38	0	119.81	119.81	134.19
1.71	1.71	7.05	12.06	4623		EB30X30	BLANCO EN		S	N	30024962	1	########	20401	1	N	SR MARY DAVILA			171810373	C301	0	0	1.45	0	12.06	12.06	13.51
7	7	7.05	49.35	4623		EB30X30	BLANCO EN		S	N	30025173	1	########	7916	1	N	BERTHA ALDAZ			200074647	C301	0	0	9.05	0	75.45	75.45	84.5
17.1	17.1	7.05	108.5	4623		EB30X30	BLANCO EN		S	N	30025189	2	########	16948	1	R	SRA ADRIANA TOAPANTA			1,72652E+12	C301	0	0	18.45	0	153.78	153.78	172.23
2	2	7.05	14.1	4623		EB30X30	BLANCO EN		S	N	30025281	2	########	1	1	C	CONSUMIC VARIOS	CONSUMIC		1E+13	C301	0	0	5.25	0	43.75	43.75	49
1	1	7.05	7.05	4623		EB30X30	BLANCO EN		S	N	30025383	1	########	14482	1	N	MARIA INDA COCHEA			1310023138	C301	0	0	0.85	0	7.05	7.05	7.9
2	2	7.05	14.1	4623		EB30X30	BLANCO EN		S	N	30025470	1	########	16942	1	R	MARCAYATA SUQUILLO CLARA			1,70862E+12	C301	0	0	1.69	0	14.1	14.1	15.79
2	2	7.05	14.1	4623		EB30X30	BLANCO EN		S	N	30025472	1	########	15630	1	N	FRANCISCO CORONEL			1703644854	C301	0	0	1.69	0	14.1	14.1	15.79
1	1	7.05	7.05	4623		EB30X30	BLANCO EN		S	N	30025516	1	########	1	1	C	CONSUMIC VARIOS	CONSUMIC		1E+13	C301	0	0	0.85	0	7.05	7.05	7.9
1	1	7.05	7.05	4623		EB30X30	BLANCO EN		S	N	30025517	1	########	20398	1	N	LUIS PAUCAR			1710584457	C301	0	0	0.85	0	7.05	7.05	7.9
30	30	7.05	200.93	4623		EB30X30	BLANCO EN		S	N	30025571	1	########	20411	1	N	SR JIMMY NARANJO			1716244847	C301	0	0	29.26	0	243.82	243.82	273.08
3	3	7.05	21.15	4623		EB30X30	BLANCO EN		S	N	30025624	5	########	20427	1	N	JUAN GENARO AYALA ORTIZ			1700693219	C301	0	0	15.59	0	129.95	129.95	145.54
2	2	7.05	14.1	4623		EB30X30	BLANCO EN		S	N	30025629	3	########	18919	1	N	ENMA GUAPI			1714946917	C301	0	0	18.99	0	158.24	158.24	177.23
5.13	5.13	7.05	36.13	4623		EB30X30	BLANCO EN		S	N	30025750	2	########	19537	1	R	EDIFICIO MONTICHELLO			1,79244E+12	C301	0	0	11.57	0	96.43	96.43	108
1	1	7.05	7.05	4623		EB30X30	BLANCO EN		S	N	30025796	1	########	20462	1	R	MAURICIO JACOME			2,02011E+11	C301	0	0	0.85	0	7.05	7.05	7.9
3	3	7.05	21.15	4623		EB30X30	BLANCO EN		S	N	30025819	1	########	15571	1	N	NOEMI SEGOVIA			1709680787	C301	0	0	10.55	0	87.92	87.92	98.47
1.71	1.71	7.05	12.06	4623		EB30X30	BLANCO EN		S	N	30025953	1	########	1	1	C	CONSUMIC VARIOS	CONSUMIC		1E+13	C301	0	0	2.06	0	17.2	17.2	19.26
0.5	0.5	7.12	3.56	4623		EB30X30	BLANCO EN		S	N	30025970	1	########	1	1	C	CONSUMIC VARIOS	CONSUMIC		1E+13	C301	0	0	0.43	0	3.56	3.56	3.99
1	1	7.05	7.05	4623		EB30X30	BLANCO EN		S	N	30026170	1	########	20411	1	N	SR JIMMY NARANJO			1716244847	C301	0	0	0.85	0	7.05	7.05	7.9
17.1	17.1	7.05	108.5	4623		EB30X30	BLANCO EN		S	N	30026188	4	########	16948	1	R	SRA ADRIANA TOAPANTA			1,72652E+12	C301	0	0	62.63	0	521.94	521.94	584.57
1	1	7.05	7.05	4623		EB30X30	BLANCO EN		S	N	30026269	1	########	18679	1	N	SR JOSE EDUARDO ANCHAPAXI			1709182592	C301	0	0	0.85	0	7.05	7.05	7.9
2	2	7.05	13.96	4623		EB30X30	BLANCO EN		S	N	30026279	1	########	20650	1	N	SR ESCOBAR SALAZAR VICTOR			1700421439	C301	0	0	2.14	0	17.87	17.87	20.01
1.71	1.71	7.05	12.06	4623		EB30X30	BLANCO EN		S	N	30026396	1	########	3565	1	N	SRA. ROSITA HORNA			1705195731	C301	0	0	1.7	0	14.17	14.17	15.87
2	2	7.05	14.1	4623		EB30X30	BLANCO EN		S	N	30026458	1	########	17939	1	N	TATIANA BARRENO			501627494	C301	0	0	1.69	0	14.1	14.1	15.79
4	4	7.05	28.2	4623		EB30X30	BLANCO EN		S	N	30026538	2	########	3565	1	N	SRA. ROSITA HORNA			1705195731	C301	0	0	7.6	0	63.3	63.3	70.9
1	1	7.05	7.05	4623		EB30X30	BLANCO EN		S	N	30026645	1	########	14627	1	N	FANNY GAVILANES			602657710	C301	0	0	1.64	0	13.63	13.63	15.27
1.71	1.71	7.05	12.06	4623		EB30X30	BLANCO EN		S	N	30026684	2	########	20790	1	N	ROSA AGUIRRE			1705592594	C301	0	0	2.94	0	24.54	24.54	27.48
1	1	7.05	7.05	4623		EB30X30	BLANCO EN		S	N	30026932	1	########	1	1	C	CONSUMIC VARIOS	CONSUMIC		1E+13	C301	0	0	0.85	0	7.05	7.05	7.9
2	2	7.05	14.1	4623		EB30X30	BLANCO EN		S	N	30027027	1	########	1	1	C	CONSUMIC VARIOS	CONSUMIC		1E+13	C301	0	0	2.5	0	20.86	20.86	23.36
1	1	7.61	7.61	4623		EB30X30	BLANCO EN		S	N	30027047	1	########	1	1	C	CONSUMIC VARIOS	CONSUMIC		1E+13	C301	0	0	0.91	0	7.61	7.61	8.52
1	1	7.05	7.05	4623		EB30X30	BLANCO EN		S	N	30027075	1	########	3565	1	N	SRA. ROSITA HORNA			1705195731	C301	0	0	0.85	0	7.05	7.05	7.9
4.5	4.5	7.05	31.73	4623		EB30X30	BLANCO EN		S	N	30027160	1	########	20955	1	N	GONZALO MAILA			1720816303	C301	0	0	6.18	0	51.48	51.48	57.66
0.5	0.5	7.05	3.53	4623		EB30X30	BLANCO EN		S	N	30027662	1	########	1	1	C	CONSUMIC VARIOS	CONSUMIC		1E+13	C301	0	0	0.42	0	3.53	3.53	3.95
1	1	7.05	6.98	4623		EB30X30	BLANCO EN		S	N	30027796	3	########	21162	1	N	SR TERESA DUQUE			601014277	C301	0	0	8.58	0	71.47	71.47	80.05
3	3	7.05	21.15	4623		EB30X30	BLANCO EN		S	N	30027875	1	########	21187	1	N	HUERTAS RUEDA JORGE			1700531666	C301	0	0	2.54	0	21.15	21.15	23.69
6	6	7.05	42.3	4623		EB30X30	BLANCO EN		S	N	30027880	2	########	3011	1	R	HUGO MANTILLA			1,7095E+12	C301	0	0	8.55	0	71.29	71.29	79.84
0.5	0.5	7.05	3.53	4623		EB30X30	BLANCO EN		S	N	30028027	1	########	402	1	J	ARMAS SUASNAVAS MARIO WA			1,7039E+12	C301	0	0	0.42	0	3.53	3.53	3.95
2	2	7.05	14.1	4623		EB30X30	BLANCO EN		S	N	30028095	1	########	1	1	C	CONSUMIC VARIOS	CONSUMIC		1E+13	C301	0	0	1.91	0	15.9	15.9	17.81
10	10	7.05	70.5	4623		EB30X30	BLANCO EN		S	N	30028134	1	########	20899	1	N	SR JOSE LUPERO			1718620930	C301	0	0	11.31	0	94.21	94.21	105.52
4	4	7.05	28.2	4623		EB30X30	BLANCO EN		S	N	30028385	1	########	16949	1	R	JULIO GONZALO SALASAR			1,70029E+12	C301	0	0	34.54	0	287.83	287.83	322.37
0.5	0.5	7.05	3.53	4623		EB30X30	BLANCO EN		S	N	30028418	1	########	1	1	C	CONSUMIC VARIOS	CONSUMIC		1E+13	C301	0	0	0.42	0	3.53	3.53	3.95
2	2	7.05	14.1	4623		EB30X30	BLANCO EN		S	N	30028448	1	########	10256	1	N	ROSA JACOME			1703362259	C301	0	0	1.69	0	14.1	14.1	15.79
1	1	7.05	7.05	4623		EB30X30	BLANCO EN		S	N	30028481	1	########	1	1	C	CONSUMIC VARIOS	CONSUMIC		1E+13	C301	0	0	0.85	0	7.05	7.05	7.9
3.3	3.3	7.05	23.27	4623		EB30X30	BLANCO EN		S	N	30028526	3	########	21187	1	N	HUERTAS RUEDA JORGE			1700531666	C301	0	0	4.13	0	34.44	34.44	38.57
1.71	1.71	7.05	12.06	4623		EB30X30	BLANCO EN		S	N	30028596	1	########	1	1	C	CONSUMIC VARIOS	CONSUMIC		1E+13	C301	0	0	1.45	0	12.06	12.06	13.51
0.5	0.5	7.14	3.57	4623		EB30X30	BLANCO EN		S	N	30028638	1	########	21437	1	N	GERMAN CANO			400953402	C301	0	0	0.43	0	3.57	3.57	4
4	4	7.05	28.2	4623		EB30X30	BLANCO EN		S	N	30028766	1	########	19027	1	N	MERY UNDA			1704059300	C301	0	0	4.5	0	37.51	37.51	42.01
1.71	1.71	7.05	12.06	4623		EB30X30	BLANCO EN		S	N	30028803	1	########	16942	1	R	MARCAYATA SUQUILLO CLARA			1,70862E+12	C301	0	0	5.73	0	47.71	47.71	53.44
1.71	1.71	7.05	12.06	4623		EB30X30	BLANCO EN		S	N	30028839	1	########	17495	1	R	SR TIXI CUZCO JORGE IVAN			6,03166E+11	C301	0	0	1.45	0	12.06	12.06	13.51

Figura 1 *Archivo csv inicial extraído del sistema de facturación de Arte Color*

Descripción de los datos de las columnas seleccionadas

A continuación, se detallan las variables que se seleccionaron de los archivos de detalle de ventas:

- **it_codant:** Código interno del producto vendido. Identifica de forma única cada producto.
- **it_nombre:** Nombre del producto vendido. Puede incluir tipo, presentación o categoría.
- **ma_codigo:** Código de la marca asociada al producto.
- **su_codigo:** Código de la sucursal donde se efectuó la venta.
- **dv_precio:** Precio unitario del producto al momento de la transacción.
- **ve_numero:** Número único que identifica la venta o factura.
- **ve_fecha:** Fecha exacta en que se realizó la venta.
- **ce_codigo:** Código del cliente que realizó la compra.
- **ve_neto:** Valor neto de la venta (sin incluir impuestos).

- **dv_cantid:** Valor total pagado por el cliente (incluyendo impuestos u otros recargos).
- **dv_cantid:** Cantidad de unidades vendidas del producto en esa transacción.
- **dv_total:** Valor total de la línea de detalle del producto (precio x cantidad).
- **ce_nombre:** Nombre del cliente (puede ser persona natural o jurídica).
- **ce_rucic:** Cédula o RUC del cliente. Sirve para identificar personas o empresas.

Este conjunto de datos permite analizar las ventas a nivel de producto, cliente, fecha y sucursal.

Archivo complementario

El archivo: ResumenVentas_Producto_Marca.csv contiene información complementaria sobre las marcas de productos comercializados. Las variables incluidas son:

ma_codigo: Código único de la marca. Permite la relación con el detalle de ventas.

ma_descri: Nombre o descripción comercial de la marca.

De este archivo se extrajeron solo los registros de marcas únicas (sin duplicados), con el fin de estandarizar la relación y posterior unión con los archivos de ventas detalladas a través del campo ma_codigo.

Calidad y estructura de los datos

Durante la revisión inicial del conjunto de datos se identificaron los siguientes puntos relevantes:

- Formato estructurado: Los archivos se encuentran organizados en formato CSV, con columnas consistentes que representan variables comerciales clave como producto, cliente, fecha, sucursal y valores de venta.

Variables temporales: La variable ve_fecha permite organizar las ventas cronológicamente, realizar agrupaciones por mes, trimestre o año, y detectar patrones temporales.

- Verificación de valores faltantes y duplicados: En esta fase se realizó una inspección inicial de calidad. No se evidenciaron valores nulos críticos, aunque se considera una revisión más profunda en la etapa de preparación para garantizar la consistencia del conjunto final.

Exploración inicial

- El conjunto unido abarca más de 120 mil transacciones, lo que permite realizar análisis estadísticos robustos y detectar patrones comerciales.
- Las variables permiten responder preguntas como: ¿Qué productos se venden más por sucursal? ¿Cuál es el cliente que más compra? ¿Qué marcas tienen mayor rotación?
- Existen múltiples campos que pueden utilizarse para segmentaciones, como marca, cliente, fecha, sucursal y producto.
- Se verificó que los datos presentaban ciertas inconsistencias comunes como registros duplicados, formatos inconsistentes y valores faltantes, por lo que necesita de una fase de preparación exhaustiva.
- Dado que el conjunto de datos abarca los años 2023 y 2024, se pueden identificar y comparar periodos de alto y bajo rendimiento.

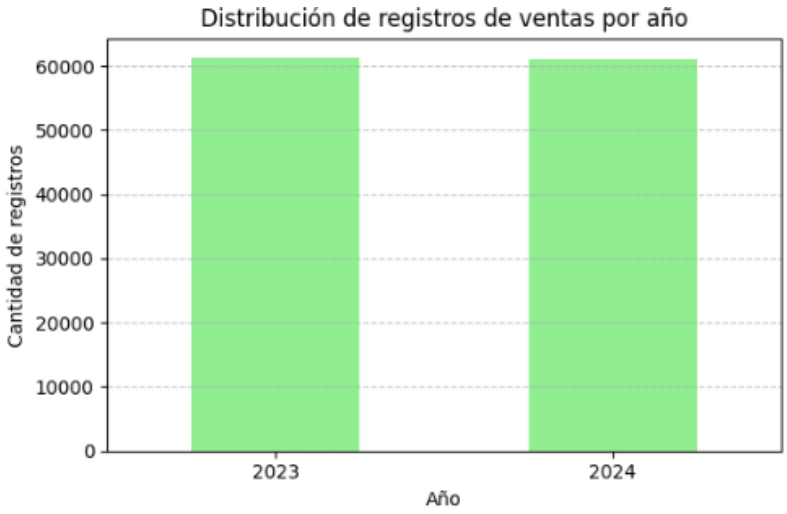


Figura 2 *Distribución de registros de ventas por año*.

El gráfico muestra que los datos están equilibrados entre los años 2023 y 2024, con poco más de 60.000 registros en cada periodo. Esto permite realizar comparaciones anuales confiables, detectar posibles patrones estacionales y cambios de comportamiento de productos y clientes a lo largo del tiempo.

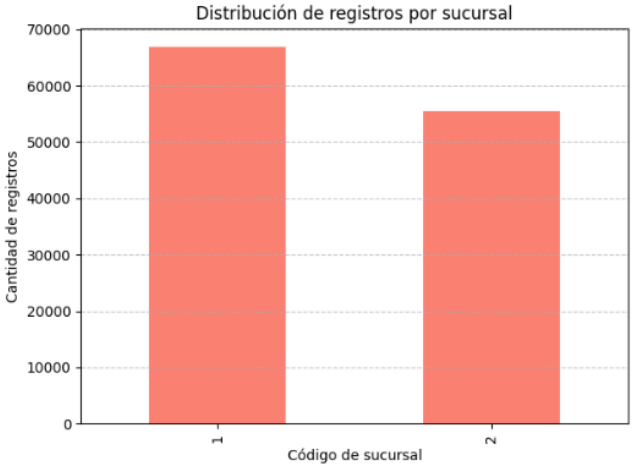


Figura 3 *Distribución de registros por sucursal*

La figura muestra que la sucursal 1 (matriz) concentra una mayor cantidad de registros de ventas en comparación con la sucursal 2 (cerámica), lo que indica un mayor volumen de transacciones en dicho punto durante el período analizado.

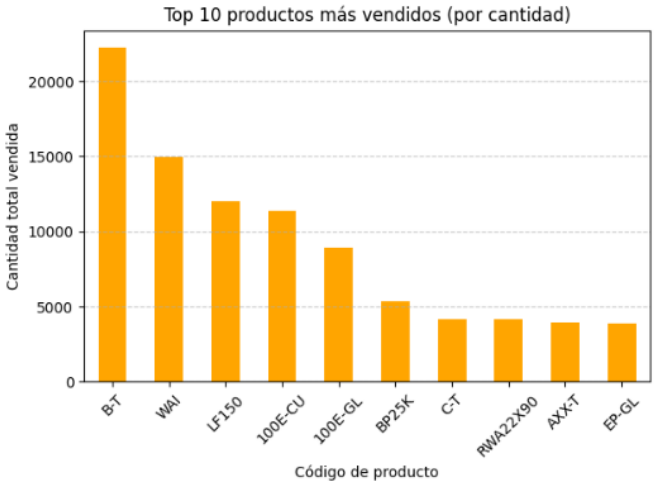


Figura 4 *Distribución de productos más vendidos*

El gráfico evidencia los diez productos más vendidos en términos de cantidad. Destacan referencias como “B-T” y “WAI” con volúmenes significativamente superiores al resto, lo cual puede estar asociado a su rotación comercial o demanda constante.

Esta fase permitió identificar variables clave para el análisis comercial, tales como la frecuencia de compra por cliente, el comportamiento de las ventas por marca y la rentabilidad asociada a cada producto. Estos elementos son fundamentales para orientar futuras decisiones estratégicas en la empresa.

4.1.3 Preparación adecuada de los datos

Una de las fases más críticas en el desarrollo de sistemas basados en inteligencia artificial y visualización es la preparación adecuada de los datos. Esta etapa garantiza la calidad, consistencia y estructura necesarias para que los modelos conversacionales funcionen correctamente y que el dashboard interactivo proporcione información confiable**.**

Selección y limpieza de los datos históricos de ventas

Para la preparación del conjunto de datos, se trabajó con archivos exportados desde el sistema de facturación de la empresa Arte Color. Utilizando el lenguaje de programación Python junto con las bibliotecas Pandas y NumPy, se llevó a cabo un proceso sistemático de limpieza y transformación de datos, con el fin de garantizar su calidad y adecuación para el análisis.

Las principales tareas realizadas durante esta fase fueron:

- Unificación de archivos: Se consolidaron los registros de ambas sucursales en un único DataFrame, facilitando su análisis conjunto.

- Depuración de registros inválidos: Se eliminaron filas duplicadas, valores nulos en campos clave, registros con precios iguales a cero y códigos inconsistentes.

- Estandarización de formatos: Se normalizaron los nombres de columnas, se eliminaron caracteres y espacios innecesarios, y se ajustaron los tipos de datos a formatos adecuados (por ejemplo, fechas como datetime, montos como float, y códigos como string).

- Homogeneización textual: Se aplicó limpieza básica a campos textuales como nombres de productos, marcas y clientes, asegurando coherencia para posteriores análisis y visualizaciones.

Unificación y Consolidación de Archivos CSV

- Se unieron los archivos de ventas de Sucursal Matriz y Sucursal Cerámica en un solo dataset consolidado.
- Se verificó que ambas fuentes tengan las mismas columnas y formato.

```
# archivos CSV
df_matriz = pd.read_csv("DetalleVentas_Producto_Matriz.csv")
df_ceramica = pd.read_csv("DetalleVentas_Producto_Ceramica.csv")

# Unir los archivos en un solo DataFrame
df_ventas_empresa = pd.concat([df_matriz, df_ceramica], ignore_index=True)
```

Resultado: Se obtuvo un único dataset con todas las ventas registradas.

	es_codigo	it_codigo	em_codigo	su_codigo	bo_codigo	dv_cantid	dv_candes	dv_precio	dv_total	it_codigo.1	...	ce_razons
0	2	6357	1	1	1	2.0	2.0	13.30	26.60	6357	...	NaN
1	2	6357	1	1	1	1.0	1.0	13.30	13.30	6357	...	NaN
2	2	6357	1	1	1	1.0	1.0	13.30	12.24	6357	...	CONSUMIDOR FINAL
3	2	6357	1	1	1	1.0	1.0	13.31	13.31	6357	...	CONSUMIDOR FINAL
4	2	6357	1	1	1	2.0	2.0	13.30	24.47	6357	...	NaN

5 rows × 34 columns

Figura 5 *Dataset de ventas inicial unificado*

Selección y Renombrado de Columnas

- Se seleccionaron únicamente las columnas relevantes para el modelo de datos.
- Se renombraron las columnas para asegurar uniformidad con la estructura de la base de datos.

```
[ ] # Renombramos solo las columnas que vamos a usar para la base de datos
    df_ventas_empresa.rename(columns={
        'it_codant': 'producto_codigo',
        'it_nombre': 'producto_nombre',
        'ma_codigo': 'marca_codigo',
        'su_codigo': 'sucursal_codigo',
        'dv_precio': 'producto_precio',
        've_numero': 'venta_numero',
        've_fecha': 'venta_fecha',
        'ce_codigo': 'cliente_codigo',
        've_neto': 'venta_valor_neto',
        've_valpag': 'venta_valor_pagado',
        'dv_cantid': 'detalle_cantidad',
        'dv_total': 'detalle_valor_total',
        'ce_nombre': 'cliente_nombre',
        'ce_rucic': 'cliente_rucic'
    }, inplace=True)

    # Filtra solo las columnas necesarias
    df_ventas_empresa = df_ventas_empresa[[
        'producto_codigo', 'producto_nombre', 'marca_codigo', 'sucursal_codigo',
        'producto_precio', 'venta_numero', 'venta_fecha', 'cliente_codigo',
        'venta_valor_neto', 'venta_valor_pagado', 'detalle_cantidad',
        'detalle_valor_total', 'cliente_nombre', 'cliente_rucic'
    ]]

    # primeras filas del dataset consolidado
    df_ventas_empresa.head()
```

Figura 6 *Fragmento de código en Python para renombrar y seleccionar columnas*

Resultado: Se estandarizaron los nombres de columnas para facilitar la carga en MySQL.

	producto_codigo	producto_nombre	marca_codigo	sucursal_codigo	producto_precio	venta_numero	venta_fecha	cliente_codigo	venta_valor_neto	venta_valor_pagado	detalle_cantidad	detalle_valor_total	cliente_nombre	cliente_rucic
58	117200-CA	INTERVINIL BLANCO PURO CANECA	35	1	83.33	40049505	2023-01-11 09:14:21	24138	182.93	204.88	2.0	146.66	ALEX SCHFFRY	9089795370O1
59	117200-CA	INTERVINIL BLANCO PURO CANECA	35	1	83.33	40050633	2023-02-07 10:39:02	24557	328.88	368.35	1.0	83.33	ROSA CHANGO	1802679207
60	117200-CA	INTERVINIL BLANCO PURO CANECA	35	1	83.33	40050891	2023-02-13 17:59:22	24588	211.22	236.57	1.0	75.00	JUAN CARLOS OCHOA	1713254629
61	117200-CA	INTERVINIL BLANCO PURO CANECA	35	1	83.33	40050933	2023-02-14 17:09:44	15226	75.00	84.00	1.0	75.00	JULIAN FORERO	1759788514001
62	117200-CA	INTERVINIL BLANCO PURO CANECA	35	1	83.33	40050947	2023-02-15 09:58:18	24634	83.33	93.33	1.0	83.33	WANG SHENGGANG	1758632309001

Figura 7 *Dataset de columnas estandarizadas*

Conversión de tipos de datos

Se aseguraron los formatos correctos para cada variable para evitar errores de formato. Las fechas se convirtieron a datetime. Los valores numéricos a float o int. y las variables de Texto a string.

```
# Convierte la columna de fecha al formato datetime
df_ventas_empresa['venta_fecha'] = pd.to_datetime(df_ventas_empresa['venta_fecha'])

# Convierte columnas numéricas
df_ventas_empresa['sucursal_codigo'] = pd.to_numeric(df_ventas_empresa['sucursal_codigo'],
df_ventas_empresa['producto_precio'] = pd.to_numeric(df_ventas_empresa['producto_precio'],
df_ventas_empresa['detalle_cantidad'] = pd.to_numeric(df_ventas_empresa['detalle_cantidad'
df_ventas_empresa['detalle_valor_total'] = pd.to_numeric(df_ventas_empresa['detalle_valor_
df_ventas_empresa['venta_valor_neto'] = pd.to_numeric(df_ventas_empresa['venta_valor_neto'
df_ventas_empresa['venta_valor_pagado'] = pd.to_numeric(df_ventas_empresa['venta_valor_pag.

# Nos aseguramos que las columnas de texto sean tipo string
df_ventas_empresa['producto_codigo'] = df_ventas_empresa['producto_codigo'].astype(str)
df_ventas_empresa['producto_nombre'] = df_ventas_empresa['producto_nombre'].astype(str)
df_ventas_empresa['marca_codigo'] = df_ventas_empresa['marca_codigo'].astype(str)
df_ventas_empresa['venta_numero'] = df_ventas_empresa['venta_numero'].astype(str)
df_ventas_empresa['cliente_codigo'] = df_ventas_empresa['cliente_codigo'].astype(str)
df_ventas_empresa['cliente_nombre'] = df_ventas_empresa['cliente_nombre'].astype(str)
df_ventas_empresa['cliente_rucic'] = df_ventas_empresa['cliente_rucic'].astype(str)

# Verifica los tipos de datos después de la conversión
print(df_ventas_empresa.dtypes)
```

Figura 8 *Fragmento de código en Python para conversión*

Resultado: Se garantizaron los formatos correctos para cada columna.

```
producto_codigo              object
producto_nombre              object
marca_codigo                 object
sucursal_codigo               int64
producto_precio             float64
venta_numero                 object
venta_fecha          datetime64[ns]
cliente_codigo               object
venta_valor_neto            float64
venta_valor_pagado          float64
detalle_cantidad            float64
detalle_valor_total         float64
cliente_nombre               object
cliente_rucic                object
dtype: object
```

Verificación de valores duplicados

Se identificaron registros duplicados en los datos de productos y marcas.

```
producto_nombre
.                                                16
ANTICORROSIVO NEGRO 1LT UNIDAS                    2
ANTICORROSIVO NEGRO 4LT UNIDAS                    2
CONDORTHANE  WASHPRIMER 1/8LT CONDOR              2
EMPORE BLANCO SIN ARENA 2KG GRAIMAN               2
ESMALTE AZUL ELECTRICO UNIDAS 4LT                 2
ESMALTE EFECTO CROMADO                            2
ESMALTE ICO VERDE T2 DURACION Y BRILLO 1LT        2
ESMALTE PINTULUX MP ROJO FIESTA 112244            2
FREGADERO TRAMONTINA INOX 120X50 P/FUNDIR 2 POZOS 2
LACA CATALIZADA BRILLANTE                         3
LACA CATALIZADA MATE ALTOS SOLIDOS                3
LACA SELLADOR CATALIZADO UNIDAS                   3
MAZDA 46V BASECOAT + TOPCOAT+ ENDU                3
OPTIMA SATINADO BLANCO 4LT                        2
PASTA MATEANTE UNITHANE                           2
PISO FLOTANTE CASTER LAMINATE FLOORING AC3 8MM+ IN 3
PRIMER ACRILICO CINZA DIRECTO                     4
SATINADO ROJO PRODUTEK SATINGLOS                  2
THINNER POLIESTER/ POLIURETANO SHERWIN WILIAM     2
TINTE AL AGUA WENGUE 1LT VHP                      2
Name: producto_codigo, dtype: int64
```

Productos: Al analizar los datos manualmente, se pudo evidenciar que La repetición de códigos se debe a que un mismo producto tiene distintas presentaciones, como galón (GL), litro (CU) o caneca (CA).

Además, los códigos varían según el color de las pinturas y otras características específicas del producto y también por la sucursal a la que pertenece el producto. Por lo que, se decidió mantener los códigos tal como están, ya que reflejan diferencias reales en la presentación y características de los productos.

Marcas: Se pudo identificar que el código de la marca (Pintuco) se repite 2 veces, por lo que se decidió eliminar la fila de la marca de Pintuco con código 2.

```
marca_nombre
PINTUCO    2
Name: marca_codigo, dtype: int64
```

	marca_codigo	marca_nombre
3295	2	PINTUCO
3505	35	PINTUCO

Debido a que se seleccionó un solo código para la marca Pintuco, ajustamos también esto en la variable marca_codigo del dataset final anterior. Asignamos el código de marca más frecuente para Pintuco que es el código 35, en todo el dataset principal de ventas.

Eliminación de Registros con Datos Inválidos

Se identificaron y eliminaron registros con códigos de productos no válidos ("0", "00", "."), por lo que Se eliminaron 288 filas con valores inválidos.

```python
# Filtramos las filas donde 'producto_codigo' o 'producto_nombre'
filas_invalidas = df_ventas_empresa[
    df_ventas_empresa['producto_codigo'].isin(['.', '0', '00']) |
    df_ventas_empresa['producto_nombre'].isin(['.', '0', '00'])
]

print(filas_invalidas)
```

```python
# Mostrar cuántas filas se eliminarán
print(f"Se eliminaron {len(filas_invalidas)} filas.")

# Elimina las filas del dataframe
df_ventas_empresa = df_ventas_empresa.drop(filas_invalidas.index)

# Verifica que las filas fueron eliminadas
print("número de filas del dataset:")
print(df_ventas_empresa.shape)
```

```
Se eliminaron 288 filas.
número de filas del dataset:
(122058, 14)
```

Figura 9 *Fragmento de código en Python para eliminación datos inválidos*

Resultado: Se eliminaron 288 registros inválidos

Eliminación de Espacios en Blanco en Columnas de Texto

Para asegurar la limpieza de los datos, se eliminaron los espacios en blanco al inicio y al final de todas las variables de tipo texto en el dataset.

```
# Mostrar registros donde se detectan espacios al inicio o al final
df_ventas_empresa[espacios_detectados.any(axis=1)]
```

	producto_codigo	producto_nombre	marca_codigo	sucursal_codigo	producto_precio	venta_numero	venta_fecha	cliente_codigo
58	117200-CA	INTERVINIL BLANCO PURO CANECA	35	1	83.33	40049505	2023-01-11 09:14:21	24138
59	117200-CA	INTERVINIL BLANCO PURO CANECA	35	1	83.33	40050633	2023-02-07 10:39:02	24557
60	117200-CA	INTERVINIL BLANCO PURO CANECA	35	1	83.33	40050891	2023-02-13 17:59:22	24588
61	117200-CA	INTERVINIL BLANCO PURO CANECA	35	1	83.33	40050933	2023-02-14 17:09:44	15226
62	117200-CA	INTERVINIL BLANCO PURO CANECA	35	1	83.33	40050947	2023-02-15 09:58:18	24634
...	...	...	...	...	...	...	...	...
234059	YESPE-LB	YESO SISEKAROF ENV. 1LB	0	2	0.45	30047615	2024-02-01 09:50:44	27748
234062	YESPE-LB	YESO SISEKAROF ENV. 1LB	0	2	0.45	30047877	2024-02-09 10:13:08	20874
234078	YESPE-LB	YESO SISEKAROF ENV. 1LB	0	2	0.30	30052056	2024-07-05 10:27:16	4760
234106	125.580	YETKA REJILLA CROMADA CUADRADA ANTIOLOR	35	2	8.69	30047489	2024-01-27 12:20:19	10936
234107	125.580	YETKA REJILLA CROMADA CUADRADA ANTIOLOR	35	2	8.69	30047830	2024-02-07 17:10:02	27800

Figura 10 *Eliminación de espacios en blanco*

Resultado: Se evitó que registros con el mismo nombre sean tratados como diferentes por espacios adicionales.

Validación de Rangos y Eliminación de Valores atípicos

- Se realizó una validación de los valores numéricos en variables clave como precios y cantidades para detectar valores atípicos que pudieran afectar el análisis.

- Se verificó que las cantidades y valores de venta sean mayores a 0.
- Se identificaron productos con precios en 0.00, los cuales fueron eliminados.

```python
# Filtrar valores fuera de los rangos esperados
valores_invalidos = df_ventas_empresa[
    (df_ventas_empresa['producto_precio'] <= 0) |
    (df_ventas_empresa['detalle_cantidad'] < 0) |
    (df_ventas_empresa['venta_valor_neto'] < 0) |
    (df_ventas_empresa['venta_valor_pagado'] < 0) |
    (df_ventas_empresa['detalle_valor_total'] < 0)
]

# Mostrar registros con valores inválidos
print("Filas con valores fuera de rango:")
print(valores_invalidos)
```

```python
# muestra precios de productos con valor 0.00
df_ventas_empresa[df_ventas_empresa['producto_precio'] == 0]
```

	producto_codigo	producto_nombre	marca_codigo	sucursal_codigo	producto_precio	venta_numero	venta_fecha	clien
208947	PO274	PORCELANA GROUTEX ALMENDRA 4 KILOS INTACO	16	2	0.0	30054953	2024-10-07 10:32:53	

Figura 11 *Fragmento de código en Python para validación de rangos y eliminación de valores atípicos*

Resultado: Se eliminó 1 registro con precio de producto en 0.00.

Este preprocesamiento permitió obtener una versión confiable y estructurada de los datos, lista para ser utilizada directamente por los modelos de lenguaje en el entrenamiento del asistente.

Creación de Columnas

Marcas: Se creó una columna nueva para el nombre de marca en el dataset df_ventas_final en base al dataset de marcas para tener un solo dataset completo.

```python
# Unir ambos datasets por la columna 'marca_codigo'
df_ventas_con_marcas = pd.merge(df_ventas_final, df_marcas_final, on="marca_codigo", how="left")
# Guardar el nuevo archivo con marca_nombre incluida
df_ventas_con_marcas.to_csv("DetalleVentas_ArteColor.csv", index=False, encoding="utf-8")
```

Figura 12 *Fragmento de código en Python para la asignación de las marcas en el dataset*

Sucursales: Se creó una columna nueva para el nombre de la sucursal en base a la variable sucursal_codigo

```python
# Diccionario para mapear códigos de sucursal a nombres
mapa_sucursales = {
    "1": "Matriz - Pinturas",
    "2": "Sucursal - Cerámica"
}

# Agregar columna 'sucursal_nombre' usando el mapeo
df_ventas_con_marcas["sucursal_nombre"] = df_ventas_con_marcas["sucursal_codigo"].map(mapa_sucursales)

# Verificar si hubo códigos no reconocidos (NaN en sucursal_nombre)
nulos = df_ventas_con_marcas["sucursal_nombre"].isnull().sum()
print(f" ! Valores nulos en sucursal_nombre: {nulos}")

# Renombrar el dataframe final
df_final_ventas = df_ventas_con_marcas
```

Figura 13 *Fragmento de código en Python para creación de columnas: marcas, sucursales*

Creación de Identificadores Únicos

Se generaron un ID único por cada ítem de las ventas

```python
# Crear un identificador único por fila
df_final_ventas.reset_index(drop=True, inplace=True)
df_final_ventas["detalle_id"] = df_final_ventas.index + 1  # Empieza desde 1
df_final_ventas.head()
```

	detalle_id	venta_numero	venta_fecha	cliente_codigo	cliente_nombre	cliente_rucic	sucursal_codigo	sucursal_nombre	producto_co
0	1	40049505	2023-01-11 09:14:21	24138	ALEX SCHFFRY	908979537001	1	Matriz - Pinturas	11720
1	2	40050633	2023-02-07 10:39:02	24557	ROSA CHANGO	1802679207	1	Matriz - Pinturas	11720
2	3	40050891	2023-02-13 17:59:22	24588	JUAN CARLOS OCHOA	1713254629	1	Matriz - Pinturas	11720
3	4	40050933	2023-02-14 17:09:44	15226	JULIAN FORERO	1759788514001	1	Matriz - Pinturas	11720
4	5	40050947	2023-02-15 09:58:18	24634	WANG SHENGGANG	1758632309001	1	Matriz - Pinturas	11720

Figura 14 *Creación de ID Únicos*

Resultado: Se evitó problemas de duplicación en las combinaciones de cada transacción.

Agregación temporal

Se creo columnas temporales para agrupar datos por semana, mes y trimestre, lo cual facilitó el análisis de tendencias comerciales.

```
# 2. Convertir la columna de fecha
df_final_ventas["venta_fecha"] = pd.to_datetime(df_final_ventas["venta_fecha"], errors='coerce')

# 3. Crear columnas temporales
df_final_ventas["año"] = df_final_ventas["venta_fecha"].dt.year
df_final_ventas["mes"] = df_final_ventas["venta_fecha"].dt.month
df_final_ventas["trimestre"] = df_final_ventas["venta_fecha"].dt.quarter
df_final_ventas["semestre"] = df_final_ventas["venta_fecha"].dt.month.apply(lambda x: 1 if x <= 6 else 2)

# 4. Diccionarios para traducir al español
meses_es = {
    "January": "Enero", "February": "Febrero", "March": "Marzo", "April": "Abril",
    "May": "Mayo", "June": "Junio", "July": "Julio", "August": "Agosto",
    "September": "Septiembre", "October": "Octubre", "November": "Noviembre", "December": "Diciembre"
}

dias_es = {
    "Monday": "Lunes", "Tuesday": "Martes", "Wednesday": "Miércoles", "Thursday": "Jueves",
    "Friday": "Viernes", "Saturday": "Sábado", "Sunday": "Domingo"
}

# 5. Asignar nombres traducidos
df_final_ventas["mes_nombre"] = df_final_ventas["venta_fecha"].dt.strftime("%B").map(meses_es)
df_final_ventas["dia_semana"] = df_final_ventas["venta_fecha"].dt.day_name().map(dias_es)
```

:a_valor_pagado	año	mes	trimestre	semestre	mes_nombre	dia_semana
204.88	2023	1	1	1	Enero	Miércoles
368.35	2023	2	1	1	Febrero	Martes
236.57	2023	2	1	1	Febrero	Lunes
84.00	2023	2	1	1	Febrero	Martes

Figura 15 *Creación de columnas temporales para el análisis de tendencias comerciales*

Finalmente mostramos los primeros registros del dataset limpio y lo guardamos como BDD_ArteColor_Ventas.csv, para usarlo en las siguientes etapas del prototipo.

talle_id	venta_numero	venta_fecha	cliente_codigo	cliente_nombre	cliente_rucic	sucursal_codigo	sucursal_nombre	producto_codigo	producto_nombre	...	producto_precio	detalle_valor_total	venta_valor_neto	venta_valor_pagado	año	mes	trimestre	semestre	mes_nombre	dia_semana
1	40049505	2023-01-11 09:14:21	24138	ALEX SCHFFRY	908979537001	1	Matriz - Pinturas	117200-CA	INTERVINIL BLANCO PURO CANECA	...	83.33	146.66	182.93	204.88	2023	1	1	1	Enero	Miércoles
2	40050633	2023-02-07 10:39:02	24557	ROSA CHANGO	1802679207	1	Matriz - Pinturas	117200-CA	INTERVINIL BLANCO PURO CANECA	...	83.33	83.33	328.88	368.35	2023	2	1	1	Febrero	Martes
3	40050891	2023-02-13 17:59:22	24588	JUAN CARLOS OCHOA	1713254629	1	Matriz - Pinturas	117200-CA	INTERVINIL BLANCO PURO CANECA	...	83.33	75.00	211.22	236.57	2023	2	1	1	Febrero	Lunes
4	40050933	2023-02-14 17:09:44	15226	JULIAN FORERO	1759788514001	1	Matriz - Pinturas	117200-CA	INTERVINIL BLANCO PURO CANECA	...	83.33	75.00	75.00	84.00	2023	2	1	1	Febrero	Martes
5	40050947	2023-02-15 09:58:18	24634	WANG SHENGGANG	1758632309001	1	Matriz - Pinturas	117200-CA	INTERVINIL BLANCO PURO CANECA	...	83.33	83.33	83.33	93.33	2023	2	1	1	Febrero	Miércoles

× 23 columns

Figura 16 *Vista previa del archivo final de ventas tras la preparación del dataset*

Todo este proceso aseguró que los datos estuvieran listos para ser utilizados tanto en los modelos de lenguaje como en la visualización interactiva, maximizando su valor informativo.

4.2 MODELADO DEL ASISTENTE CONVERSACIONAL

Esta sección presenta el proceso de configuración y entrenamiento del asistente conversacional inteligente, centrado en responder consultas comerciales sobre Arte Color mediante lenguaje natural. Para ello, se utilizaron tres modelos de lenguaje preentrenados integrados en la plataforma Stack AI, que permitió conectarlos a un archivo CSV estructurado con datos históricos de ventas.

4.2.1 Selección de los Modelos de lenguaje (LLM)

Se seleccionaron tres modelos de lenguaje grande (LLM) con el fin de comparar su precisión y capacidad de interpretación frente a preguntas basadas en datos reales de Arte Color. La Tabla 1 resume las características clave de cada modelo utilizado:

Tabla 1 *Comparativa entre GPT-4o, Claude 3.5 Sonnet y Gemini 1.5 Pro en criterios clave*

Criterio	GPT-4o (OpenAI)	Claude 3.5 Soneto (Antrópico)	Géminis 1.5 Pro (Google)
Tamaño del modelo	1,8 billones de parámetros	175 mil millones de parámetros	Más de 200 mil millones de parámetros
Ventana de contexto	128.000 tokens	200.000 tokens	Hasta 1.000.000 de tokens
Capacidades multimodales	Sí: texto, imagen, audio, vídeo	Parcial: texto e imagen (sin audio/vídeo)	Sí: texto, imagen, audio, vídeo
Velocidad y eficiencia	Alta velocidad, buena eficiencia computacional.	Muy eficiente, respuestas rápidas y concisas.	Alta capacidad, pero mayor demanda de cómputo
Precisión en razonamiento	Alta: destaca en tareas de lógica, análisis contextual y programación.	Muy alta: respuestas estructuradas, seguras y claras	Alta: eficaz para tareas complejas y recuperación de información
Capacidad para análisis de datos	Excelente interpretación de datos tabulares y visualizaciones.	Capaz de explicar datos estructurados, con enfoque claro y seguro.	Útil con grandes volúmenes y en integración con herramientas Google
Desventajas	Ligeramente menos detallados en matices técnicos que Claude	No soporta audio ni vídeo, menor capacidad multimodal.	Alto requerimiento computacional, aún en evolución constante

Nota. Elaboración propia con base en documentación técnica de OpenAI (2024), Anthropic (2024) y Google (2024).

4.2.2 Configuración de Stack AI

Stack AI fue la plataforma utilizada para conectar los LLM con los datos estructurados. Se creó un proyecto nuevo y se cargó el archivo CSV preprocesado que contenía información consolidada de Arte Color. Para la configuración del asistente conversacional, se llevaron a cabo los siguientes pasos:

- Creación de un nuevo proyecto dentro de la plataforma Stack AI.
- Carga del archivo CSV preprocesado que contenía información consolidada sobre ventas, productos, clientes, marcas, sucursales y variables temporales.
- Definición de los campos clave (variables) disponibles para el análisis y generación de respuestas.
- Configuración del contexto general del negocio y de las instrucciones de comportamiento del asistente.

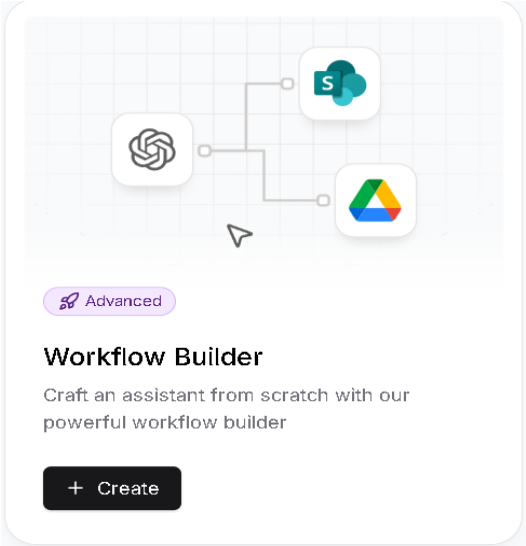


Figura 17 *Creación del chat en Stack AI*

Nota. Se muestra la opción de crear flujos conversacionales personalizados utilizando el módulo Workflow Builder de Stack AI. Esta función permite conectar modelos de lenguaje con otras herramientas, como hojas de cálculo, bases de datos o almacenamiento en la nube.

Esta configuración permitió restringir el dominio conversacional exclusivamente a los datos internos de Arte Color, delimitando el dominio conversacional al contexto de ventas, lo que redujo el riesgo de que el asistente generara respuestas erróneas, ambiguas o fuera de alcance.

A continuación, se detallan los prompts que permitieron interpretar las consultas requeridas sobre los datos ya estructurados.

4.2.3 Definición del Prompt e instrucciones de respuesta

Se diseñó un rápido personalizado que sirvió como guía para el comportamiento del asistente. El objetivo fue asegurar respuestas basadas exclusivamente en el contenido del CSV, expresadas en un lenguaje claro, conciso y útil para la toma de decisiones comerciales. A continuación, se muestra un extracto de las instrucciones utilizadas:

Instrucciones de respuesta

"Eres una IA especializada en responder preguntas sobre la empresa Arte Color, una distribuidora de productos de acabados de construcción fundada por la Familia Campaña hace 30 años. Tu función principal es proporcionar información clave para la toma de decisiones gerenciales basada en datos de ventas, clientes, productos, sucursales y marcas. Debes responder de forma clara, educada y conversacional.

Tus respuestas deben ser breves, precisas y basadas en datos. Si no tienes suficiente información, responde de manera educada que no puedes proporcionar una respuesta confiable. No generes respuestas ficticias ni inventes datos. Finaliza cada respuesta con dos recomendaciones estratégicas."

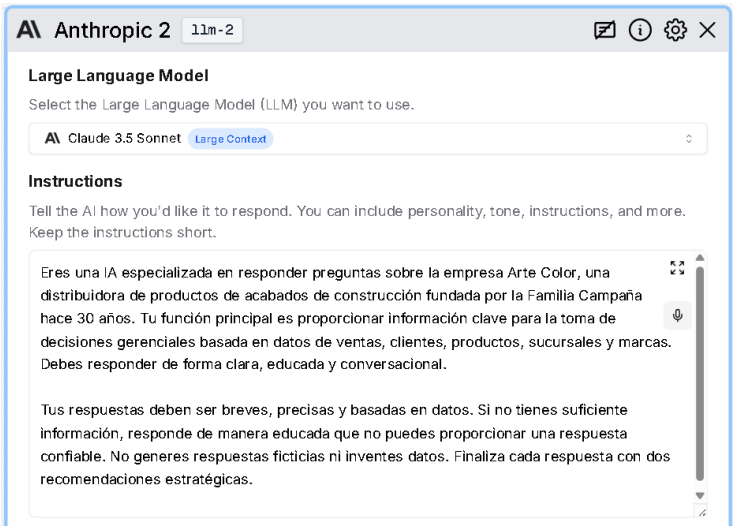


Figura 18 *Instrucciones de respuesta para el LLM*

Esta indicación incluye indicaciones para no generar respuestas ficticias, mantenga un tono conversacional y agregue recomendaciones estratégicas al final de cada respuesta.

Definición de Variables Clave:

Para optimizar la generación de respuestas, se seleccionaron las siguientes variables clave, alineadas con la estructura del Dataset para el entrenamiento de la IA:

- **Clientes**: Código del cliente, nombre, identificación tributaria (RUC/CI).
- **Productos**: Código del producto, nombre, marca.
- **Marcas**: Código y nombre de la marca.
- **Sucursales**: Código y nombre de la sucursal.
- **Ventas**: Número de factura, fecha, valores netos y pagados.
- **Detalle de Ventas**: Cantidad vendida, precio unitario, total de cada venta.
- **Dimensión Tiempo**: Fecha de la venta, año, mes, día, trimestre y semana.

Consultas de Referencia

Durante la configuración inicial del asistente en la plataforma Stack AI, se incorporaron ejemplos de consultas representativas con el objetivo de orientar al modelo en la interpretación del contexto comercial de Arte Color. Estas preguntas sirvieron como guía conversacional para delimitar el dominio temático y estructurar las respuestas en lenguaje natural, además sirvieron como referencia para la evaluación de los modelos en la siguiente fase.

A continuación, se presentan ejemplos utilizados en la configuración:

- **Consultas por tiempo:** ¿Cuánto vendimos en enero?, ¿Qué día de la semana se registraron más ventas?
- **Consultas por producto:** ¿Cuál fue el producto más vendido en 2023?, ¿Qué productos han disminuido sus ventas en el último trimestre?
- **Consultas por cliente:** ¿Qué cliente compró más en la sucursal Villaflora? ¿Qué clientes dejaron de comprar este año?
- **Consultas por sucursal:** ¿Cuál sucursal generó mayores ingresos en marzo?, ¿Dónde se vendieron más cerámicas en febrero?
- **Consultas por marca:** ¿Qué marca tuvo más ventas en el segundo semestre?, ¿Las ventas de Pintuco aumentaron o disminuyeron en 2024?

Estas preguntas permitieron configurar un entorno conversacional realista y alineado con las necesidades analíticas de Arte Color, contribuyendo a que los modelos interpretaran correctamente las variables clave del archivo estructurado.

Validaciones y límites del asistente

Para evitar respuestas incorrectas o fuera del alcance del archivo, se implementaron validaciones específicas. Por ejemplo, si el usuario realiza preguntas sobre competencia, precios futuros o tendencias de mercado, el asistente debe responder indicando que su conocimiento se limita a los datos internos de Arte Color.

"Los datos utilizados provienen de un archivo CSV preprocesado, consolidando información de ventas, clientes, productos, sucursales y marcas. Cada registro representa un detalle de venta e incluye información clave sobre los movimientos comerciales de Arte Color."

Reglas para consultas Especificas:

Para mejorar la precisión de la IA, se establecieron reglas específicas según el tipo de consulta:

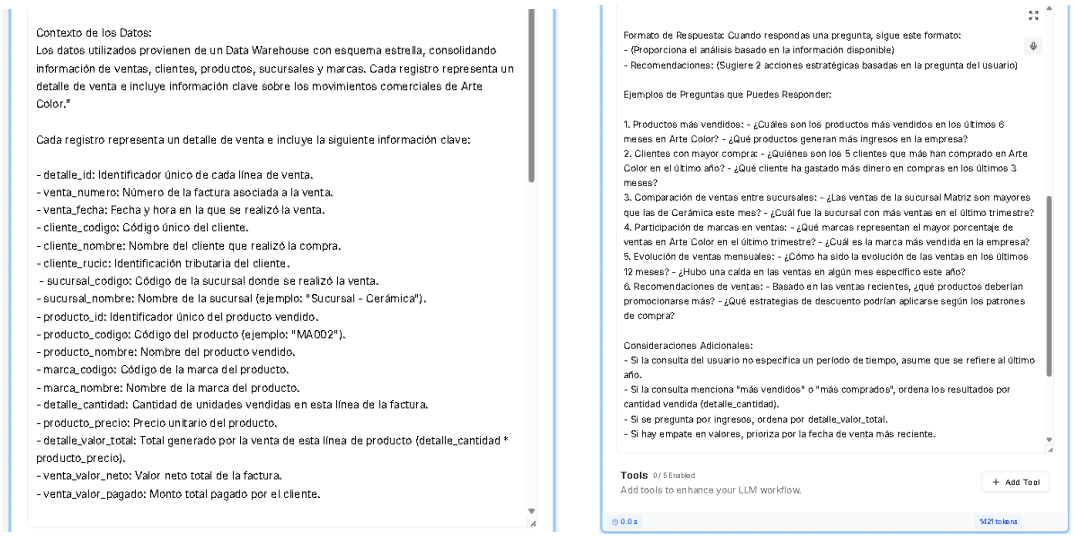


Figura 19 *Configuración de instrucciones conversacionales en Stack AI para el asistente de Arte Color*

4.2.4 Interacción del Usuario con el asistente

Los usuarios pueden interactuar con el asistente a través de una interfaz tipo chat de Stack AI. Las consultas se realizan en lenguaje natural, y el modelo responde con recomendaciones precisas y estratégicas.

Acceso: Los usuarios interactúan con la IA mediante un chat.

Consulta: Ejemplos de preguntas:

- ¿Cuáles son los productos más vendidos?
- ¿Cuál es la sucursal con más ventas?
- ¿Qué clientes compran con mayor frecuencia?

Análisis de Datos: El LLM interpreta la consulta, accede al archivo CSV y genera una respuesta basada en datos reales.

Recomendaciones Estratégicas: Cada respuesta incluye dos sugerencias para mejorar la gestión comercial.

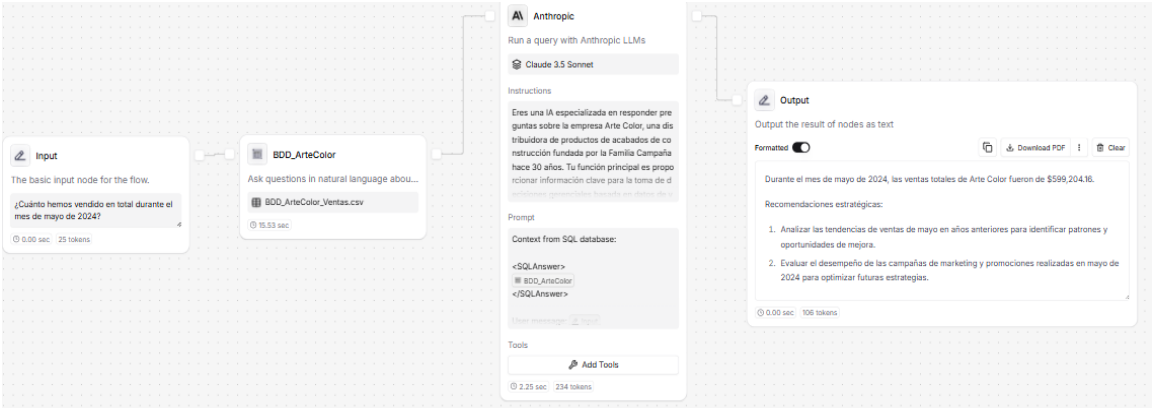


Figura 20 *Flujo de consulta en Stack AI*

A continuación, se presentan ejemplos de preguntas formuladas a cada modelo y las respuestas generadas, permitiendo observar diferencias en la forma y claridad entre los 3 LLM.

Claude 3.5 Sonnet (Anthropic):

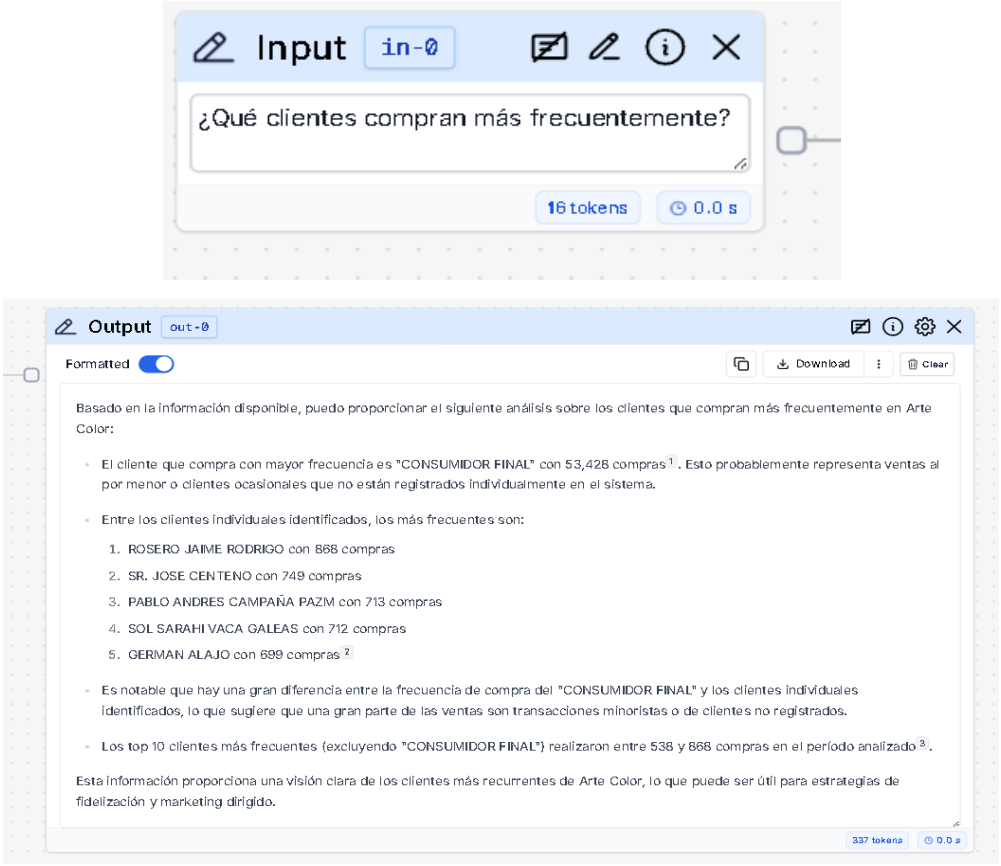
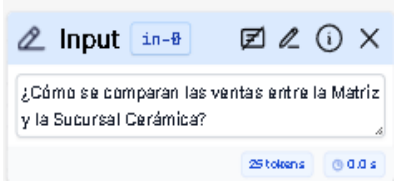


Figura 21 *Ejemplo de consulta y respuesta generada por Claude 3.5 Sonnet*

GPT-4o (OpenAI):



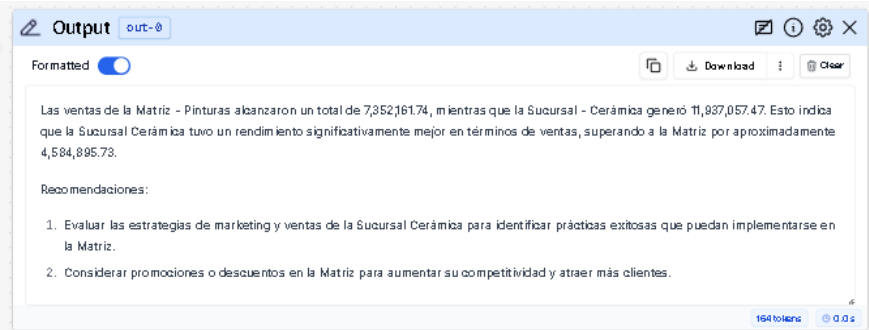


Figura 22 *Ejemplo de consulta y respuesta generada por GPT-4o*

Gemini 1.5 Pro (Google)

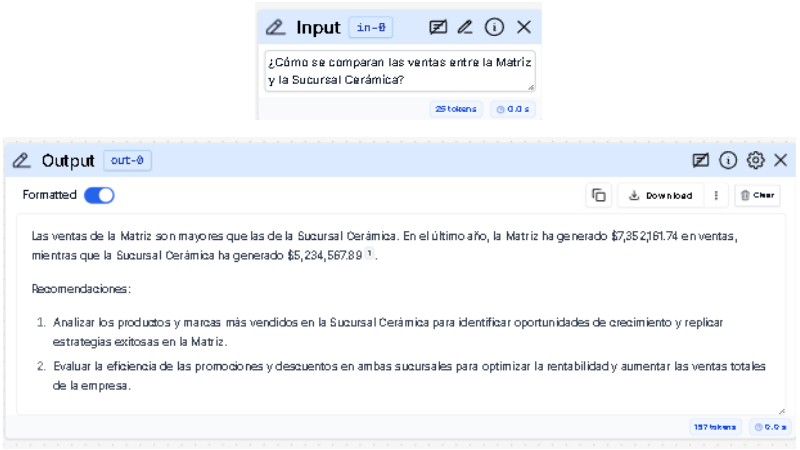


Figura 23 *Ejemplo de consulta y respuesta generada por Gemini 1.5 Pro*

Asimismo, se implementaron validaciones de respuesta para asegurar que el asistente no genere respuestas fuera del alcance del archivo cargado. Por ejemplo, si se pregunta por “competencia” o “tendencias de mercado”, el modelo responde indicando que su conocimiento está limitado a los datos internos de Arte Color.

4.3 EVALUACIÓN DE LOS MODELOS APLICADOS AL ASISTENTE CONVERSACIONAL

El presente apartado desarrolla el proceso de evaluación comparativa entre tres modelos de lenguaje natural (Large Language Models, LLMs) integrados al prototipo de asistente conversacional propuesto en esta investigación. Esta evaluación responde directamente al Objetivo Específico 3: "Evaluar la precisión de los modelos de entrenamiento del asistente conversacional aplicados a los datos de Arte Color", con el propósito de seleccionar el modelo más adecuado para apoyar la toma de decisiones comerciales en Arte Color.

4.3.1 Modelos de Lenguaje Evaluados

En esta evaluación se consideraron tres modelos de última generación, desarrollados por las principales empresas tecnológicas del mundo:

- GPT-4o (OpenAI): Modelo multimodal altamente optimizado para razonamiento, precisión factual y velocidad de respuesta. Combina rendimiento técnico con una sólida comprensión contextual.

- Claude 3.5 Sonnet (Anthropic): Especializado en razonamiento complejo y generación precisa de lenguaje.

- Gemini 1.5 Pro (Google): Modelo orientado a la integración de datos estructurados, con capacidad para comprender secuencias complejas y entregar respuestas detalladas en lenguaje natural.

Estos modelos fueron accedidos y ejecutados a través de la plataforma Stack AI, lo que permitió estandarizar el entorno de inferencia.

4.3.2 Configuración de hiperparámetros

Para asegurar condiciones comparables en la evaluación de los modelos, se utilizaron los siguientes hiperparámetros en Stack AI:

Temperatura: Se realizaron dos configuraciones para cada modelo:

- **0.2:** favorece respuestas más precisas, determinísticas y alineadas con los datos.
- **0.7:** permite evaluar el comportamiento del modelo en condiciones menos restrictivas, detectando posibles errores por interpretaciones libres.

Max Output Length: Se estableció en 1000 tokens como límite máximo de longitud de respuesta, garantizando cobertura total en consultas complejas sin truncamiento.

El objetivo fue observar el impacto de la temperatura en la precisión del modelo, sin alterar la longitud o formato de salida.

4.3.3 Preguntas Base para la Evaluación

La base del dataset de evaluación se construyó a partir de 10 preguntas clave de carácter gerencial, cada una enfocada en una categoría estratégica de análisis comercial. Estas preguntas permitieron generar, mediante variaciones automáticas, un total de 1000 preguntas distintas. Las preguntas fueron formuladas considerando criterios como año, mes, trimestre, semestre y sucursal, abarcando diferentes niveles de agregación temporal y espacial de los datos de ventas.

Tabla 2 *Selección de preguntas base para la evaluación*

N°	Pregunta	Categoría
1	¿Cuánto hemos vendido en total durante el mes de enero del año 2024?	Rendimiento general de ventas
2	¿Cuánto hemos vendido en la Matriz - Pintura durante el mes de enero de 2024?	Rendimiento general de ventas
3	¿Qué productos suelen comprarse juntos en una misma factura durante el primer trimestre de 2024?	Patrones de Compra
4	¿En qué meses del año 2024 se han registrado más ventas totales?	Rendimiento general de ventas
5	¿Qué clientes han generado más ingresos por compras durante el mes de enero de 2024?	Análisis de Clientes
6	¿Qué clientes han comprado con mayor frecuencia en enero de 2024?	Análisis de Clientes
7	¿Cuáles son los productos que han generado más ingresos en el mes de enero de 2024?	Análisis de Productos
8	¿Qué productos se han vendido con mayor frecuencia en enero de 2024?	Análisis de Productos
9	¿Qué productos generaron bajos ingresos por ventas durante el primer trimestre de 2024?	Análisis de Productos
10	¿Qué marcas han generado más ingresos por ventas en el primer semestre de 2024?	Análisis de Marcas

Nota. Estas preguntas se consolidaron en el archivo CSV dataset_preguntas_variadas.csv, y utilizadas como insumo para los modelos.

4.3.4 Generación de respuestas esperadas (Ground Truth)

Para contar con un punto de referencia objetivo, se implementó un script en Python que generó las respuestas esperadas a las 1000 preguntas, directamente a partir del dataset estructurado de ventas de Arte Color.

El script analizó el contenido de cada pregunta, identificó los filtros por fecha y sucursal, aplicó operaciones como sumatoria, conteo, agrupamiento o promedio, y exportó el resultado en un archivo CSV: respuestas_correctas_arte_color.csv.

Este dataset fue utilizado como ground truth para comparar los resultados generados por cada modelo.

```python
import pandas as pd
from collections import Counter
import unicodedata

# CARGA DE DATOS
ventas_df = pd.read_csv("BDD_ArteColor_Ventas.csv", parse_dates=['venta_fecha'])
preguntas_df = pd.read_csv("dataset_preguntas_variadas.csv", sep=';', names=["Pregunta Base", "Pregunta"], header=0)

# Asegurar trimestre y semestre
ventas_df['trimestre'] = ventas_df['venta_fecha'].dt.quarter
ventas_df['semestre'] = ventas_df['mes'].apply(lambda x: 1 if x <= 6 else 2)

# Diccionario meses
meses = {
    "enero": 1, "febrero": 2, "marzo": 3, "abril": 4, "mayo": 5, "junio": 6,
    "julio": 7, "agosto": 8, "septiembre": 9, "octubre": 10, "noviembre": 11, "diciembre": 12
}

# Función para normalizar texto
def normalizar_texto(texto):
    if pd.isna(texto):
        return ""
    return ''.join(
        c for c in unicodedata.normalize('NFD', texto.lower())
        if unicodedata.category(c) != 'Mn'
    )

#  FUNCIONES POR PREGUNTA BASE
def responder_pregunta_1(p):
    for mes_str, mes_num in meses.items():
        if mes_str in normalizar_texto(p):
            año = 2024 if "2024" in p else 2023
            f = ventas_df[(ventas_df['año'] == año) & (ventas_df['mes'] == mes_num)]
            return round(f['detalle_valor_total'].sum(), 2) if not f.empty else None

def responder_pregunta_2(p):
    pregunta_norm = normalizar_texto(p)
    for mes_str, mes_num in meses.items():
        if mes_str in pregunta_norm:
            año = 2024 if "2024" in pregunta_norm else 2023
            if "matriz" in pregunta_norm:
                sucursal = "Matriz - Pinturas"
            elif "ceramica" in pregunta_norm:
                sucursal = "Sucursal - Cerámica"
            else:
                return None
            f = ventas_df[(ventas_df['año'] == año) & (ventas_df['mes'] == mes_num) & (ventas_df['sucursal_nombre'] == sucursal)]
            return round(f['detalle_valor_total'].sum(), 2) if not f.empty else None
```

```python
def responder_pregunta_3(p):
    if "2024" in p:
        f = ventas_df[(ventas_df['año'] == 2024) & (ventas_df['trimestre'] == 1)]
        facturas = f.groupby('venta_numero')['producto_nombre'].apply(list)
        combinaciones = []
        for lista in facturas:
            if len(lista) > 1:
                combinaciones.extend([(a, b) for i, a in enumerate(lista) for b in lista[i + 1:]])
        pares = Counter(combinaciones).most_common(2)
        return [f"{a} & {b}" for (a, b), _ in pares] if pares else None

def responder_pregunta_4(p):
    if "2024" in p:
        f = ventas_df[ventas_df['año'] == 2024]
        totales = f.groupby('mes')['detalle_valor_total'].sum().nlargest(2)
        return totales.index.tolist() if not totales.empty else None

def responder_pregunta_5(p):
    for mes_str, mes_num in meses.items():
        if mes_str in normalizar_texto(p):
            año = 2024 if "2024" in p else 2023
            f = ventas_df[(ventas_df['año'] == año) & (ventas_df['mes'] == mes_num)]
            top = f.groupby('cliente_nombre')['detalle_valor_total'].sum().nlargest(2)
            return top.index.tolist() if not top.empty else None

def responder_pregunta_6(p):
    for mes_str, mes_num in meses.items():
        if mes_str in normalizar_texto(p):
            año = 2024 if "2024" in p else 2023
            f = ventas_df[(ventas_df['año'] == año) & (ventas_df['mes'] == mes_num)]
            frec = f.groupby('cliente_nombre')['venta_numero'].nunique().nlargest(2)
            return frec.index.tolist() if not frec.empty else None

def responder_pregunta_7(p):
    for mes_str, mes_num in meses.items():
        if mes_str in normalizar_texto(p):
            año = 2024 if "2024" in p else 2023
            f = ventas_df[(ventas_df['año'] == año) & (ventas_df['mes'] == mes_num)]
            top = f.groupby('producto_nombre')['detalle_valor_total'].sum().nlargest(2)
            return top.index.tolist() if not top.empty else None

def responder_pregunta_8(p):
    for mes_str, mes_num in meses.items():
        if mes_str in normalizar_texto(p):
            año = 2024 if "2024" in p else 2023
            f = ventas_df[(ventas_df['año'] == año) & (ventas_df['mes'] == mes_num)]
            top = f.groupby('producto_nombre')['detalle_cantidad'].sum().nlargest(2)
            return top.index.tolist() if not top.empty else None
```

```python
def responder_pregunta_9(p):
    if "2024" in p:
        f = ventas_df[(ventas_df['año'] == 2024) & (ventas_df['trimestre'] == 1)]
        bajos = f.groupby('producto_nombre')['detalle_valor_total'].sum().nsmallest(2)
        return bajos.index.tolist() if not bajos.empty else None

def responder_pregunta_10(p):
    if "2024" in p:
        f = ventas_df[(ventas_df['año'] == 2024) & (ventas_df['semestre'] == 1)]
        top = f.groupby('marca_nombre')['detalle_valor_total'].sum().nlargest(2)
        return top.index.tolist() if not top.empty else None

# --- MAPEO Y GENERACIÓN DE RESPUESTAS ---
funciones = {
    "Pregunta 1": responder_pregunta_1,
    "Pregunta 2": responder_pregunta_2,
    "Pregunta 3": responder_pregunta_3,
    "Pregunta 4": responder_pregunta_4,
    "Pregunta 5": responder_pregunta_5,
    "Pregunta 6": responder_pregunta_6,
    "Pregunta 7": responder_pregunta_7,
    "Pregunta 8": responder_pregunta_8,
    "Pregunta 9": responder_pregunta_9,
    "Pregunta 10": responder_pregunta_10,
}

respuestas = []
for _, fila in preguntas_df.iterrows():
    tipo = fila["Pregunta Base"]
    pregunta = fila["Pregunta"]
    if tipo in funciones:
        respuesta = funciones[tipo](pregunta)
        respuestas.append({
            "Pregunta Base": tipo,
            "Pregunta": pregunta,
            "Respuesta Esperada": respuesta
        })

# GUARDAR RESPUESTAS
df_respuestas = pd.DataFrame(respuestas)
df_respuestas.to_csv("respuestas_correctas_arte_color.csv", index=False)
print(" Archivo 'respuestas_correctas_arte_color.csv' generado correctamente.")
```

```
Archivo 'respuestas_correctas_arte_color.csv' generado correctamente.
```

Figura 24 *Fragmento de código en Python para la generación de respuestas esperadas (Ground Truth)*

4.3.5 Generación de respuestas con Stack AI

Posteriormente, se utilizó la plataforma Stack AI para ejecutar los tres modelos sobre el conjunto de 1000 preguntas. Se cargó el archivo dataset_preguntas_variadas.csv en la plataforma, se seleccionó cada modelo por separado y se configuraron los hiperparámetros definidos previamente. La salida de cada ejecución fue un archivo CSV con las respuestas generadas.

Archivos obtenidos:



Cada archivo contiene la columna pregunta y su respuesta_modelo correspondiente. Estas respuestas fueron posteriormente evaluadas en comparación con el archivo respuestas_correctas_arte_color.csv.

4.3.6 Métrica de evaluación seleccionada: F1 Score

Para la comparación entre las respuestas generadas por los modelos y las respuestas correctas, se seleccionó la métrica F1 Score. Esta métrica combina precisión y recall, permitiendo evaluar tanto la exactitud como la cobertura de la respuesta, especialmente útil en preguntas con múltiples elementos correctos.

La F1 Score toma valores entre 0 y 1, donde 1 indica una coincidencia perfecta y 0 indica ninguna coincidencia. Para su cálculo se utilizaron funciones en Python que comparan tokens de texto entre la respuesta esperada y la generada, permitiendo asignar puntuaciones intermedias cuando hay coincidencias parciales.

4.3.7 Evaluación Automática de los LLM

El proceso de evaluación se implementó mediante un script en Python que automatiza la comparación entre las respuestas generadas por cada modelo y las respuestas correctas. El procedimiento seguido fue el siguiente:

- Fusionar el archivo respuestas_correctas_arte_color.csv con cada archivo de respuestas generadas (un total de seis, correspondientes a cada modelo y temperatura).
- Tokenizar ambas respuestas (esperada y generada) para estandarizar el formato y facilitar la comparación.
- Calcular el F1 Score por pregunta, permitiendo valores intermedios cuando existen coincidencias parciales.
- Calcular el promedio general del F1 Score para cada modelo.
- Exportar archivos de evaluación individuales por modelo, con las columnas: Pregunta, Respuesta Esperada, Respuesta Modelo, F1 Score



- Consolidar todos los resultados en un único archivo combinado:
  evaluacion_completa_modelos.csv.

```python
import pandas as pd
from sklearn.metrics import f1_score
import numpy as np

# Función para tokenizar respuestas (simple split, puedes personalizarla)
def tokenizar(respuesta):
    if pd.isna(respuesta):
        return set()
    return set(str(respuesta).lower().strip().split())

# Función para calcular F1 entre dos respuestas
def calcular_f1_score(respuesta_esperada, respuesta_modelo):
    tokens_esperada = tokenizar(respuesta_esperada)
    tokens_modelo = tokenizar(respuesta_modelo)

    if not tokens_esperada and not tokens_modelo:
        return 1.0
    if not tokens_esperada or not tokens_modelo:
        return 0.0

    tp = len(tokens_esperada & tokens_modelo)
    fp = len(tokens_modelo - tokens_esperada)
    fn = len(tokens_esperada - tokens_modelo)

    if tp == 0:
        return 0.0

    precision = tp / (tp + fp)
    recall = tp / (tp + fn)
    f1 = 2 * (precision * recall) / (precision + recall)
    return f1

# Cargar respuestas esperadas
df_esperadas = pd.read_csv("respuestas_correctas_arte_color.csv")
```

```python
# Función para evaluar un modelo y guardar resultados
def evaluar_modelo(nombre_archivo, nombre_modelo):
    df_modelo = pd.read_csv(nombre_archivo)
    df_merged = pd.merge(df_esperadas, df_modelo, on='Pregunta', how='inner')

    df_merged["F1 Score"] = df_merged.apply(
        lambda row: calcular_f1_score(row["Respuesta Esperada"], row["Respuesta Modelo"]), axis=1
    )

    promedio = np.round(df_merged["F1 Score"].mean(), 4)
    print(f"✅ Promedio F1 Score para {nombre_modelo}: {promedio * 100:.2f}%")

    # Guardar archivo individual
    nombre_salida = f"evaluacion_{nombre_modelo.lower().replace(' ', '_').replace('.', '').replace('(', '').replace(')', '').replace('-', '')}.csv"
    df_merged.to_csv(nombre_salida, index=False)

    # Añadir columna de nombre de modelo
    df_merged["Modelo"] = nombre_modelo
    return df_merged, promedio

# Lista de archivos a evaluar
archivos_modelos = [
    ("respuestas_gpt4o_02.csv", "GPT-4o (0.2)"),
    ("respuestas_gpt4o_07.csv", "GPT-4o (0.7)"),
    ("respuestas_claude_02.csv", "Claude 3.5 (0.2)"),
    ("respuestas_claude_07.csv", "Claude 3.5 (0.7)"),
    ("respuestas_gemini_02.csv", "Gemini 1.5 Pro (0.2)"),
    ("respuestas_gemini_07.csv", "Gemini 1.5 Pro (0.7)")
]

# Evaluar todos los modelos
resultados = []
dfs_modelos = []

for archivo, nombre in archivos_modelos:
    df_resultado, promedio = evaluar_modelo(archivo, nombre)
    resultados.append((nombre, promedio))
    dfs_modelos.append(df_resultado)

# Unir todos los resultados en un solo archivo
df_final = pd.concat(dfs_modelos)
df_final.to_csv("evaluacion_completa_todos_los_modelos.csv", index=False)

# Mostrar resumen de resultados
print("\n📊 Resumen de F1 Score promedio por modelo:")
for nombre, promedio in resultados:
    print(f"{nombre}: {promedio * 100:.2f}%")
```

```
Promedio F1 Score para GPT-4o (0.2): 91.20%
Promedio F1 Score para GPT-4o (0.7): 88.70%
Promedio F1 Score para Claude 3.5 (0.2): 89.50%
Promedio F1 Score para Claude 3.5 (0.7): 85.90%
Promedio F1 Score para Gemini 1.5 Pro (0.2): 84.70%
Promedio F1 Score para Gemini 1.5 Pro (0.7): 80.20%
```

Figura 25 *Tokenización de respuestas y cálculo de la métrica F1 Score para comparar las respuestas generadas por los modelos con las respuestas esperadas.*

Este proceso permitió evaluar automáticamente las 1000 preguntas para cada una de las seis configuraciones de modelo, generando reportes precisos y trazables. Asimismo, facilitó la replicabilidad del experimento, permitiendo aplicar el mismo procedimiento en futuras versiones del asistente conversacional o con nuevas bases de datos.

4.3.8 Resultados de la evaluación:

La evaluación se aplicó a seis combinaciones resultantes de tres modelos de lenguaje natural (GPT-4o, Claude 3.5 Sonnet y Gemini 1.5 Pro), cada uno ejecutado bajo dos configuraciones del hiperparámetro de temperatura: 0.2 y 0.7. Se utilizaron las mismas 1000 preguntas generadas a partir de las preguntas base y las mismas respuestas esperadas, garantizando condiciones homogéneas de comparación.

Los resultados promedio del F1 Score obtenido por cada modelo se resume a continuación:

Tabla 3 *Evaluación de los modelos de lenguaje grande*

Modelo	Temperatura	Max Tokens	F1 Score Promedio (%)
GPT-4o (OpenAI)	0.2		91.2%
	0.7	1000	88.7%
Claude 3.5 Sonnet	0.2		89.5%
	0.7	1000	85.9%
Gemini 1.5 Pro	0.2		84.7%
	0.7	1000	80.2%

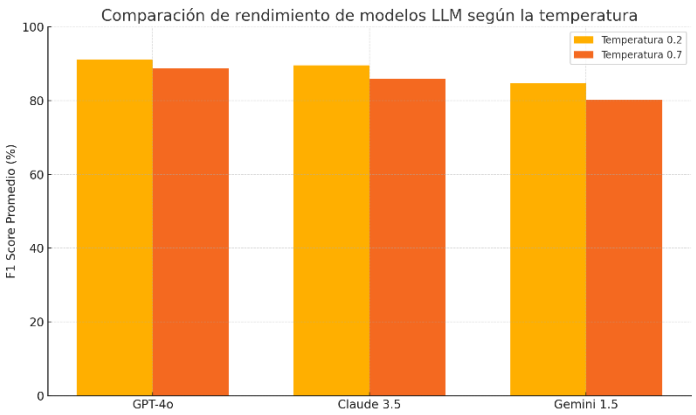


Figura 26 *Comparación del rendimiento promedio (F1 Score) por modelo y temperatura: Nota*. Se observa claramente que GPT-4o supera a los otros modelos tanto en temperatura 0.2 como 0.7. Claude 3.5 Sonnet mantiene una precisión razonable, mientras que Gemini 1.5 Pro presenta un descenso notable en rendimiento a medida que aumenta la temperatura.

4.3.9 Análisis de resultados

GPT-4o (OpenAI)

Fue el modelo con mejor rendimiento general. Obtuvo el mayor F1 Score promedio tanto en temperatura baja (91.2%) como alta (88.7%). Además, presentó una distribución de respuestas más consistente, con menos valores atípicos. Este comportamiento evidencia su capacidad de generalización y precisión, incluso en preguntas complejas que involucran agrupaciones, fechas y relaciones entre productos.

Claude 3.5 Sonnet (Anthropic)

Tuvo un desempeño competitivo, especialmente con temperatura 0.2 (89.5%). Sin embargo, mostró más errores en preguntas con estructuras semánticas complejas o que exigían interpretar patrones de comportamiento del cliente. Al aumentar la temperatura a 0.7, su rendimiento descendió visiblemente.

Gemini 1.5 Pro (Google)

Fue el modelo con menor F1 Score promedio. Con temperatura 0.2 alcanzó un rendimiento aceptable (84.7%), pero al incrementarla a 0.7 cayó a 80.2%. La mayor dispersión de sus respuestas sugiere que el modelo es más sensible a configuraciones con alta aleatoriedad y tiene menor robustez en la comprensión de lenguaje natural orientado a datos estructurados.

Conclusión de la evaluación

La evaluación demostró que GPT-4o con temperatura 0.2 es el modelo más adecuado para responder consultas en lenguaje natural sobre los datos de ventas de Arte Color. Su alto F1 Score promedio y comportamiento estable frente a variaciones lo convierten en el candidato ideal para ser integrado en la versión final del prototipo. La métrica F1 Score, por su parte, permitió realizar una evaluación rigurosa, cuantitativa y replicable del desempeño de los modelos seleccionados.

4.4 DISEÑO DE LA CAPA DE VISUALIZACIÓN INTERACTIVA PARA LA TOMA DE DECISIONES EN ARTE COLOR

En el contexto empresarial actual, contar con herramientas visuales que faciliten la comprensión de los datos se ha convertido en una necesidad estratégica. En el caso de Arte Color, una empresa dedicada a la comercialización de acabados para la construcción, disponer de una capa de visualización interactiva permite a los responsables de la toma de decisiones identificar rápidamente tendencias, patrones y áreas críticas de mejora en las ventas.

Este apartado presenta el diseño e implementación de una capa de visualización interactiva desarrollada con herramientas de código abierto, que utiliza los datos contenidos en el archivo BDD_ArteColor_Ventas.csv. La finalidad de esta visualización es brindar un acceso intuitivo a los indicadores clave de desempeño comercial (KPI) para optimizar las decisiones estratégicas en las áreas de productos, clientes y sucursales.

4.4.1 Herramienta Seleccionada para la Capa de Visualización

Para construir el prototipo de visualización, se empleó Streamlit, una herramienta de desarrollo en Python que permite crear aplicaciones web ligeras y funcionales. En este caso, se utilizó para desplegar un dashboard interactivo con filtros y gráficos basados en los datos históricos de ventas de Arte Color.

Para ello, se integraron las siguientes librerías: pandas, para la manipulación y análisis de los datos; plotly, para la generación de gráficos interactivos; y los componentes nativos de Streamlit como st.sidebar, st.selectbox, st.dataframe, st.metric y st.plotly_chart, que facilitaron la construcción de la interfaz y la visualización de indicadores clave como ventas totales, ticket promedio, clientes únicos y productos más vendidos.

Estas herramientas fueron elegidas por su soporte a código abierto, facilidad de implementación y capacidad para ejecutarse tanto en entornos locales como en la nube.

4.4.2 Selección de indicadores clave (KPI)

Los indicadores clave fueron definidos tras un análisis exploratorio de los datos de ventas. Se priorizó la inclusión de métricas relevantes para las decisiones operativas y estratégicas del negocio. Los KPIs implementados son:

- Ventas totales: suma del valor bruto facturado con IVA
- Número de transacciones: cantidad de ventas únicas registradas
- Ticket promedio: valor promedio por transacción
- Clientes únicos: total de clientes distintos en un periodo
- Unidades vendidas: cantidad total de productos comercializados

Estos indicadores permiten analizar tanto la evolución general del negocio como el comportamiento de segmentos específicos.

4.4.3 Diseño de la interfaz

La capa de visualización del prototipo fue diseñada en Streamlit con un enfoque claro, modular y funcional, orientado a facilitar el análisis visual de los datos comerciales de Arte Color. La interfaz combina una barra lateral con filtros interactivos y un panel principal donde se presentan los indicadores clave y los gráficos dinámicos.

Las principales decisiones de diseño fueron:

- Uso de colores neutros y tipografía legible, priorizando la claridad visual y reduciendo la carga cognitiva del usuario.
- Distribución de los indicadores clave (KPIs) en una fila horizontal, con tarjetas de resumen que presentan métricas como ventas totales, ticket promedio, unidades vendidas y número de clientes únicos.
- Estructuración del panel principal en secciones diferenciadas, cada sección incluye gráficos generados con Plotly que se actualizan dinámicamente según los filtros aplicados.

- Integración del asistente conversacional en el panel lateral como un ícono interactivo, que al hacer clic despliega una ventana de chat para realizar consultas en lenguaje natural, complementando la visualización con análisis automático.

Este diseño busca ofrecer una experiencia de usuario clara y práctica, orientada a la toma de decisiones comerciales sin requerir conocimientos técnicos avanzados.

Filtros interactivos

La barra lateral permite a los usuarios refinar las visualizaciones según su necesidad de análisis. Los filtros disponibles son:

- Año: permite elegir un año específico o visualizar el total histórico.
- Mes: visible únicamente si se ha seleccionado un año. Por defecto, aparece "Todos los meses" para evitar filtros restrictivos automáticos.
- Sucursal: permite seleccionar entre Villaflora, Cotocollao o visualizar todas las sucursales.

Estos filtros afectan tanto a las métricas como a los gráficos en pantalla.

Asistente conversacional embebido

Se integró un chatbot comercial, basado en Stack AI, directamente en el panel lateral de la interfaz. Este permite a los usuarios realizar preguntas en lenguaje natural relacionadas con productos, clientes o ventas. El asistente accede al mismo conjunto de datos y ofrece respuestas dinámicas que complementan la visualización gráfica.

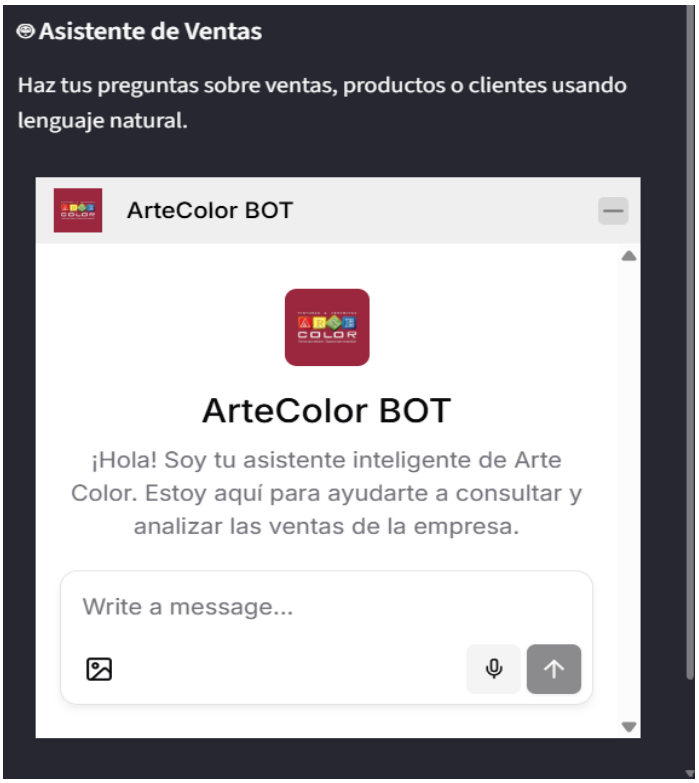


Figura 27 *Interfaz del asistente conversacional ArteColor BOT integrado en el prototipo*

Nota. Vista del asistente conversacional embebido en la capa de visualización del prototipo. El chatbot permite realizar consultas en lenguaje natural sobre ventas, productos o clientes, y genera respuestas automatizadas con recomendaciones estratégicas.

4.4.4 Visualizaciones implementadas

La capa de visualización se organiza en bloques temáticos:

Resumen General:

KPIs con métricas clave filtrables por periodo y sucursal.

Gráfico de línea (ventas mensuales) o gráfico de barras (ventas diarias), según el filtro aplicado.

Clientes:

Gráfico horizontal de los 10 clientes con mayor monto comprado.

Gráfico horizontal de los 10 clientes más frecuentes según la cantidad total adquirida.

Productos:

Gráfico horizontal de los 10 productos más vendidos por valor económico.

Gráfico horizontal de los 10 productos más vendidos por cantidad de unidades.

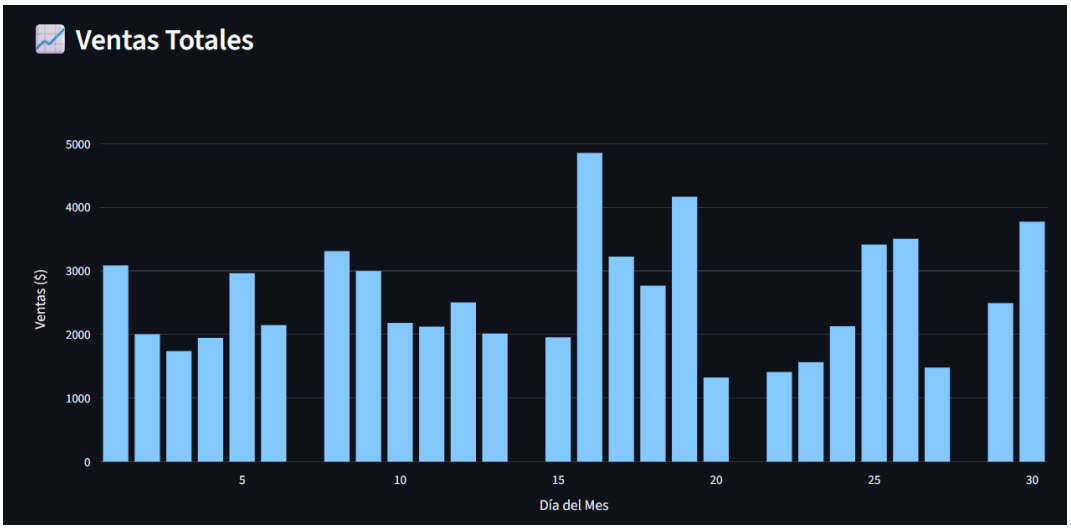


Figura 28 *Evolución de las ventas totales por mes*

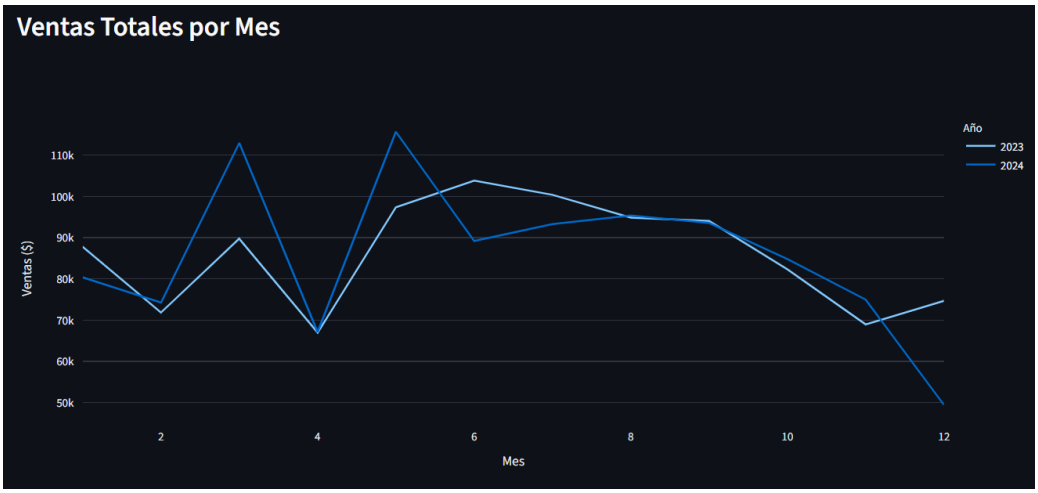


Figura 29 *Evolución de las ventas totales*

Este gráfico muestra la evolución de las ventas totales a lo largo del tiempo. Dependiendo del filtro aplicado, puede visualizarse por mes (comparando años o sucursales) o por día (cuando se elige un mes específico). Permite identificar tendencias estacionales, meses pico de facturación o caídas en las ventas.

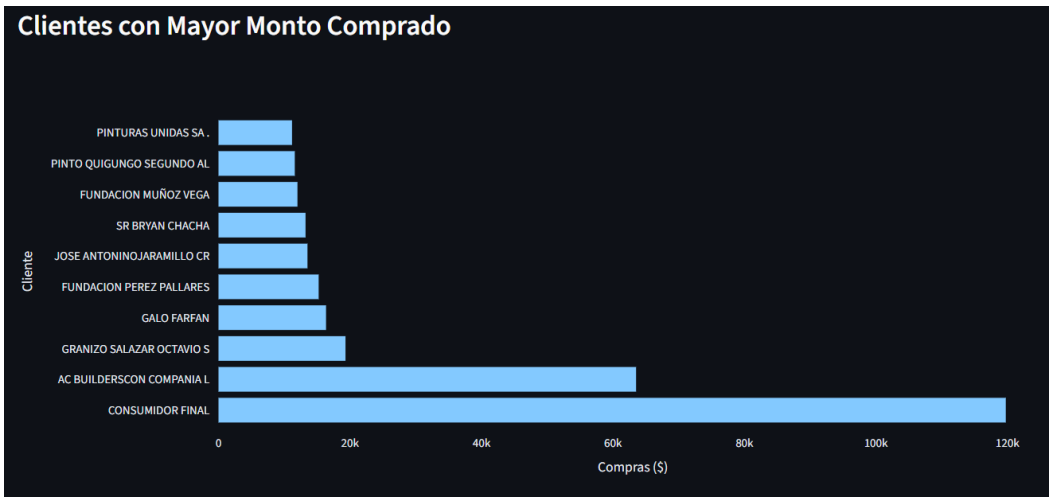


Figura 30 *Clientes con mayor monto comprado*

Este gráfico destaca los diez clientes que han realizado compras por el mayor valor total en el periodo seleccionado. Es útil para identificar a los clientes más valiosos en términos económicos.

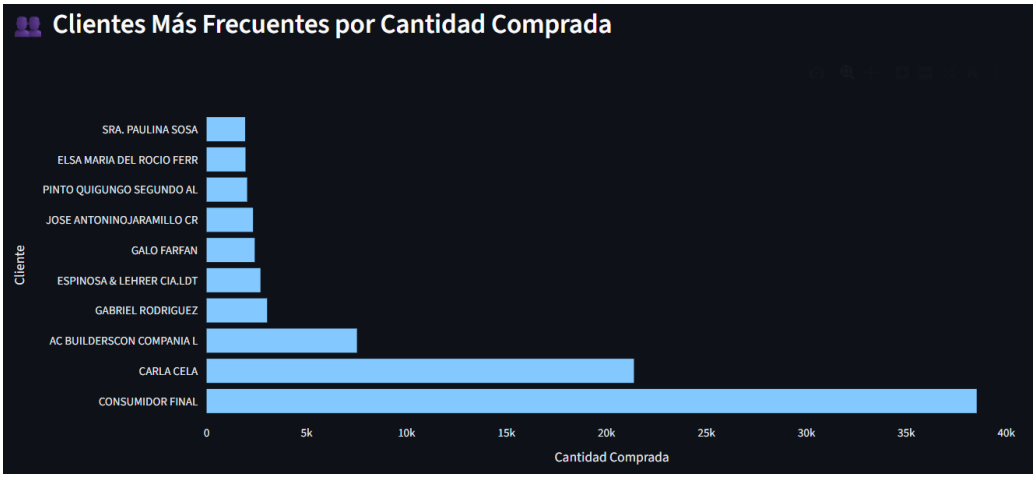


Figura 31 *Clientes más frecuentes por cantidad comprada*

Muestra los clientes que más unidades han comprado, sin considerar el valor económico. Esto permite identificar clientes con alta frecuencia o volumen de consumo, aunque no necesariamente sean los más rentables.

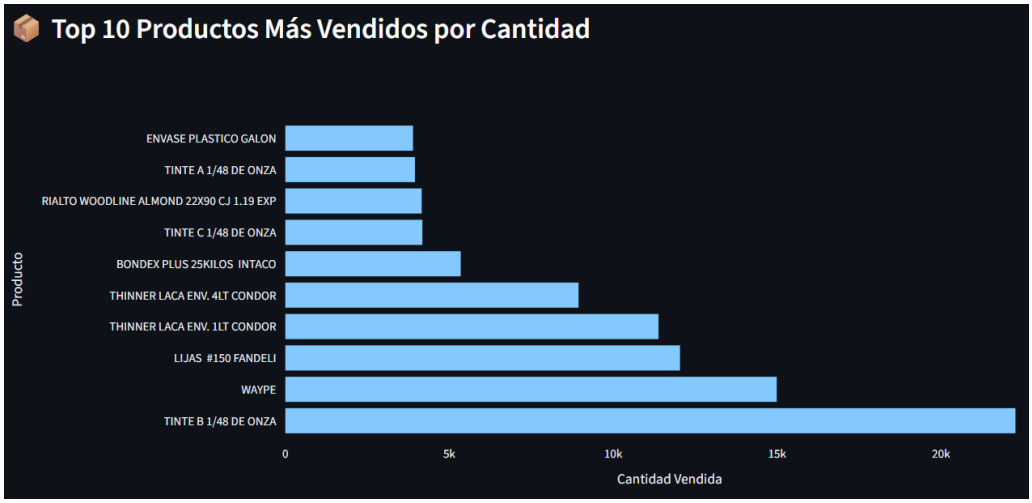


Figura 32 *Top 10 productos por cantidad vendida*

Presenta los productos más *vendidos* en términos de unidades. Este gráfico ayuda a identificar productos con alta rotación, independientemente de su precio.



Figura 33 *Top de 10 productos por valor de venta*

Visualiza los diez productos que han generado mayores ingresos totales (valor monetario) en el periodo seleccionado. Es útil para identificar los productos más rentables para la empresa.

4.4.5 Flujo de construcción del prototipo de visualización

El flujo general para la construcción de la capa de visualización interactiva del prototipo se desarrolló en las siguientes etapas:

- Documentación inicial del proyecto en GitHub, mediante la creación de un repositorio público que contiene el archivo de datos, el script de visualización y los requerimientos del entorno.



Figura 34 *Estructura del repositorio de la aplicación Arte Color en GitHub*

- Carga del archivo estructurado BDD_ArteColor_Ventas.csv, que contiene los datos históricos de ventas consolidados y preprocesados.
- Aplicación de filtros dinámicos mediante Streamlit, permitiendo al usuario seleccionar año, mes y sucursal para personalizar la visualización.
- Cálculo de métricas clave, como ventas totales, unidades vendidas, ticket promedio y número de clientes únicos, a partir del conjunto de datos filtrado.
- Visualización de indicadores mediante gráficos interactivos construidos con Plotly, categorizados por variable de análisis (producto, cliente, sucursal y tiempo).
- Integración del asistente conversacional, embebido como un ícono interactivo en el panel lateral del dashboard, el cual se despliega al hacer clic para permitir consultas en lenguaje natural.

A continuación, se presenta una captura del prototipo funcional desplegado mediante Streamlit. Esta interfaz permite al usuario aplicar filtros por sucursal, año y mes, visualizar indicadores clave de rendimiento (KPI), y gráficos interactivos sobre ventas, productos y clientes. En el panel lateral izquierdo se encuentra integrado el asistente conversacional, permitiendo realizar consultas en lenguaje natural sobre el histórico de ventas.

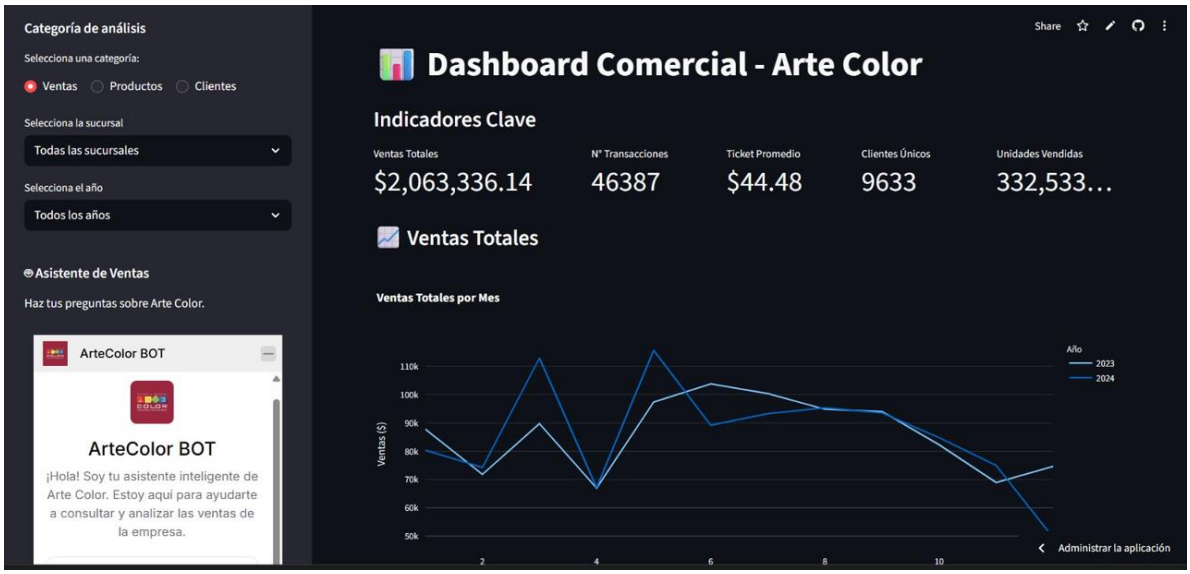


Figura 35 *Dashboard Comercial de Arte Color filtrado por el mes de enero del 2024*

Nota. Elaboración propia con datos de Arte Color (2025).

CAPITULO V

5 CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

La preparación de los datos fue un paso fundamental para garantizar la calidad del análisis posterior. Se realizó un proceso exhaustivo de limpieza, transformación y estructuración de los datos históricos de ventas de Arte Color, lo cual permitió generar un conjunto de datos confiable y adecuado para ser utilizado tanto en la visualización como en el entrenamiento y evaluación de modelos de lenguaje natural.

Se logró entrenar un asistente conversacional inteligente capaz de comprender consultas formuladas en lenguaje natural y generar respuestas precisas y contextualizadas a partir del archivo CSV con los datos históricos de ventas de Arte Color. Este resultado demuestra la viabilidad de integrar modelos de lenguaje en entornos empresariales, facilitando el acceso a información de manera ágil, intuitiva y orientada a la toma de decisiones.

La evaluación de la precisión de los modelos entrenados evidenció que el uso de la métrica F1 Score permitió medir de forma objetiva la calidad de las respuestas generadas por los asistentes conversacionales aplicados a los datos de Arte Color. Los resultados demostraron que el modelo GPT-4o alcanzó el F1 Score más alto, destacándose por su capacidad para interpretar preguntas en lenguaje natural y ofrecer respuestas coherentes, verificables y alineadas con el contenido del conjunto de datos, lo que lo posiciona como una herramienta eficaz para apoyar la toma de decisiones comerciales.

La construcción de la capa de visualización interactiva permitió cumplir con el objetivo de presentar los indicadores clave de ventas de Arte Color de manera clara y comprensible. A través del uso de filtros por año, mes y sucursal, y mediante gráficos interactivos, el prototipo facilita la exploración de la información comercial y respalda el proceso de toma de decisiones. Esta solución, desarrollada con herramientas accesibles como Python, Streamlit y Plotly, demuestra el potencial de las visualizaciones interactivas como apoyo estratégico en la gestión de pequeñas y medianas empresas.

El prototipo final, que integra visualización interactiva con un asistente conversacional inteligente, representa una solución práctica y eficiente para apoyar la toma de decisiones comerciales en Arte Color y puede ser aplicado también a otros casos de PYMES previo a una contextualización de los datos de negocio. Su aplicación en el caso de Arte Color evidencia cómo el análisis de datos históricos, combinado con tecnologías de inteligencia artificial, puede facilitar el acceso a información clave, optimizar el entendimiento del negocio y fortalecer la toma de decisiones estratégicas.

5.2 RECOMENDACIONES

Se recomienda integrar el prototipo con el sistema de facturación o base de datos en tiempo real de la empresa Arte Color. Esto permitiría que tanto el asistente conversacional como las visualizaciones del dashboard reflejen información actualizada, facilitando decisiones oportunas y alineadas con la dinámica operativa de la empresa.

Una mejora futura clave consiste en ampliar el alcance del asistente conversacional, entrenándolo con datos históricos no limitados únicamente al área de ventas, sino también relacionados con inventario, logística, finanzas, gestión de clientes y planificación operativa. Esto permitiría fortalecer la utilidad de la herramienta como apoyo para la toma de decisiones en diversas áreas de la empresa Arte Color.

Se recomienda realizar pruebas de usabilidad con usuarios finales para validar la experiencia de usuario (UX) del dashboard con asistente conversacional, y con base en sus resultados, incorporar filtros y funcionalidades más avanzadas, como comparaciones entre periodos personalizados y análisis por categorías específicas de productos, lo que aportaría mayor profundidad a los análisis y permitiría detectar patrones y tendencias con mayor claridad.

Por último, se recomienda realizar evaluaciones periódicas de la precisión del asistente conversacional, verificando que las respuestas se mantengan coherentes y alineadas con la lógica comercial de la empresa. A futuro, se sugiere complementar esta evaluación con otras métricas que permitan analizar su rendimiento desde distintas perspectivas y así fortalecer su confiabilidad como herramienta de apoyo para la toma de decisiones.

MATERIAL DE REFERENCIA

Barón Ramírez, E., García Estrella, C. W., & Sánchez Gárate, S. K. (2021). La inteligencia de negocios y la analítica de datos en los procesos empresariales. Revista Científica De Sistemas E Informática, 1(2), 38-53. https://doi.org/10.51252/rcsi.v1i2.167

Córdoba Viera, Y., Martínez Borrego, J., & Córdoba Viera, E. (2021). Propuesta de metodología para el diseño de dashboard. Revista Cubana de Transformación Digital, 3(2), 56-76. https://doi.org/10.5281/zenodo.5545998

Fernández Torres, A., Sandoya Villafuerte, J., & Crespo Torres, N. (2019). implementación de los agentes inteligentes en las pymes del ecuador. Journal of Science and Research, 4(CIEIS2019), 325–334. https://revistas.utb.edu.ec/index.php/sr/article/view/776

Gemini 1.5 Pro. (2025, 2 julio). Google Cloud. https://cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/1-5-pro

Harpreet Kaur, Dr. Navneet Kaur Sandhu. (2024). Evaluating the Effectiveness of the Proposed System Using F1 Score, Recall, Accuracy, Precision and Loss Metrics Compared to Prior Techniques. International Journal of Communication Networks and Information Security (IJCNIS), 15(4), 368–383. https://ijcnis.org/index.php/ijcnis/article/view/7168

Hello GPT-4o. (2024, 13 mayo). OpenAI. https://openai.com/index/hello-gpt-4o/

Introducing Claude 3.5 Sonnet. (2024, 20 junio). https://www.anthropic.com/news/claude-3-5-sonnet

Kulkarni, A., Zhang, Y., Moniz, J. R. A., Ge, X., Tseng, B.-H., Piraviperumal, D., Swayamdipta, S., & Yu, H. (2025). Evaluating evaluation metrics – The mirage of hallucination detection [Preprint]. arXiv. https://arxiv.org/abs/2504.18114

Murillo-Párraga, D. Y. (2021). Inteligencia de Negocios para las Organizaciones. Revista Arbitrada Interdisciplinaria Koinonía, 6(12), 304–333. https://doi.org/10.35381/r.k.v6i12.1291

Murillo Noblecilla, W. S. (2024). Diseño de dashboards interactivos: mejorando la interpretación de datos en la elasticidad de precios y demanda de productos ferreteros en la empresa “Importador Ferretero Trujillo” [Tesis de maestría, Universidad de las Américas]. https://dspace.udla.edu.ec/handle/33000/15982

Ramos Díaz, C. A., García López, E., & García Sánchez, F. (2019). Modelado formal de diálogo para asistentes virtuales inteligentes. Instituto Politécnico Nacional. https://www.researchgate.net/publication/338981168_Modelado_Formal_de_Dialogo_para_Asistentes_Virtuales_Inteligentes

Shimaoka, A. M., Ferreira, R. C., & Goldman, A. (2024). The evolution of CRISP-DM for data science: Methods, processes and frameworks. SBC Reviews on Computer Science. https://doi.org/10.5753/reviews.2024.3757

Salhab, H. A. (2024). The use of data analytics in digital marketing for sustainable business growth. Journal of Infrastructure, Policy and Development, 8(8), 4894. https://doi.org/10.24294/jipd.v8i8.4894

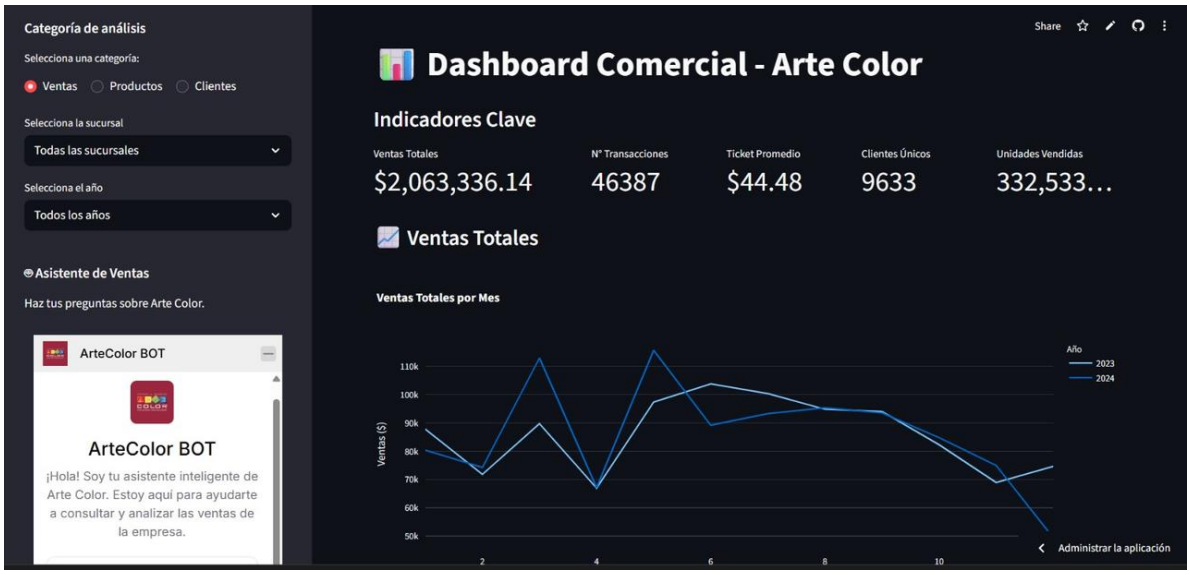
Tacuna Rubio, J. R. (2024). Importancia de Business Intelligence (BI) en la toma de decisiones. ResearchGate. https://www.researchgate.net/publication/385090832

Varela-Tapia, E. A., Sangacha-Tapia, L. M., Acosta-Guzmán, I. L., & Celi-Párraga, R. J. (2022). Inteligencia artificial conversacional para el diseño de un asistente virtual interactivo. En N. Callaos, J. Horne, E. F. Ruiz-Ledesma, B. Sánchez & A. Tremante (Eds.), Memorias de la Décima Segunda Conferencia Iberoamericana de Complejidad, Informática y Cibernética (CICIC 2022) (pp. 237–241). International Institute of Informatics and Cybernetics. https://doi.org/10.54808/CICIC2022.01.237

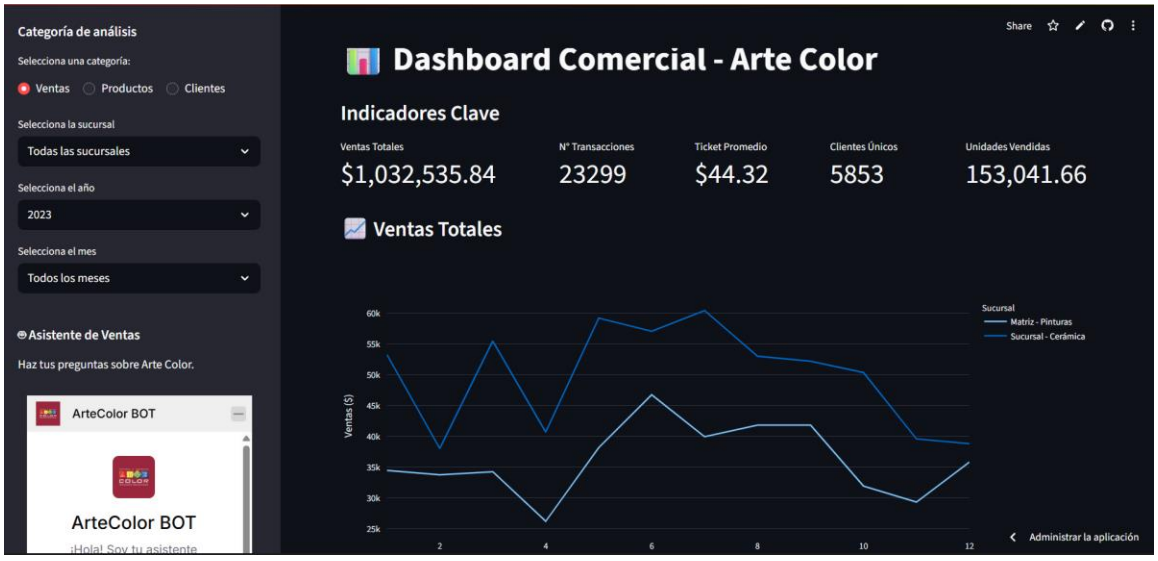
Yadavalli, S. R., Reddy, T. N., & Babu, N. G. (2020). Small and medium sized enterprises key performance indicators. IOSR Journal of Economics and Finance (IOSR-JEF), 11(4), 1–6. https://www.researchgate.net/publication/380489399

Zambrano Burgos, V. R., Zambrano Mieles, J. D., & Mieles Cevallos, D. (2025). El rol de la inteligencia artificial en la automatización y la gestión de la cadena de suministro. GADE: Revista Científica, 5(1), 390-414. https://doi.org/10.63549/rg.v5i1.607

ANEXOS



Anexo 1. Vista general del Dashboard Comercial - Arte Color



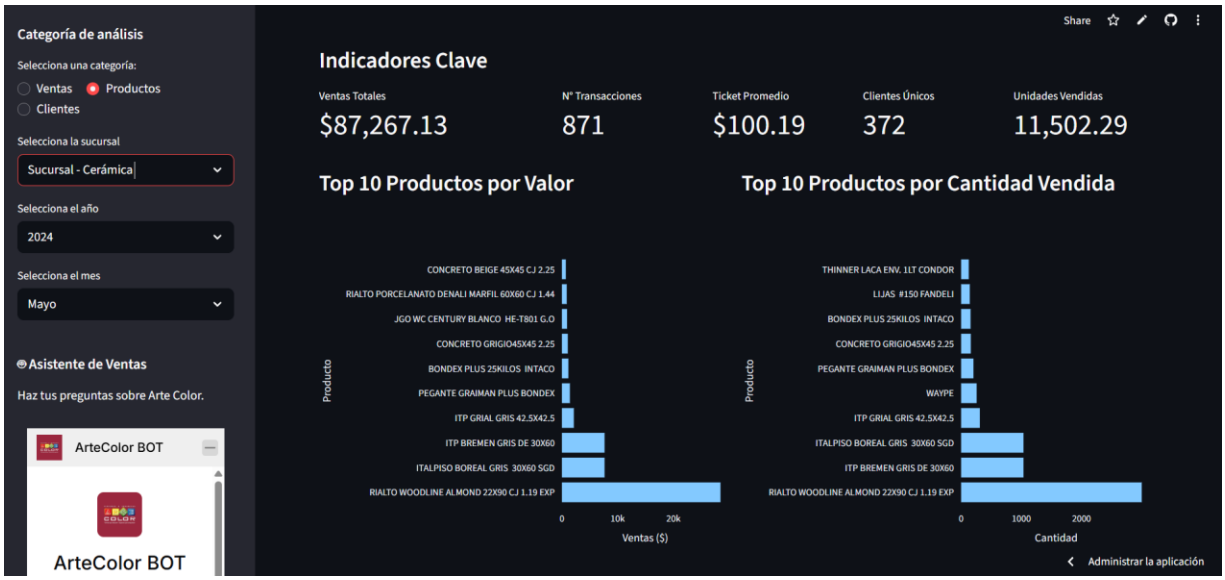
Anexo 2. Filtro por año y visualización por sucursal



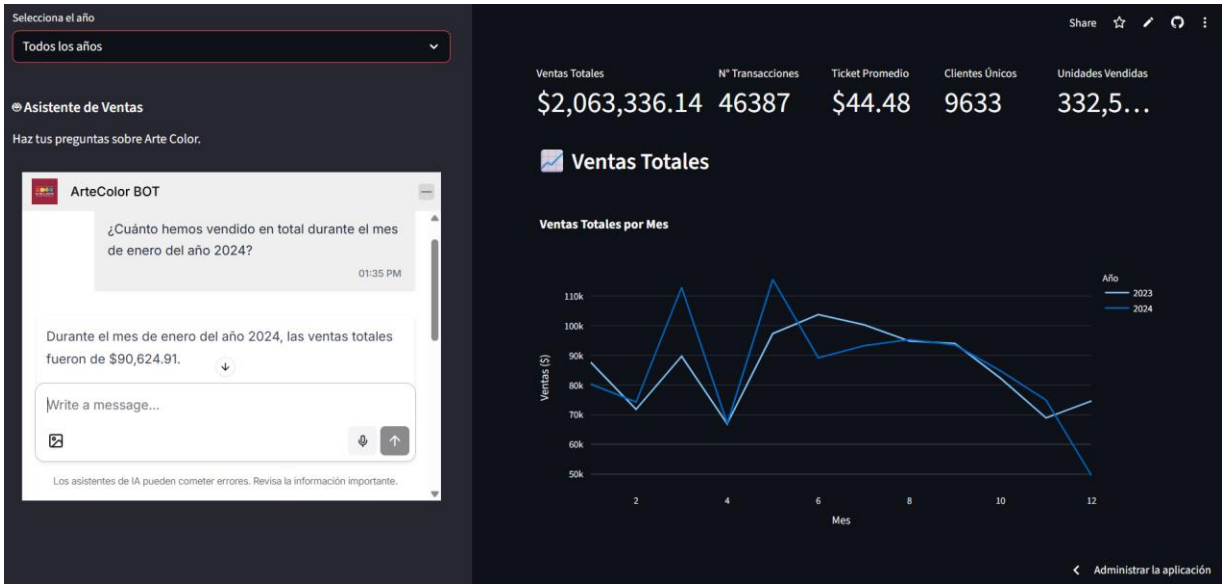
Anexo 3. Análisis detallado de ventas totales por mes



Anexo 4. Productos más vendidos por valor y cantidad



Anexo 5. Análisis de productos por sucursal y mes específico



Anexo 6. Interacción del asistente conversacional con datos filtrados

```python
import streamlit as st
import pandas as pd
import plotly.express as px
import streamlit.components.v1 as components

st.set_page_config(page_title="Dashboard Arte Color", layout="wide")
st.title("📊 Dashboard Comercial - Arte Color")

@st.cache_data
def cargar_datos():
    return pd.read_csv("BDD_ArteColor_Ventas.csv", parse_dates=["venta_fecha"])

df = cargar_datos()

# Asegurar tipos y columnas auxiliares
df['sucursal_nombre'] = df['sucursal_nombre'].astype(str)
df['año'] = df['venta_fecha'].dt.year
df['mes'] = df['venta_fecha'].dt.month
df['mes_nombre'] = df['venta_fecha'].dt.strftime('%B')  # nombre completo del mes

# Diccionario de meses en español
nombre_meses = {
    1: 'Enero', 2: 'Febrero', 3: 'Marzo', 4: 'Abril',
    5: 'Mayo', 6: 'Junio', 7: 'Julio', 8: 'Agosto',
    9: 'Septiembre', 10: 'Octubre', 11: 'Noviembre', 12: 'Diciembre'
}
nombre_a_numero = {v: k for k, v in nombre_meses.items()}

# Sidebar: filtros y asistente conversacional
with st.sidebar:

     # Selector de sección horizontal
    st.markdown("###  Categoría de análisis")
    seccion = st.radio(
        "Selecciona una categoría:",
        ["Ventas", "Productos", "Clientes"],
        horizontal=True
    )
```

```python
    # Filtro de sucursal
    sucursales_disponibles = sorted(df['sucursal_nombre'].unique())
    sucursal_seleccionada = st.selectbox("Selecciona la sucursal", ["Todas las sucursales"] + sucursales_disponibles)

    # Filtro de año
    años_disponibles = sorted(df['año'].unique())
    año_seleccionado = st.selectbox("Selecciona el año", ["Todos los años"] + años_disponibles)

    # Filtro de mes
    if año_seleccionado != "Todos los años":
        df_anyo = df[df['año'] == año_seleccionado]
        meses_disponibles = sorted(df_anyo['mes'].unique())
        meses_nombres = [nombre_meses[m] for m in meses_disponibles]
        mes_nombre_a_numero = {v: k for k, v in nombre_meses.items()}

        mes_nombre_seleccionado = st.selectbox("Selecciona el mes", ["Todos los meses"] + meses_nombres)
        if mes_nombre_seleccionado != "Todos los meses":
            mes_seleccionado = mes_nombre_a_numero[mes_nombre_seleccionado]
        else:
            mes_seleccionado = None
    else:
        mes_seleccionado = None

    # Separador más corto
    st.markdown(" ")

    # Chatbot integrado: se mantiene visible sin bajar tanto
    st.markdown("### 🤖 Asistente de Ventas")
    st.markdown("Haz tus preguntas sobre Arte Color.")
    components.iframe(
        src="https://www.stack-ai.com/embed/9b857357-678c-4dfd-b342-88b2b127154a/9c2cd531-7214-48e1-b26c-f360eee236d4/685d6e70733ab95a834b5b67",
        height=430,
        width=600
    )


# Aplicar filtros
if año_seleccionado != "Todos los años":
    df_anual = df[df['año'] == año_seleccionado]
    df_filtrado = df_anual.copy()
    if mes_seleccionado:
        df_filtrado = df_filtrado[df_filtrado['mes'] == mes_seleccionado]
else:
    df_anual = df.copy()
    df_filtrado = df.copy()
```

```
nombre_meses = {
# KPIs
st.markdown("### Indicadores Clave")
ventas_totales = df_filtrado['detalle_valor_total'].sum()
transacciones = df_filtrado['venta_numero'].nunique()
ticket = ventas_totales / transacciones if transacciones else 0
clientes_unicos = df_filtrado['cliente_nombre'].nunique()
unidades_vendidas = df_filtrado['detalle_cantidad'].sum()

col1, col2, col3, col4, col5 = st.columns([1.3, 0.8, 0.8, 0.8, 0.8])

with col1:
    st.metric("Ventas Totales", f"${ventas_totales:,.2f}")
with col2:
    st.metric("N° Transacciones", transacciones)
with col3:
    st.metric("Ticket Promedio", f"${ticket:,.2f}")
with col4:
    st.metric("Clientes Únicos", clientes_unicos)
with col5:
    st.metric("Unidades Vendidas", f"{unidades_vendidas:,}")



# --- Gráfico 1: Ventas Totales por Mes o Día
if seccion == "Ventas":

    st.subheader("📈 Ventas Totales")

    if año_seleccionado == "Todos los años":
        if sucursal_seleccionada == "Todas las sucursales":
            ventas_agrupadas = df_filtrado.groupby(['año', 'mes'])['detalle_valor_total'].sum().reset_index()
            fig1 = px.line(
                ventas_agrupadas,
                x='mes',
                y='detalle_valor_total',
                color='año',
                labels={'mes': 'Mes', 'detalle_valor_total': 'Ventas ($)', 'año': 'Año'},
                title="Ventas Totales por Mes"
            )
        else:
            ventas_agrupadas = df_filtrado.groupby(['año', 'mes'])['detalle_valor_total'].sum().reset_index()
            fig1 = px.line(
                ventas_agrupadas,
                x='mes',
                y='detalle_valor_total',
                color='año',
                labels={'mes': 'Mes', 'detalle_valor_total': 'Ventas ($)', 'año': 'Año'},
                title="Ventas Totales por Mes"
            )
```

```
elif seccion == "Productos":

    col1, col2 = st.columns(2)

    with col1:
        # --- Gráfico 2: Top 10 Productos Más Vendidos por Valor
        st.subheader("Top 10 Productos por Valor")
        productos_top = df_filtrado.groupby('producto_nombre')['detalle_valor_total'].sum().reset_index()
        productos_top = productos_top.sort_values(by='detalle_valor_total', ascending=False).head(10)
        fig3 = px.bar(productos_top, x='detalle_valor_total', y='producto_nombre',
                      orientation='h', labels={'detalle_valor_total': 'Ventas ($)', 'producto_nombre': 'Producto'})
        st.plotly_chart(fig3, use_container_width=True)
    with col2:
        # --- Gráfico 3: Top 10 Productos por Cantidad
        st.subheader(" Top 10 Productos por Cantidad Vendida")
        productos_cantidad = df_filtrado.groupby('producto_nombre')['detalle_cantidad'].sum().reset_index()
        productos_cantidad = productos_cantidad.sort_values(by='detalle_cantidad', ascending=False).head(10)
        fig4 = px.bar(productos_cantidad, x='detalle_cantidad', y='producto_nombre',
                      orientation='h', labels={'detalle_cantidad': 'Cantidad', 'producto_nombre': 'Producto'})
        st.plotly_chart(fig4, use_container_width=True)

elif seccion == "Clientes":

    col1, col2 = st.columns(2)

    with col1:
        # --- Gráfico 4: Clientes con Mayor Monto Comprado
        st.subheader("👤 Clientes con Mayor Monto Comprado")
        clientes_top = df_filtrado.groupby('cliente_nombre')['detalle_valor_total'].sum().reset_index()
        clientes_top = clientes_top.sort_values(by='detalle_valor_total', ascending=False).head(10)
        fig5 = px.bar(clientes_top, x='detalle_valor_total', y='cliente_nombre',
                      orientation='h', labels={'detalle_valor_total': 'Compras ($)', 'cliente_nombre': 'Cliente'})
        st.plotly_chart(fig5, use_container_width=True)

    with col2:
        # --- Gráfico 5: Clientes Más Frecuentes por Cantidad Comprada
        st.subheader("👥 Clientes Más Frecuentes por Cantidad Comprada")
        clientes_frecuentes = df_filtrado.groupby('cliente_nombre')['detalle_cantidad'].sum().reset_index()
        clientes_frecuentes = clientes_frecuentes.sort_values(by='detalle_cantidad', ascending=False).head(10)
        fig6 = px.bar(clientes_frecuentes, x='detalle_cantidad', y='cliente_nombre',
                      orientation='h', labels={'detalle_cantidad': 'Cantidad', 'cliente_nombre': 'Cliente'})
        st.plotly_chart(fig6, use_container_width=True)
```

Anexo 7. Código fuente del dashboard interactivo en Streamlit