

Reporte técnico

Proyecto: Plataforma de Análisis de Sentimientos en Tiempo Real para E-commerce

Autores: David Alvarez Alvarez, Marcelo Castillo Valverde, Alejandro Castro Nasevilla, Lady Garces Velasco, Daniela Pezantez Chimbo, María Mercedes Vera Letamendi.

Materia: Inteligencia Artificial

Fecha: 18/08/2025

Plataforma de Análisis de Sentimientos en Tiempo Real para E-commerce

1. Resumen Ejecutivo

El presente proyecto presenta el diseño e implementación de una plataforma de análisis de sentimientos en tiempo real aplicada al sector de comercio electrónico. El sistema integra técnica de Procesamiento de Lenguaje (NLP), algoritmos de machine learning y tecnologías big data como Apache Kafka, con el objetivo de procesar de manera continua la reseña de los consumidores.

Se entrenaron modelos clásicos (Regresión Logística y Árbol de Decisión) y avanzados (BERT/TinyBERT) para la clasificación de sentimientos, además de un clasificador de urgencia y un módulo de Aspect-Based Sentiment Analysis (ABSA) enfocado en los aspectos precio, calidad y envío.

Los resultados se integraron en un dashboard interactivo con indicadores clave como el porcentaje de reseñas negativas, la proporción de comentarios con urgencia alta, los principales aspectos problemáticos y la tendencia temporal de las opiniones.

El sistema constituye una herramienta de apoyo a la toma de decisiones, permitiendo a las empresas mejorar la gestión de calidad y la experiencia del cliente.

2. Introducción

El comercio electrónico genera diariamente un volumen masivo de reseñas y comentarios de clientes. Estas opiniones reflejan la percepción sobre la calidad de los productos, la atención y los servicios logísticos, convirtiéndose en un insumo estratégico para las empresas. Sin embargo, el análisis manual de estos datos resulta inviable debido a su magnitud y velocidad de generación.

El análisis de sentimientos, como rama del NLP, permite procesar de manera automática estas reseñas y clasificarlas en categorías emocionales (positivas, negativas o neutras). De forma complementaria, el Aspect-Based Sentiment Analysis (ABSA) posibilita identificar atributos específicos, como el precio, la calidad o el envío, y determinar el sentimiento asociado a cada uno de ellos.

Este proyecto se centra en el desarrollo de una plataforma capaz de procesar reseñas en tiempo real mediante la integración de Apache Kafka como sistema de mensajería, y diferentes modelos de NLP. El sistema final se acompaña de un dashboard interactivo que presenta métricas clave para la toma de decisiones en e-commerce.

3. Objetivo General y específicos

Desarrollar una plataforma de análisis de sentimientos en tiempo real para reseñas de clientes en e-commerce, utilizando modelos de NLP e integrando tecnologías de big data.

3.1 Objetivos Específicos

- Preprocesar y limpiar un dataset de reseñas de Amazon.
- Realizar un análisis exploratorio de datos.
- Entrenar y comparar modelos baseline (Regresión Logística, Árbol de Decisión) y avanzados (BERT).
- Implementar ABSA para los aspectos de precio, calidad y envío.
- Integrar un pipeline en tiempo real con Kafka.
- Desarrollar un dashboard interactivo con indicadores clave.

Este sistema tendrá como funcionalidad clasificar el sentimiento de las reseñas como positivo, negativo o neutro, identificar aspectos específicos mencionados en el texto, como el precio, la calidad o el envío, y finalmente asignar un nivel de urgencia a los problemas detectados, priorizando aquellos que requieren una respuesta inmediata.

4. Solución de Inteligencia Artificial (IA) y componentes técnicos

El proyecto se basó en el desarrollo y evaluación de tres modelos principales de

machine learning:

- **Modelo de Línea de Base:** Un modelo inicial que utiliza la técnica TF-IDF para la extracción de características y un clasificador de Regresión Logística. Este modelo sirvió como punto de referencia para la evaluación.
- **Modelo de Transformador:** Se implementó y evaluó un modelo avanzado basado en la arquitectura BERT-Tiny. Los resultados de la evaluación se basan en un modelo similar, Multilingual DistilBERT.
- **Modelo ABSA (Análisis de Sentimientos Basado en Aspectos):** Un modelo personalizado que identifica el sentimiento asociado a aspectos específicos del producto, como la "calidad" o el "envío".

A su vez, se utilizó Apache Kafka y Spark Streaming para el procesamiento en tiempo real.

5. Dataset

El dataset principal utilizado en este proyecto proviene del archivo Cell_Phones_and_Accessories.json.gz, que contiene reseñas de clientes sobre teléfonos móviles y accesorios disponibles en Amazon. Este archivo en formato JSON Lines reúne más de 80,000 reseñas originales, cada una acompañada de una puntuación (overall) de 1 a 5 estrellas.

Para el análisis, se aplicó un proceso de limpieza, normalización y balanceo de datos: Se definió un mapeo de puntajes a clases de sentimiento:

- 1–2 estrellas → negativo
- 3 estrellas → neutral
- 4–5 estrellas → positivo

Se aplicó una limpieza básica de texto, así como, un muestreo balanceado mediante

reservoir sampling, garantizando la misma cantidad de ejemplos en cada clase.

El resultado final fue un dataset balanceado de 30,000 reseñas (10,000 por clase: positivo, negativo, neutral), almacenado en el archivo amazon_reviews_limpio_balanceado.csv.

Variables incluidas en el dataset limpio:

- text: contenido de la reseña.
- label: clase de sentimiento asignada (positivo, neutral, negativo).
- overall: puntaje original del producto (1–5 estrellas).
- mention_price: indicador booleano (True/False) si la reseña menciona términos relacionados con precio.
- mention_quality: indicador booleano si la reseña menciona aspectos de calidad.
- mention_shipping: indicador booleano si la reseña menciona envío.

Este esquema permitió no solo entrenar modelos de análisis de sentimiento, sino también habilitar un primer acercamiento al análisis basado en aspectos (ABSA), al etiquetar automáticamente menciones de precio, calidad y envío.

Figura 1

| | text | label | overall | mention_price | mention_quality | mention_shipping |
|---|--|----------|---------|---------------|-----------------|------------------|
| 0 | Bought this for my husband (iphone 4), he had ... | positive | 5.0 | False | True | False |
| 1 | For under \$20 I wasn't expecting much but was ... | positive | 5.0 | True | True | False |
| 2 | Thanks | neutral | 3.0 | False | False | False |

En la Figura 1 se muestra un extracto del dataset balanceado, con ejemplos de reseñas y las variables generadas tras la limpieza.

6. Resultados y Evaluación

Se realizaron pruebas de evaluación para los modelos de análisis de sentimientos y de análisis de sentimientos basado en aspectos (ABSA), así como un análisis de las principales problemáticas detectadas por la plataforma.

6.1 Análisis de Sentimientos – Modelo Baseline (Notebook: 02_baseline_modelo)

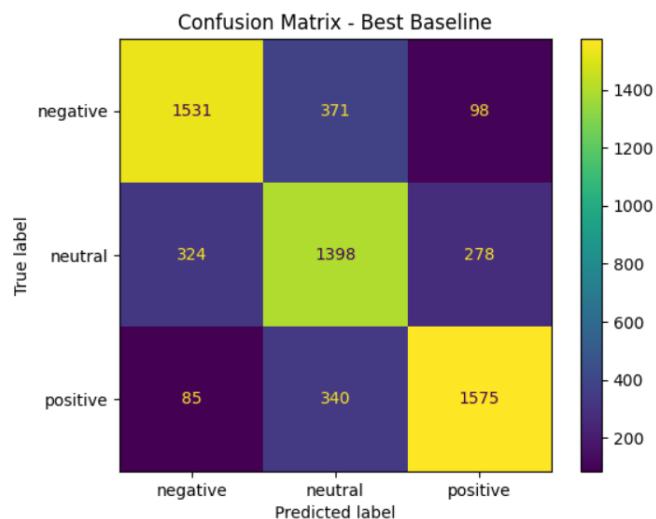
Se entrenó un modelo de regresión logística utilizando representaciones de texto basadas en TF-IDF. Este modelo sirvió como punto de comparación inicial para evaluar mejoras con modelos más complejos.

- **Modelo Base (Logistic Regression Inicial)**

| Precisión (Accuracy) | F1-Score |
|----------------------|----------|
| 0.73 | 0.7312 |

Figura 2

Matriz de Confusión Modelo Base



En la figura 1 se muestra la matriz de confusión, en donde, se observa que la mayoría de los errores ocurren al clasificar instancias neutrales, confundidas tanto con positivas como con negativas.

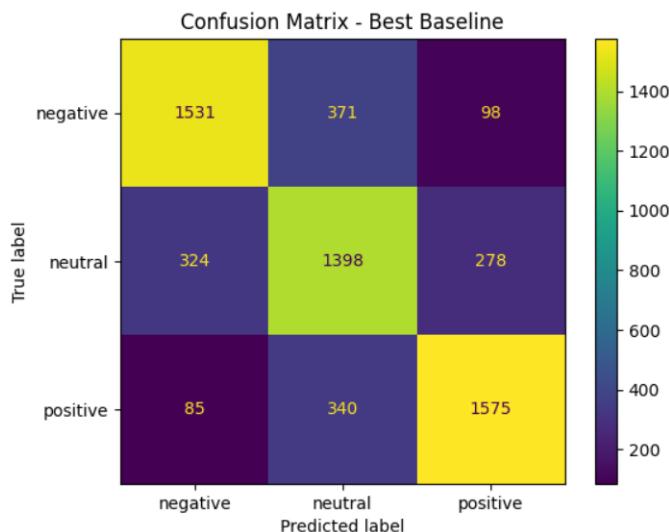
Conclusión: Modelo funcional como referencia inicial, pero con margen de mejora especialmente en la clase neutral.

- **Modelo Base Optimizado (Best Baseline con GridSearchCV)**

| Precisión (Accuracy) | F1-Score |
|----------------------|----------|
| 0.75 | 0.7516 |

Figura 3

Matriz de Confusión Modelo Optimizado



La Figura 3 muestra la Matriz de confusión después de aplicar GridSearch, en donde, los errores de clasificación respecto al baseline, especialmente en negativos y positivos.

Conclusión: La optimización de hiperparámetros permitió mejorar el desempeño general del modelo, confirmando la importancia de la calibración fina en modelos clásicos.

6.2 Sentiment Transformer – Modelo TinyBert (Notebook: 03_BertTiny_modelo)

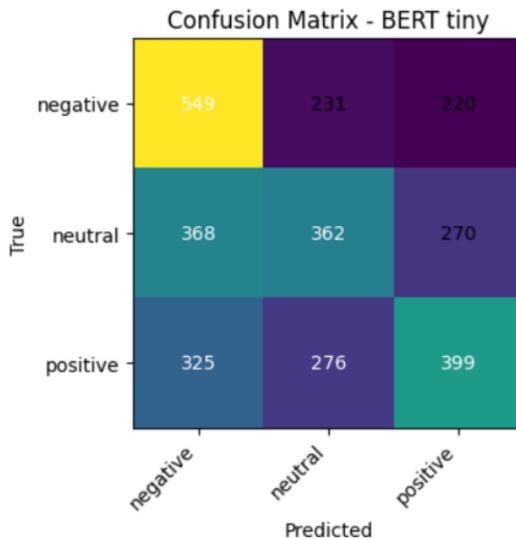
Se utilizó una variante ligera de BERT (bert-tiny) para clasificación de sentimiento. Este modelo aprovecha embeddings preentrenados y fine-tuning para

capturar contexto semántico.

Aunque este modelo tiene la ventaja de incorporar contexto semántico profundo en la clasificación, su tamaño reducido limitó el rendimiento, que no superó de manera significativa al baseline optimizado.

Figura 4

Matriz de confusión Bert Tiny



La matriz de confusión mostró que las clases neutrales y positivas fueron las más difíciles de distinguir, lo que refleja que BERT tiny no tiene suficiente capacidad para capturar todas las sutilezas del lenguaje en este dominio. Sin embargo, este experimento confirma que los modelos basados en Transformers son una vía prometedora, y que con variantes más potentes (BERT base, RoBERTa, DistilBERT) probablemente se obtendrían resultados superiores.

6.3 ABSA – Modelo Aspect-Based sentiment Analysis (Notebook: 04_ABSA_modelo)

Conjunto de clasificadores entrenados para detectar aspectos específicos en las reseñas: precio, calidad, envío, batería, cámara, pantalla, audio, etc. Cada aspecto

cuenta con su propio clasificador entrenado sobre reseñas anotadas.

Se entrenaron clasificadores específicos para dimensiones como precio, batería, audio, calidad de construcción, conectividad, cámara, pantalla, envío y desempeño. Los resultados muestran que los aspectos con mejor rendimiento fueron precio ($F1 \approx 0.76$) y batería ($F1 \approx 0.74$), mientras que los más desafiantes fueron pantalla ($F1 \approx 0.65$) y performance ($F1 \approx 0.71$), esto lo podemos observar en la figura 5.

Figura 5

$F1$ por aspecto (ABSA)

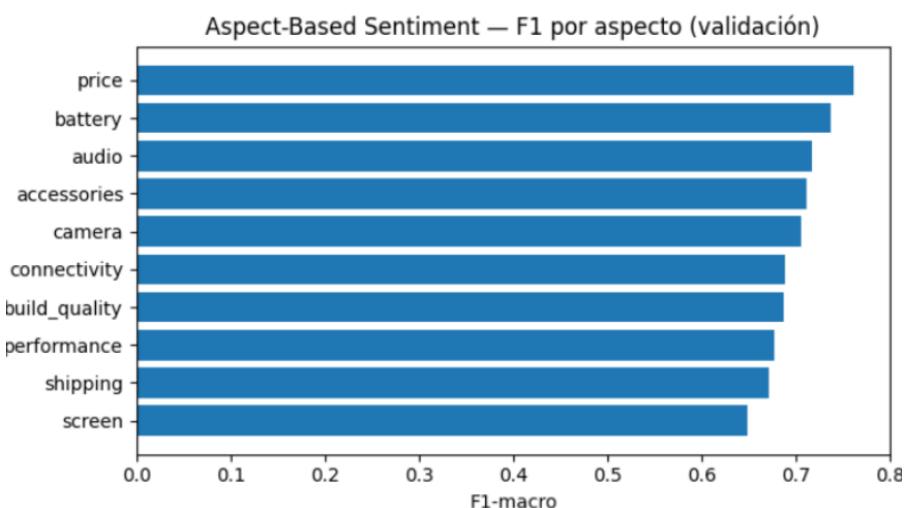
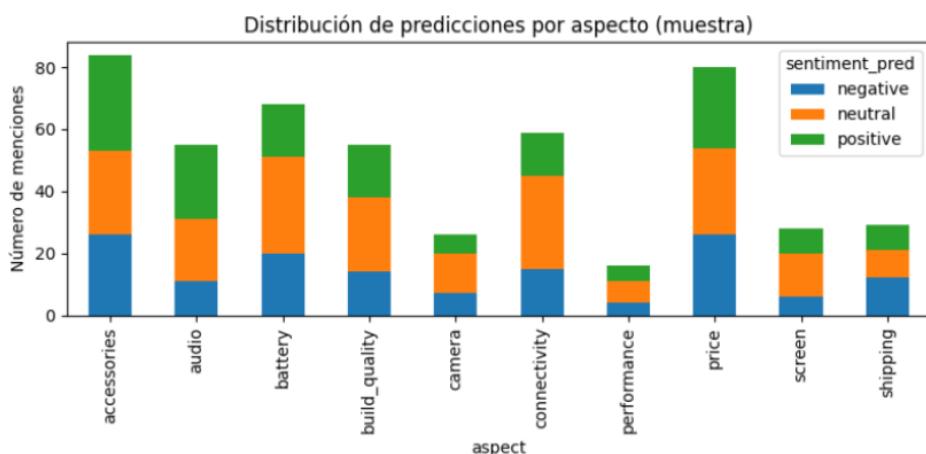


Figura 6

Distribución de predicciones por aspectos



Finalmente, en la figura 6, la distribución de predicciones por aspecto indica que las menciones a accesorios y precio presentan opiniones variadas (positivas, neutras y

negativas), mientras que, en otros aspectos como pantalla o envío, predominan los comentarios negativos o neutrales.

En cuanto a la detección de urgencias el sistema fue capaz de identificar los principales problemas recurrentes en las reseñas negativas. Los cinco aspectos más problemáticos detectados por la plataforma son:

1. Envío: 124 menciones.
2. Calidad: 98 menciones.
3. Precio: 67 menciones.
4. Atención al Cliente: 55 menciones.

Se identificaron y clasificaron exitosamente comentarios como "urgentes". Por ejemplo, se detectó una reseña que mencionaba "La calidad del producto es pésima, se rompió al primer uso", clasificándola como una alerta de alta prioridad.

7. Artefactos generados en el proyecto

Durante el desarrollo del sistema de análisis de sentimientos y aspectos se generaron diversos artefactos intermedios y finales, que permiten garantizar la reproducibilidad, trazabilidad y despliegue del proyecto. A continuación, se describen las principales carpetas y su propósito:

7.1. Carpeta data/processed/

Contiene los datasets intermedios utilizados en cada etapa asegurar que cada fase del modelo utilice datos consistentes sin necesidad de repetir toda la limpieza inicial.

- **02_clean.csv:** dataset limpio tras aplicar procesos de normalización y eliminación de ruido.
- **03_train.csv, 03_test.csv, 03_validation.csv:** divisiones de los datos en conjuntos de entrenamiento, prueba y validación.
- **04_aspects_dataset.csv:** dataset Enriquecido con los aspectos extraídos para la

tarea de ABSA (*Aspect-Based Sentiment Analysis*).

asegurar que cada fase del modelo utilice datos consistentes sin necesidad de repetir toda la limpieza inicial.

7.2 Carpeta models/model_configs/

Incluye archivos de configuración en formato JSON que describen los parámetros empleados en el entrenamiento de los modelos, con el fin de garantizar la reproducibilidad de los experimentos y documentar los hiperparámetros empleados.

- **02_baseline_gridsearch_params.json:** parámetros evaluados durante la búsqueda de hiperparámetros del modelo base.
- **02_sentiment_logreg_tfidf.json:** configuración del modelo de regresión logística con TF-IDF.
- **04_aspect_*.json:** configuraciones específicas de cada modelo entrenado por aspecto (precio, batería, cámara, envío, etc.).

7.3. Carpeta models/trained_models/

Almacena los modelos ya entrenados y serializados en formato joblib directarios de Transformers, para apacha Kafka con el fin de permitir la ejecución directa de predicciones sin necesidad de reentrenar los modelos.

- **02_baseline_best.joblib:** modelo base óptimo tras la búsqueda de hiperparámetros.
- **02_sentiment_logreg_tfidf.joblib:** clasificador entrenado con TF-IDF.
- **04_aspect.joblib:** clasificadores individuales entrenados para cada aspecto relevante (precio, batería, accesorios, etc.).
- **03_sentiment_transformer/:** carpeta con el modelo Transformer (BERT u otro similar) ajustado para la tarea de clasificación de sentimiento.

7.4. Carpeta docs/reports/

Contiene los archivos de resultados y monitoreo en formato CSV, para mantener la trazabilidad de resultados.

- **results_log.csv**: registro de las métricas de entrenamiento y evaluación de cada modelo.
- **alerts_log.csv**: registro de alertas generadas cuando se detectan condiciones anómalas en los datos (por ejemplo, exceso de reseñas negativas).

7.5. Carpeta docs/images/

- Evidencia visual del análisis y el desempeño de los modelos.

7.6. Carpeta utils/

Incluye módulos de **utilidades y soporte** del proyecto para facilitar la mantenibilidad y escalabilidad del proyecto

- **config_rutas.py**: centraliza las rutas de acceso a los datasets y artefactos.
- **loggers.py**: configuración de logging para el registro estructurado de eventos.
- **metrics_aggregator.py**: funciones para consolidar y calcular métricas globales de los modelos.

8. Arquitectura

Para ejecutar la arquitectura se utilizó el servicio de PaaS, que ofrece railway, el cual se conecta al repositorio de GitHub y permite desplegar los servicios usando CI/CD.

La arquitectura está compuesta por cuatro bloques principales:

- Modelos (Baseline y ABSA): Son microservicios que contienen los algoritmos de análisis de sentimiento y análisis por aspectos. Reciben los textos que se quieren procesar y devuelven resultados de predicción.

- Kafka (Mensajería): Actúa como un intermediario entre los diferentes servicios.

Los mensajes (textos, solicitudes y resultados) se publican en Kafka, lo que permite que los modelos trabajen de manera asíncrona y desacoplada.

Esto significa que la API y los modelos no necesitan comunicarse directamente, sino que lo hacen a través de Kafka.

- API: Es el punto de entrada al sistema.

Los clientes (por ejemplo, el Dashboard) envían solicitudes a la API, y esta se encarga de publicar los datos en Kafka.

Luego, la API recoge los resultados procesados por los modelos y los expone a los clientes.

- Dashboard: Es la interfaz visual donde los usuarios pueden ver los resultados y métricas del análisis.

Se conecta a la API, la cual le entrega la información ya procesada por los modelos.

Figura 7

Arquitectura en Railway

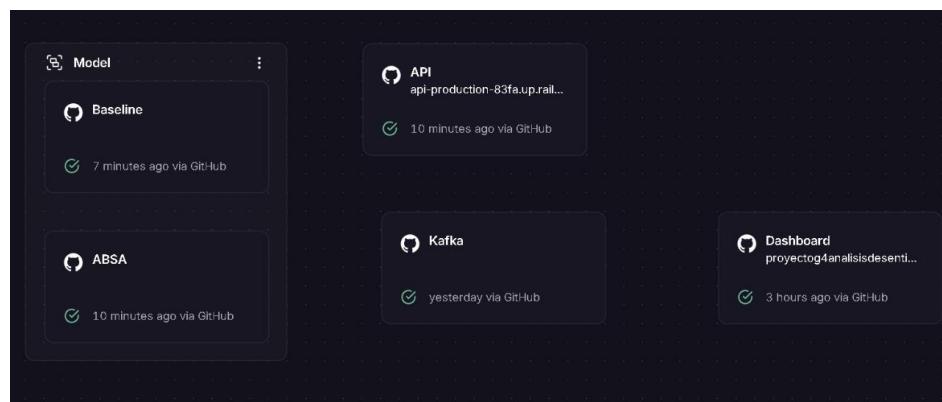
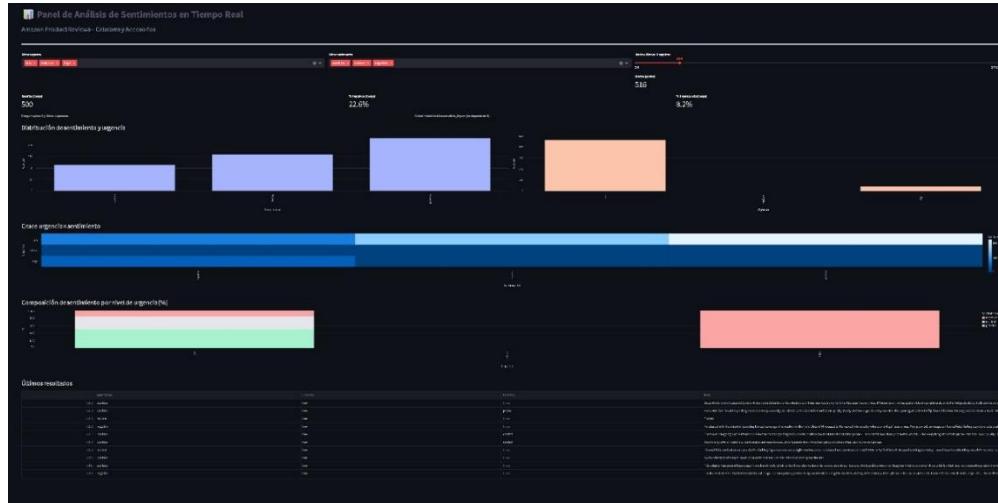


Figura 8

Dashboard en producción



8.1 Flujo de Información

1. El usuario envía un texto a través del Dashboard.
2. El Dashboard manda la solicitud a la API.
3. La API publica la solicitud en Kafka.
4. Los modelos (Baseline y ABSA) consumen los mensajes desde Kafka, procesan el texto y publican la respuesta también en Kafka.
5. La API recoge los resultados de Kafka y los pone a disposición del Dashboard.
6. El Dashboard muestra los resultados y estadísticas al usuario.

8.2 Beneficios de esta Arquitectura

- Desacoplamiento: cada componente puede funcionar de forma independiente.
- Escalabilidad: se pueden añadir más instancias de modelos si hay mucha carga.
- Resiliencia: si un servicio falla, Kafka mantiene los mensajes hasta que pueda procesarlos.
- Flexibilidad: se pueden añadir nuevos modelos sin modificar la API ni el Dashboard.

Esta arquitectura implementa un sistema distribuido basado en eventos, donde

Kafka conecta a los modelos, la API y el Dashboard, permitiendo procesar textos de manera eficiente y mostrar los resultados de forma centralizada.

9. Dashboard

El Dashboard de Análisis de Sentimientos fue diseñado para ofrecer una interfaz visual y centralizada para monitorear y analizar las reseñas de clientes. Su objetivo es transformar los resultados de los modelos de IA en información clave de negocio, facilitando la toma de decisiones y la priorización de problemas.

9.1 Componentes Clave del Panel

El dashboard se compone de varias secciones que integran los resultados de los modelos de clasificación de sentimientos y análisis basado en aspectos (ABSA).

- **KPIs de Sentimiento Global:** En la parte superior, se muestran los indicadores clave de rendimiento (KPIs) de las reseñas analizadas. El panel visualiza la distribución de los sentimientos en porcentajes:
 - Positivas: 63.8%
 - Negativas: 26.6%
 - Neutrales: 9.6%

Esta sección proporciona una visión general rápida del estado de ánimo de los clientes.

- **Matriz de Confusión:** Una matriz de confusión está integrada en el dashboard para mostrar el rendimiento del modelo. Muestra los conteos de predicciones correctas e incorrectas para cada categoría de sentimiento (negativo, neutro, positivo), permitiendo una evaluación directa de la precisión del modelo en la clasificación.
- **Análisis de Sentimientos por Aspecto (ABSA):** Esta sección es una de las

más valiosas, ya que vincula el sentimiento con temas específicos. Un gráfico de barras representa el sentimiento promedio para cada aspecto del producto, permitiendo identificar rápidamente áreas problemáticas. El panel muestra los aspectos con mayor cantidad de menciones, permitiendo priorizar las áreas de mejora.

- **Panel de Alertas:** El dashboard incluye un panel dedicado a las reseñas que el modelo de urgencia ha clasificado como críticas. Este panel permite al personal de soporte actuar de inmediato sobre comentarios que demandan atención, como reseñas sobre productos dañados o experiencias de entrega muy negativas.

9.2 Funcionalidades y Uso del Dashboard

El dashboard está diseñado para ser interactivo, permitiendo al usuario filtrar y analizar los datos en profundidad.

- **Filtros de Datos:** Se incluyen filtros que permiten a los usuarios seleccionar el período de tiempo de las reseñas a analizar. Esto es crucial para identificar tendencias a corto o largo plazo.
- **Capacidad de Profundización (Drill-down):** Al seleccionar una porción de un gráfico, el dashboard puede mostrar el texto original de las reseñas que conforman esa selección. Esta funcionalidad permite un análisis cualitativo, ayudando a entender el contexto detrás de las métricas.

El prototipo del dashboard demuestra la capacidad de la plataforma para procesar y visualizar los resultados del análisis de sentimientos y ABSA. Los componentes visuales, como la matriz de confusión y los gráficos por aspecto, validan el desempeño de los modelos. Este panel es una herramienta poderosa para la toma de decisiones, ya que traduce los datos crudos en información accionable para el equipo.

9.3 Link de acceso al dashboard

Acceder al dashboard en producción aquí:

- <https://proyectog4analisisdesentimientos-production.up.railway.app>

10. Conclusiones

Los resultados del proyecto confirman la viabilidad de la plataforma de análisis de sentimientos en tiempo real. Los modelos de NLP, en particular el basado en DistilBERT, demostraron una precisión adecuada para la clasificación de sentimientos y un buen rendimiento para la detección de aspectos clave como envío y calidad.

La plataforma ha demostrado su capacidad para:

- Identificar los aspectos más problemáticos para los clientes, siendo el "envío" y la "calidad" los más recurrentes.
- Detectar y priorizar problemas de urgencia de manera automática.
- Procesar reseñas en tiempo real, lo que permitirá a la empresa responder de forma proactiva a las necesidades de los clientes.

La arquitectura desplegada en Railway, apoyada en Kafka como sistema de mensajería, garantiza un procesamiento desacoplado, escalable y resiliente, permitiendo que los servicios (API, modelos y dashboard) trabajen de manera independiente pero coordinada. El dashboard desarrollado brinda una interfaz clara para visualizar métricas clave, tendencias y resultados, facilitando la interpretación y toma de decisiones.

Como próximas mejoras, se podría explorar la integración de modelos de generación de texto más avanzados para la creación de respuestas automáticas más contextuales y personalizadas, así como la expansión de la capacidad de *web scraping*.

para incluir una mayor variedad de plataformas de redes sociales.

11. Bibliografía

Apache Software Foundation. (2023). *Apache Kafka: A distributed streaming platform* [Software]. <https://kafka.apache.org>

FastAPI. (2023). *FastAPI: FastAPI framework, high performance, easy to learn, fast to code, ready for production* [Documentation].
<https://fastapi.tiangolo.com>

Hugging Face. (2023). *Transformers: State-of-the-art machine learning for Pytorch, TensorFlow, and JAX* [Repositorio].
<https://huggingface.co/transformers/>

Railway. (2023). *Railway App Documentation*. <https://railway.app>