



Escuela Politécnica Nacional
Facultad de Ingeniería de Sistemas

Ingeniería en Computación

Recuperación de la Información

Ing. Iván Carrera

Estudiantes: Danny Iñaguazo, Roberth Gancino

Fecha: 04/02/2026

Informe

Proyecto de 2do Bimestre Sistema de Recuperación Multimodal de Información

1. Objetivo:

Diseñar e implementar un sistema de recuperación de información multimodal para e-commerce que permita buscar productos usando texto o imágenes. El sistema debe combinar información textual y visual, aplicar técnicas de re-ranking para mejorar la relevancia de los resultados y generar respuestas explicativas mediante generación aumentada por recuperación (RAG), justificando cada recomendación con evidencia.

2. Descripción del Corpus y Preparación de Datos

Para el desarrollo de este proyecto se seleccionó un dataset especializado en el sector retail de moda, lo que permite evaluar la eficacia del sistema en un entorno con alta variabilidad visual y descriptiva.

2.1. Fuente de Datos

Dataset: [Fashion Images](#) de Kaggle.

Volumen: 2,907 registros de productos individuales.

Atributos clave: El corpus integra información estructurada y no estructurada, incluyendo:

Imágenes: Fotografías de los productos para extracción de características visuales.

Metadatos: Género, Categoría, Subcategoría, Tipo de Producto, Color, Uso y Título del Producto.

2.2. Codificación e Indexación Multimodal

El proceso de preparación de datos se centró en la creación de una representación vectorial única que unificara la semántica visual y textual.

Modelo de Embeddings: Se utilizó CLIP (openai/clip-vit-base-patch32). Este modelo es fundamental para la búsqueda multimodal, ya que ha sido pre-entrenado para alinear imágenes y sus descripciones en un espacio vectorial compartido de 512 dimensiones.

Estrategia de Vectorización:

Texto: Se construyó una descripción enriquecida concatenando todos los atributos del producto (ej: "Men Apparel Shirts Casual Blue Cotton Shirt").

Imagen: Se procesaron las URLs de las imágenes, convirtiéndolas en tensores mediante CLIPProcessor para obtener sus características visuales.

Fusión Multimodal: Se implementó un embedding promedio $\text{Vector_final} = (\text{V_texto} + \text{V_imagen})/2$, permitiendo que el índice vectorial responda con igual precisión a consultas de imagen o de texto.

Almacenamiento y Recuperación:

Índice Vectorial: Se utilizó la librería FAISS (Facebook AI Similarity Search). Se optó por un índice de tipo IndexFlatIP (Inner Product).

Normalización: Los vectores fueron normalizados utilizando `faiss.normalize_L2`, lo que permite que la búsqueda por producto interno funcione como una similitud de coseno, midiendo el ángulo (y por tanto la relevancia semántica) entre la consulta y los productos indexados.

2.3. Persistencia de Datos

El pipeline genera dos archivos críticos para la fase de búsqueda en tiempo real:

productos.faiss: El índice vectorial optimizado para búsquedas de alta velocidad.

metadata.json: Un mapeo de los IDs de los productos con sus atributos legibles, necesario para alimentar el contexto del modelo RAG y la interfaz de usuario.

3. Arquitectura del Pipeline y Lógica del Sistema

El sistema implementa un pipeline de Recuperación de Información Multimodal que consta de cuatro etapas principales: Entrada y Procesamiento, Recuperación Inicial, Re-ranking y Generación Aumentada (RAG).

3.1. Pipeline de Búsqueda Multimodal (Retrieval)

El sistema permite consultas híbridas, procesando tanto lenguaje natural como imágenes a través de la clase MotorBusqueda:

Entrada de Consulta:

Texto: Las consultas se codifican directamente usando el modelo clip-ViT-B-32.

Imagen: Las imágenes subidas se convierten al espacio de color RGB para asegurar compatibilidad y se procesan a través del mismo encoder para obtener un vector en el espacio compartido de CLIP.

Recuperación Vectorial (FAISS): Se realiza una búsqueda de similitud de producto interno (IP) en el índice productos.faiss, recuperando un conjunto inicial de top-20 candidatos (top_k). La normalización L2 garantiza que la búsqueda se comporte como una similitud de coseno.

3.2. Mecanismo de Re-ranking

Para mejorar la precisión de los resultados textuales, se implementó una etapa de reordenamiento explícito:

Modelo: Se utiliza un Cross-Encoder (ms-marco-MiniLM-L-6-v2), que analiza pares de [Consulta, Descripción del Producto].

Lógica: A diferencia del retrieval inicial (bi-encoder) que es rápido, pero menos preciso, el Cross-Encoder permite capturar relaciones semánticas más profundas. Los resultados se ordenan según su puntaje de relevancia (score), seleccionando finalmente el top-3 para ser presentados al usuario y al modelo de lenguaje.

3.3. Generación Aumentada por Recuperación (RAG)

Una vez identificados los productos más relevantes, el sistema construye un contexto dinámico para el modelo generativo Gemini (gemini-3-flash-preview):

Construcción del Contexto: Se extraen los metadatos (nombre, categoría, color, uso) de los productos del top-ranking y se formatean como una lista estructurada de productos.

Prompt Engineering: Se utiliza un prompt de sistema que define al modelo como un "Fashion Stylist". Este prompt instruye a la IA para:

- Recomendar productos basándose estrictamente en el contexto recuperado.
- Justificar la recomendación mediante atributos específicos.
- Mantener la fidelidad a la información (evitar alucinaciones).

3.4. Gestión de Memoria y Contexto de Sesión

La aplicación utiliza la memoria de estado de Streamlit (st.session_state) para gestionar la continuidad de la conversación:

Historial de Chat: Se almacenan los últimos mensajes y los productos recuperados en cada turno.

Refinamiento: Al generar una respuesta, se inyectan los últimos 3 mensajes del historial en el prompt de Gemini. Esto permite que el sistema comprenda consultas de

seguimiento (ej. "ahora muéstrame el primero en azul") manteniendo la referencia de los productos previamente discutidos.

3.5. Interfaz de Usuario

Se diseñó una interfaz gráfica conversacional utilizando Streamlit, organizada de la siguiente manera:

Sidebar: Dedicado a la búsqueda visual (carga de archivos de imagen).

Chat Principal: Área de interacción donde se despliegan los mensajes del usuario, las respuestas del asistente y tarjetas visuales de los productos con sus metadatos básicos.

4. Ejemplos de consultas y resultados obtenidos

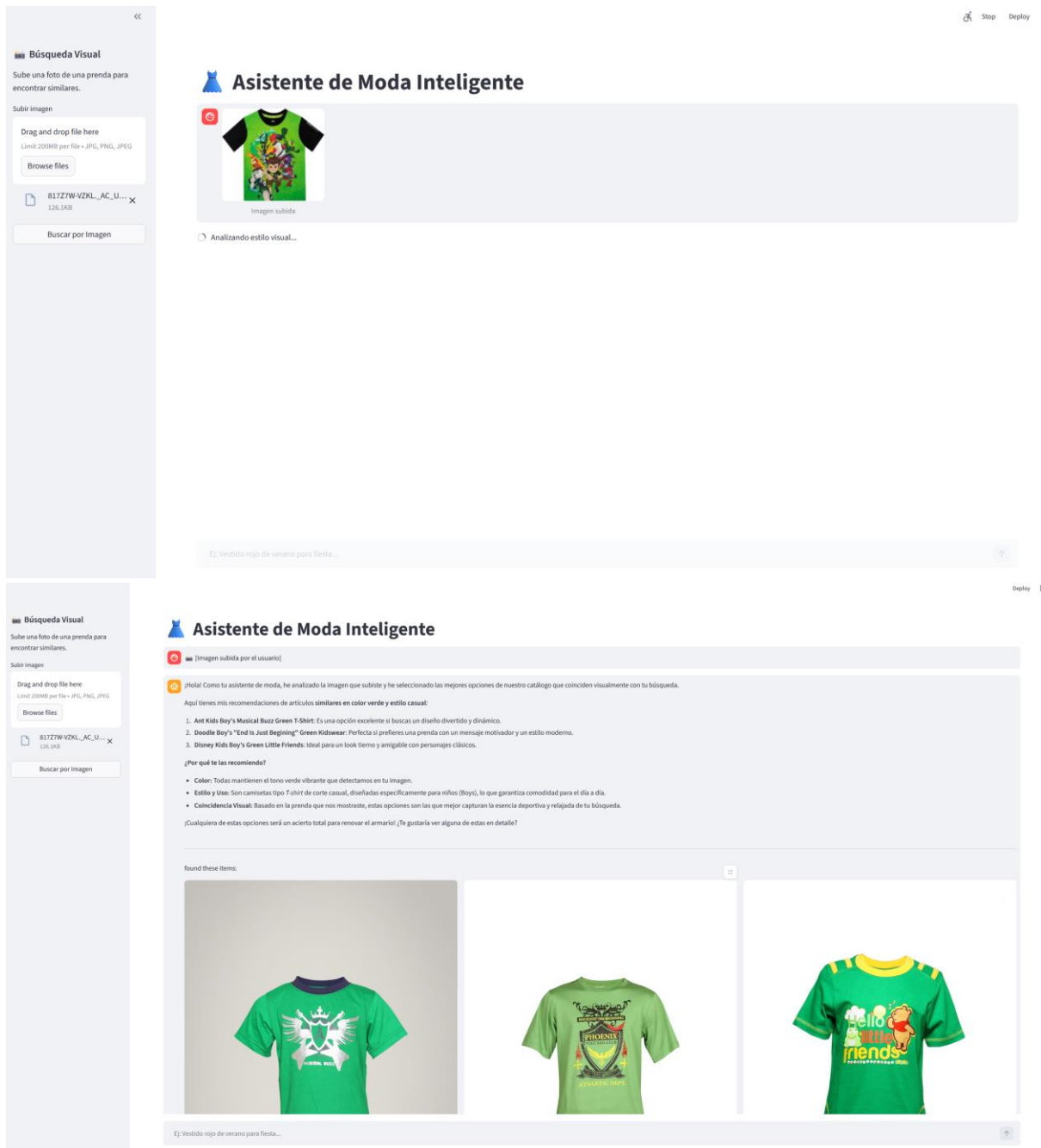
Al ejecutar el sistema se realizaron dos pruebas correspondientes a la búsqueda de texto para obtener una imagen y viceversa.

Ejemplo de búsqueda de texto a imagen



Se puede apreciar que los resultados corresponden a la búsqueda que se realizó en el chat del sistema.

Ejemplo de búsqueda de imagen a texto con resultados visuales integrados



Como se puede ver en las anteriores capturas de pantalla se cargó una imagen correspondiente a una camiseta infantil para realizar una búsqueda de una camiseta parecida, obteniendo resultados muy similares a lo que se buscaba.

4. Análisis Cualitativo del Sistema

En esta sección se evalúa el comportamiento del pipeline en escenarios reales, analizando el impacto del Re-ranking y la fidelidad del RAG frente a las limitaciones del modelo de embeddings y el corpus.

4.1. Evaluación del Impacto del Re-ranking y Retrieval

A partir de las pruebas realizadas, se identificaron los siguientes comportamientos en la fase de recuperación:

Dominancia de Atributos Visuales sobre Semánticos: Se observó que el sistema prioriza el color sobre el tipo de prenda (ej. recuperar una camiseta roja cuando se pidió un zapato rojo). Esto indica que el espacio vectorial de CLIP está fuertemente influenciado por la intensidad del color, lo que provoca que, ante la ausencia de una coincidencia exacta de categoría, el "vecino más cercano" sea determinado por la cromatografía.

Limitación en Adjetivos Abstractos: Consultas con términos como "elegante" o "casual" no mostraron una mejora significativa. Esto se debe a que estos conceptos son subjetivos y no están explícitamente etiquetados en los metadatos del corpus, dificultando que el Cross-Encoder asocie el término con una característica técnica de la prenda.

Desafíos en la Búsqueda Visual: En la búsqueda por imagen, el sistema captura "cualidades" (texturas, tonos) pero no siempre la estructura del objeto. Esto es una característica intrínseca de los modelos zero-shot como CLIP, que ven la imagen como un todo y no siempre segmentan el objeto principal de manera aislada.

4.2. Calidad de las Respuestas Generadas (RAG)

El componente de Gemini demostró ser una capa de seguridad y claridad esencial:

Justificación de Fallos: Cuando el retrieval falla (como en el caso de no encontrar el color solicitado), el modelo generativo actúa como un filtro de honestidad, explicando al usuario que presenta "las mejores coincidencias disponibles", mitigando la frustración.

Grounded Generation: El sistema se mantuvo fiel al contexto, evitando inventar productos que no estaban en el Top-K recuperado, cumpliendo con la restricción de groundedness.

5. Recomendaciones para Mitigar Errores y Mejorar la Precisión

Para optimizar el rendimiento y resolver los puntos críticos observados, se proponen las siguientes mejoras técnicas:

A. Refinamiento de la Búsqueda (Weighted Retrieval)

Para evitar que el color domine sobre el tipo de producto, se recomienda implementar una Búsqueda Híbrida Ponderada:

- Asignar un peso mayor al campo ProductType y Category durante la indexación.

- Utilizar filtros de metadatos (Hard Filters) antes de la búsqueda vectorial. Si el usuario dice "zapatos", el sistema debe filtrar el índice para mostrar solo la categoría de calzado antes de calcular la similitud por color.

B. Mejora del Re-ranking con Modelos de Dominio

El Cross-Encoder utilizado (ms-marco) está entrenado en pasajes de texto general. Utilizar un re-ranker entrenado específicamente en e-commerce o realizar un fine-tuning con pares de [Consulta -> Atributos de Producto] para que términos como "elegante" se asocien a categorías específicas (ej. Camisas de vestir, Blazers).

C. Implementación de Memoria de Atributos (Slot Filling)

Para resolver el problema de los cambios de color, el sistema debe usar una técnica de extracción de entidades:

Si el usuario dice "ahora en rojo", el sistema debe detectar que la entidad "Categoría" (ej. Zapatos) se mantiene constante de la sesión anterior y solo actualizar el atributo "Color" en la nueva consulta al índice.

D. Pre-procesamiento de Imágenes

Para mejorar la búsqueda visual:

Implementar un modelo de Detección de Objetos (como YOLOv8) antes de pasar la imagen a CLIP. Esto permitiría recortar el fondo y centrarse exclusivamente en la prenda, eliminando ruido visual que confunde al encoder.

6. Conclusiones

La implementación de una arquitectura multimodal con CLIP y FAISS demuestra que es posible unificar catálogos visuales y textuales en un solo espacio semántico, facilitando búsquedas altamente intuitivas. No obstante, las pruebas revelaron que el sistema tiende a priorizar el color sobre la categoría del producto, lo que sugiere que, para entornos de e-commerce reales, es necesario combinar los embeddings con filtros de metadatos estructurados para evitar que la similitud visual domine sobre la intención de compra.

El uso de un Cross-Encoder para el re-ranking y Gemini para la generación RAG probó ser una combinación ganadora para la experiencia del usuario. El re-ranking corrige las imprecisiones del motor vectorial inicial, mientras que el modelo de lenguaje actúa como un asesor experto que justifica cada recomendación basándose en el contexto recuperado, eliminando la sensación de "caja negra" y aumentando la confianza del cliente en los resultados mostrados.

Finalmente, el manejo de la memoria de sesión resalta como el componente crítico para la personalización. Aunque el sistema logra mantener el hilo de la conversación,

el desafío reside en la persistencia de atributos específicos durante el refinamiento. Para optimizar el pipeline, se recomienda integrar una lógica de extracción de entidades que permita al usuario modificar características aisladas, como el color, sin perder la referencia del producto ancla identificado al inicio de la sesión.