# Initial Proposal:
# Text–Image Retrieval for Fashion Recommendation

Rushil Randhar

April 17, 2025

## 1. Introduction & Problem Statement

Modern e-commerce platforms predominantly use collaborative filtering (CF)—leveraging user–item co-purchase or co-view patterns—to recommend products. While CF excels at surfacing popular or "people-like-you" items, it struggles with cold-start scenarios (new users or new items), and cannot capture nuanced style cues or "vibes" beyond transactional data. Content-based approaches add metadata (e.g. categories, attributes) but require manual tagging and still miss the implicit, aesthetic qualities that drive fashion choices.

Recent advances in transformer-based embedding models (e.g. CLIP, Sentence-BERT, and the newer TULIP encoder) enable direct mapping of free-form text into a rich semantic space aligned with image or caption embeddings. This "vibe search" paradigm lets users describe what they want—"vintage-inspired floral midi dress" or "edgy streetwear bomber jacket"—and retrieves items whose embeddings lie closest to that description.

**Project Goal:** Build and evaluate a text to image retrieval benchmark on a standard fashion dataset, comparing embedding models (CLIP, SBERT, TULIP), and demonstrate how transformer embeddings unlock "vibe"-based recommendations that go beyond traditional CF.

## 2. Data Sources

- **Fashion200k**: 200 000 fashion product images with human-written captions; standard train/val/test splits with ground-truth for recall@$K$.
- **(Possible Alternative) FashionIQ**: 30 000 triplets (query text, positive image, negative image) for a 3-way accuracy evaluation.
- Data access through the Hugging Face `datasets` library or direct download.

## 3. Embedding Model Options

First, I compared three off-the-shelf text-embedding approaches:

1. **CLIP (ViT–B/32)** – joint image/text space; strong zero-shot performance.
2. **Sentence-BERT (all-miniLM)** – lightweight, optimized for sentence similarity.
3. **TULIP** – recent text encoder with state-of-the-art semantic alignment on fashion captions.

Based on the available research I decided to select TULIP for its superior recall@$K$ on a held-out validation split.

## 4. Pipeline

1. **Precompute embeddings**:
   - Encode all test-split captions and images.
2. **Indexing**:
   - Build a FAISS index over image embeddings or caption embeddings.
3. **Retrieval**:
   - For each text query, compute its embedding and retrieve top–$K$ nearest neighbors from FAISS.
4. **Service Wrap**:
   - Package retrieval in a single Modal endpoint or Flask endpoint.

## 5. Evaluation

- **Recall@**$K$ ($K = 1, 5, 10$) on Fashion200k test split.
- **Alternative check for Triplet accuracy** on FashionIQ: fraction of times the positive image ranks above the negative.
- **Inter-annotator check** (Alternative): validate a small set of retrievals via pairwise judgments.

## 6. Deliverables

- **Codebase**: scripts/notebooks for data loading, embedding, FAISS index, retrieval, and evaluation.
- **Results notebook**: tables and plots of recall@$K$ and triplet accuracy.
- **Presentation slides**: including
  - Pipeline diagram
  - Recall@$K$ curve
  - t-SNE/UMAP of embeddings
  - Example query–retrieval grids