Semantic Product Search & Recommendation

Capstone Project – submitted by Jayesh Reddy

End-to-End Machine Learning Lifecycle Capstone Project – Formal Report

Introduction

This report presents an end-to-end application of the machine learning lifecycle to a real-world e-commerce problem, focusing on improving product discovery through natural-language search and recommendation. Using a publicly available Walmart product dataset, the project moves from problem framing and data understanding through preprocessing, feature engineering, model implementation, evaluation, and critical reflection on ethical and bias considerations. The emphasis throughout is on methodological rigour, appropriate model selection, and practical relevance, with decisions grounded in the nature of the problem rather than model complexity for its own sake.

Step 1 — Problem Understanding & Framing

Business Problem

Most e-commerce platforms support product discovery through two dominant mechanisms:

1. Keyword-based search, which assumes that customers know the correct product names or terminology; and
2. Generic recommender widgets (e.g. “Customers also bought…”), which typically depend on prior transactions or interactions with specific items.

These approaches work reasonably well when users have a clear, well-defined idea of what they are looking for. However, in reality customers are more likely to think in natural language, describing needs or feature, constraints, or use-cases rather than exact products. Examples include:

- “I’m looking for a small, quiet blender that’s easy to clean.”
- “That USB thing that lets me plug an extra screen into my laptop.”
- “A water bottle that keeps drinks cold all day but isn’t too bulky.”

When systems are unable to interpret such descriptions, users may struggle to find suitable products, leading to search abandonment, frustration, and lost sales. From a systems perspective, this creates a balancing loop: poor discovery reduces engagement, which in turn limits the behavioural data needed to improve recommendations.

The underlying business challenge is therefore one of ‘intent translation’: enabling users to express what they want in their own words and still be guided to relevant products in the catalogue.

Data Science Problem

To address this challenge, the data science challenge is to design a system that can map natural-language shopping queries to relevant products, even when the query does not match product titles or metadata exactly.

Formally, this can be framed as a 'retrieval and ranking problem':

Given a free-text query q and a catalogue of products
$\{p_1, p_2, \dots, p_N\}$,
the objective is to:

1. Retrieve a subset of products that are likely to match the user's intent; and
2. Rank these products by semantic relevance and simple business considerations (e.g. availability or price).

This project will explicitly compare two approaches:

- A Keyword-Based Baseline, using TF-IDF representations of product titles and descriptions combined with cosine similarity. This reflects conventional, term-based information retrieval techniques and provides a transparent benchmark.
- An Embedding-Based Semantic Search Model, where both product descriptions and user queries are represented using pretrained transformer-based text embeddings. By capturing contextual meaning rather than exact word overlap, this approach builds directly on earlier learning from modules which included transformer architectures and attention mechanisms.

Future development of this project could include an extension to apply a lightweight re-ranking layer incorporating basic business constraints or diversity considerations, which would link to earlier content in this program on clustering, explainability, and bias.

Task Type

- **Primary task:** Recommendation / information retrieval
- **Learning paradigm:**
    - Representation learning (via pretrained embeddings)
    - Similarity-based retrieval (nearest neighbours)
- **Modelling progression:**
    - Baseline: TF-IDF + cosine similarity
    - Improved model: embedding-based semantic similarity
    - (Optional) re-ranking using structured features

This progression mirrors the baseline-to-improvement comparisons used in earlier assignments (e.g. simpler statistical or linear models versus more expressive approaches), reinforcing a consistent analytical approach.

Success Metrics (Machine Learning)

As the output of the system is a ranked list of products, evaluation focuses on ranking metrics rather than classification or regression metrics used previously:

- **Recall@K**
  Measures the proportion of relevant products that appear in the top K of results.
- **Precision@K**
  Measures the relevance quality of the top K of recommendations.
- **nDCG@K (Normalised Discounted Cumulative Gain)**
  Accounts for ranking order by assigning higher weight to relevant products that appear earlier in the list.

Ideally, relevance will be defined using:

- Historical user interaction data (e.g. clicks or purchases), or
- A manually constructed evaluation set of query–product pairs derived from product metadata.

This choice of metrics reflects a principle emphasised throughout earlier sections of the coursework : 'evaluation methods must align with the nature of the problem being solved.'

Business KPIs

If implemented in a real-world e-commerce context, the system would aim to improve:

- Search success rate / reduced abandonment
  - as users are better able to express intent without relying on rigid filters or keywords.
- Conversion rate
  - by helping users reach suitable products more efficiently.
- Average revenue per session
  - as improved discovery can surface relevant alternatives and complementary items.
- Customer satisfaction
  - driven by a more intuitive, human-centred search experience.
- Catalogue utilisation
  - particularly for long-tail products that are often under-exposed in popularity-driven systems.

These outcomes are closely tied to ethical and system-level considerations, such as avoiding reinforcing popularity bias and ensuring fair exposure across the product catalogue.

Step 2 — Data Collection & Understanding

Dataset Source and Context

The dataset used in this project was sourced from Kaggle, in line with the assignment requirement to utilise a publicly available dataset. Specifically, the dataset is the *Walmart Product Data (2019)* published by *PromptCloud* and available at:

https://www.kaggle.com/datasets/promptcloud/walmart-product-data-2019

The dataset contains approximately 30,000 product listings from Walmart.com, covering a defined time period of December 2019. This dataset represents a snapshot of a large, real-world e-commerce platform and includes both structured product metadata and rich natural-language product descriptions.

This dataset was originally compiled via web scraping for analytical and research purposes. As such, it prioritises product-level catalogue information over user-level behavioural data (e.g. clicks, views, or purchases). This characteristic makes the dataset a good candidate for exploring product discovery and semantic retrieval, rather than personalised recommendation based on historical user interactions.

Unit of Analysis

Each row in the dataset represents one product listing (typically referred to as a Stock Keeping Unit or SKU in retail) available on Walmart.com.

This clear and consistent unit of analysis supports the objectives defined in Step 1, namely:

- semantic matching between free-text user queries and product descriptions,
- retrieval and ranking of relevant items within a catalogue,
- analysis of discovery and exposure across products.

The absence of session-level or user-interaction data means that the project focuses on content-based and semantic relevance, rather than collaborative filtering or user-specific preference learning – which reflects the original intent when this project was conceptualised.

Dataset Structure Overview

The dataset consists of approximately 30 000 rows and multiple columns spanning textual, categorical, and numerical data types. The features can be grouped into the following broad categories:

- Product identification
- Natural-language product information
- Categorical metadata
- Pricing and rating information

- Contextual metadata

This structure allows for both conventional information-retrieval techniques and embedding-based semantic approaches to be applied within a consistent framework.

Key Features and Data Types

The table below summarises the most relevant fields used in this project.

Data Dictionary (Selected Fields)

Feature Name	Data Type	Description
product_id	Categorical / ID	Unique identifier for each product listing
title	Text	Short, keyword-oriented product title
description	Text	Rich free-text product description, including features and use-cases
category	Categorical	Product category (often hierarchical)
brand	Categorical	Brand or manufacturer
price	Numerical	Listed product price
rating	Numerical	Average customer rating (where available)
review_count	Numerical	Number of customer reviews (proxy for popularity)
url	Text	Product page URL (used for context only)

Not all fields are populated for every product. The handling of missing values and inconsistencies is addressed in Step 3.

Initial Data Observations

Initial inspection of the dataset reveals several characteristics relevant to the modelling task:

- Rich natural-language descriptions
    - Product descriptions are written in consumer-facing prose and frequently include information about functionality, materials, use-cases, and key features. This makes them well suited to both TF-IDF-based keyword matching and embedding-based semantic representations.
- Variation in description quality
    - The length and level of detail in product descriptions vary considerably. Some listings contain detailed narratives, while others rely more heavily on titles and structured attributes. This variation is relevant for both retrieval performance and later bias analysis.

- Broad category coverage
    - The dataset spans a wide range of product categories, reflecting the diversity of a large general retailer. This supports analysis of discovery across different product types and reduces the risk of category-specific overfitting.
- Availability of structured metadata
    - Fields such as price, brand, and ratings provide opportunities for exploratory data analysis, interpretability, and the optional incorporation of lightweight business constraints during ranking.

Relevance to the Data Science Task

The dataset aligns closely with the semantic retrieval and ranking problem outlined in Step 1:

- Titles and descriptions support a clear comparison between keyword-based search (TF-IDF) and semantic search using pretrained text embeddings.
- Categorical attributes (e.g. category and brand) provide contextual grounding and enable analysis of exposure and diversity.
- Pricing and rating information allow for limited incorporation of business logic and explainability without introducing unnecessary model complexity.

Given that explicit user-interaction data is absent, relevance is inferred through:

- semantic similarity between queries and product descriptions,
- category consistency,
- and manually constructed or weakly supervised evaluation sets.

This limitation is acknowledged explicitly but still considered acceptable for the purposes of this project.

Limitations and Assumptions

The following limitations are recognised at this stage:

- No user-level behavioural data
    - The project evaluates semantic relevance rather than personalised preference learning.
- Marketing and boilerplate language
    - Some product descriptions contain promotional or repetitive text, which may influence similarity scores.
- Static snapshot
    - The dataset represents a fixed point in time and does not capture time-based dynamics such as seasonality or evolving user preferences.

These limitations are documented transparently and will be considered in subsequent evaluation and ethical analysis.

Justification for Dataset Selection

This dataset was selected because it:

- Aligns closely with the business problem outlined in Step 1
- Provides rich natural-language content suitable for semantic modelling
- Has a clear and well-structured tabular format
- Minimises preprocessing and coding friction
- Enables a transparent baseline-to-improvement comparison between conventional and embedding-based approaches
- Is publicly available via Kaggle, ensuring transparency and reproducibility

Overall, the dataset provides a strong and realistic foundation for demonstrating the end-to-end application of the machine learning lifecycle in an e-commerce context.

Step 3: Data Preparation, Exploration, and Feature Construction

Step 3 focuses on transforming the raw product catalogue into representations suitable for retrieval and recommendation. The work in this step is deliberately structured in two parts viz. **Step 3A** which addresses data preprocessing and applied exploratory analysis with the aim of establishing data quality, understanding the structure and limitations of the catalogue, and grounding subsequent modelling decisions in empirical observations. Building on this foundation, **Step 3B** shifts attention to feature construction, where both a conventional keyword-based representation and a semantic embedding-based representation are developed. This progression from understanding the data to engineering meaningful features reflects the baseline-to-improvement logic established earlier in the project and sets the scene for model implementation and evaluation to come in Step 4.

Step 3A: Data Preprocessing and Applied Exploratory Data Analysis

This step focused on preparing the Walmart product catalogue for the downstream task of semantic retrieval and, of equal importance, developing a grounded understanding of the data itself. In keeping with the core of the proposed solution relying on interpreting natural-language descriptions and matching these to user queries, specific attention was paid to the quality, completeness, and structure of the textual fields, with supporting consideration given to pricing and categorical metadata.

Data Cleaning and Quality Assessment

When loaded, the dataset comprised approximately 30,000 product records across 14 original variables, including product titles and descriptions, pricing information, categorical attributes, and a range of product identifiers.

An initial review of data quality revealed that not all fields were equally informative. Two variables (package_size and postal_code) were entirely empty across the dataset and were

therefore removed, as they offered no analytical value. Other fields, such as item_number, showed high levels of missing data (or 'missingness') -in the region of 70%. This pattern is typical of large, scraped e-commerce catalogues and did not impact the objectives of this project. These variables were retained for maintaining integrity and completeness but were not used in subsequent modelling steps.

In contrast, the fields central to the retrieval task exhibited very low levels of missing data. Fewer than 0.2% of product descriptions were missing, and missingness in brand and category fields was similarly limited. Importantly, no products were found to have both a missing title and a missing description. As a result, the full catalogue could be retained and no records excluded due to absent textual content.

To support consistency and reproducibility, column names were standardised to a uniform naming convention, and light text cleaning was applied to remove extraneous whitespace while preserving the semantic content of the descriptions.

Duplicate Handling

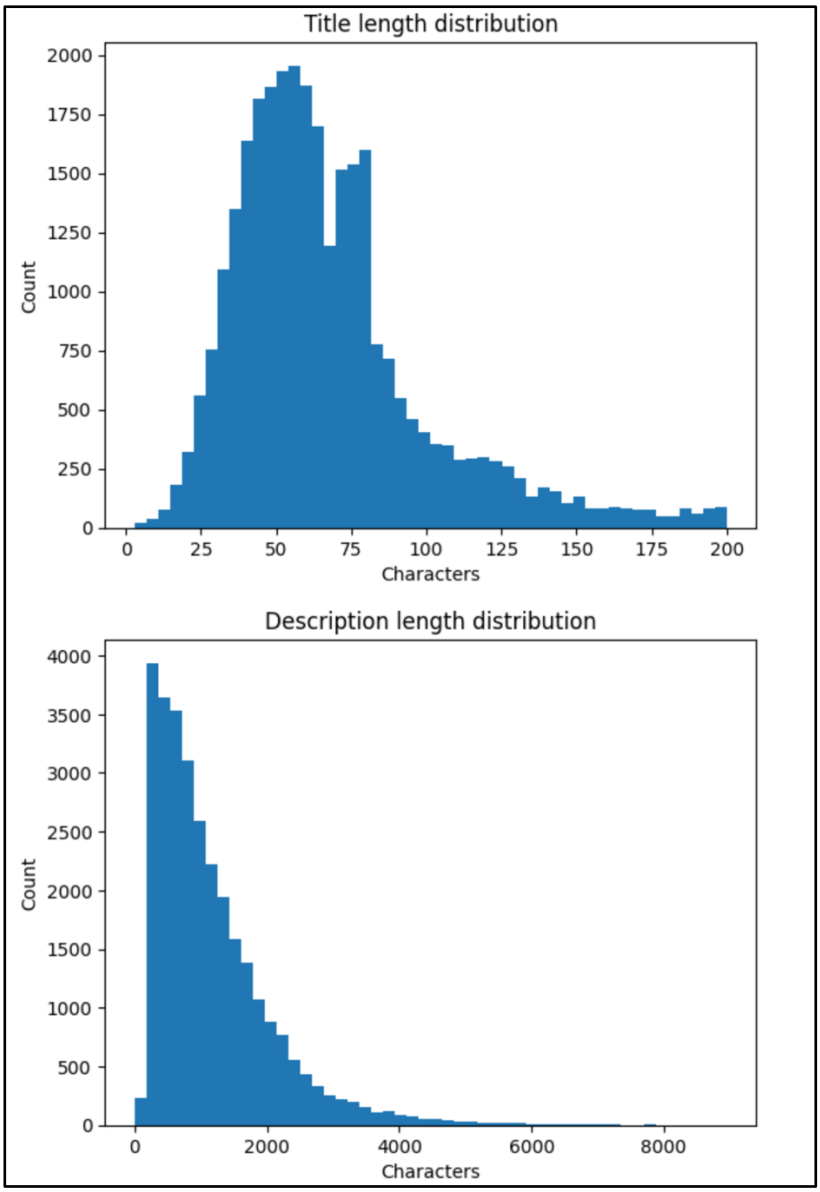
Although no full-row duplicates were identified, a small number of products shared identical combinations of cleaned titles and descriptions. In total, 167 such duplicates were removed, leaving a final working dataset of 29,833 unique products.

This approach to duplicate handling was deliberate. From the perspective of both TF-IDF and embedding-based representations, products with identical titles and descriptions would otherwise occupy the same position in the retrieval space. Removing these duplicates ensured that each product was represented once thus avoiding redundant vectors and possible distortions of similarity-based retrieval results.

Textual Feature Characteristics

With the emphasis on natural-language retrieval, the length and distribution of textual fields were examined in quite some detail.

Product descriptions displayed a strongly right-skewed distribution. While the median description length was approximately 927 characters, a subset of products included much longer descriptions, in some cases exceeding 8,000 characters. This points to substantial variation in the richness of descriptive text across the catalogue. In contras, product titles were substantially shorter and more tightly distributed, with a median length of 61 characters which reflects their role as concise identifiers rather than detailed explanations.



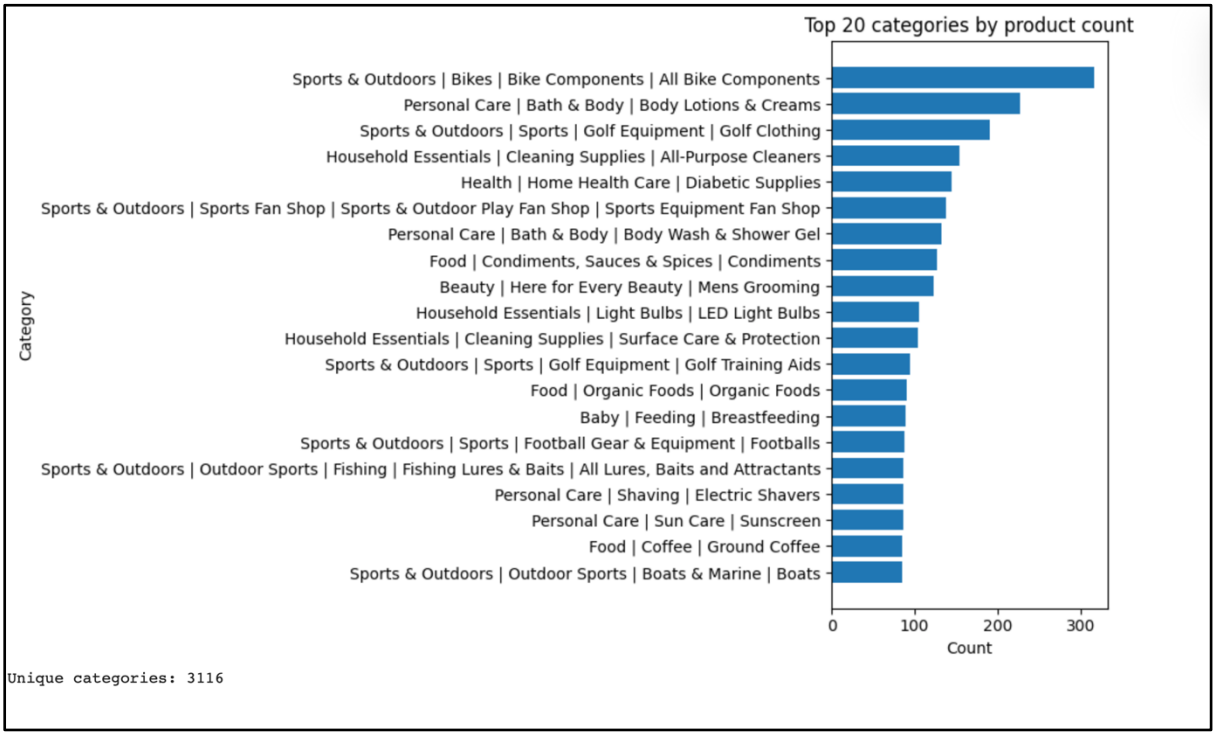
Taken together, these patterns suggest that the primary semantic signal resides in the product descriptions, with titles providing complementary but limited context. This observation directly informed the feature engineering strategy adopted in the subsequent step, where descriptions were prioritised in the construction of semantic representations.

The variation in description length also raises an important consideration for later stages of the project. Products with more detailed descriptions may be inherently easier for similarity-based models to retrieve, introducing the possibility of uneven exposure across the catalogue. This issue is revisited explicitly in the bias and fairness analysis later in Step 5.

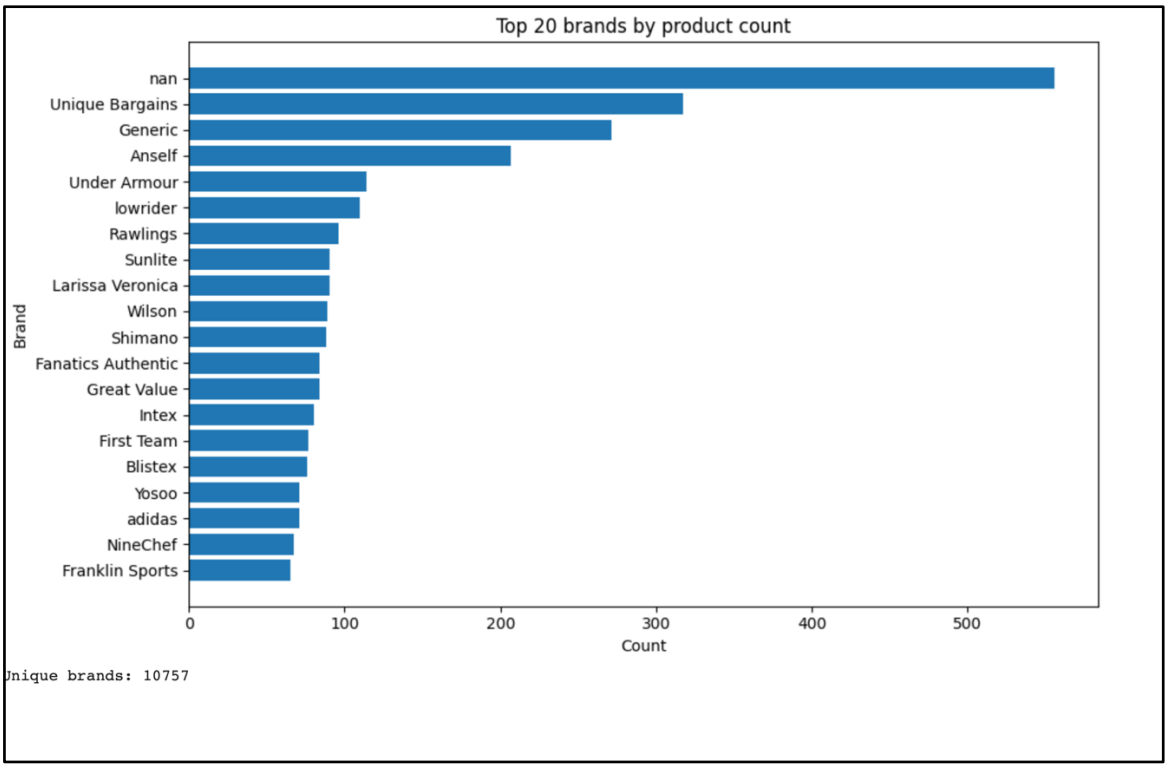
Category and Brand Structure

Exploration of categorical metadata revealed a highly fragmented catalogue. The dataset contained over 3,000 unique categories with many of them following deep, hierarchical naming conventions. Even the most common categories accounted for only a small

proportion of the overall dataset which highlighted the breadth and diversity of the product space.



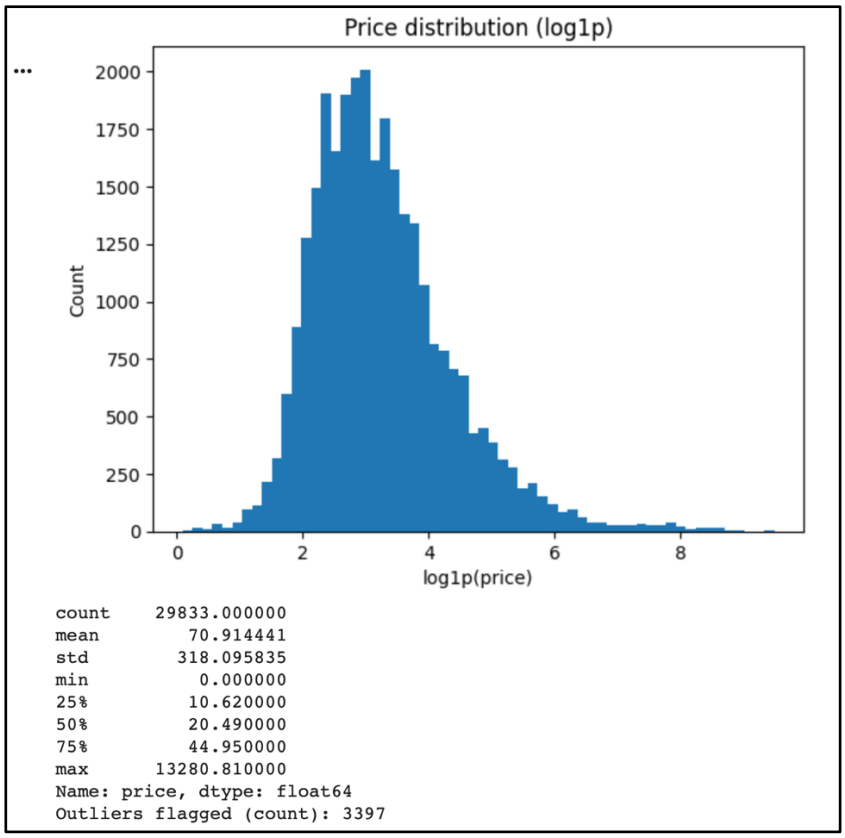
Brand distribution exhibited a similarly long-tailed pattern. A small number of brands were associated with a large number of products. In contrast, more than 10,000 unique brands appeared only once or a handful of times. A noticeable subset of products lacked explicit brand information which reflects an uneven metadata completeness that is common in large e-commerce platforms.



These structural characteristics underline the limitations of relying on category- or brand-based filtering alone and reinforce the motivation for a retrieval approach capable of operating effectively across sparse, widespread and fragmented metadata.

Price Characteristics

Price analysis showed a familiar retail pattern with values concentrated at lower price points and a long tail of higher-priced items. Visualising prices on a logarithmic scale made this structure more apparent.




```
count    29833.000000
mean        70.914441
std        318.095835
min          0.000000
25%         10.620000
50%         20.490000
75%         44.950000
max      13280.810000
Name: price, dtype: float64
Outliers flagged (count): 3397
```

Price was not treated as a primary retrieval feature in keeping with the project's focus on semantic retrieval. However, it was important to have view on and to understand the pricing profile. Price was, nevertheless, retained as a supporting attribute that could later be used for re-ranking, interpretation, or business-oriented analysis.

Step 3A Summary

The preprocessing and exploratory analysis confirmed that the dataset is well suited to the intended task. Textual coverage is strong, duplication is limited and manageable, and the catalogue exhibits realistic long-tail structures across categories, brands, and prices. The analysis simultaneously surfaced considerations around variability in description richness and metadata completeness that inform both modelling decisions and later discussions of bias.

With the data cleaned and its characteristics clearly understood, the analysis continues in Step 3B with the construction of baseline and semantic feature representations for retrieval.

Step 3B: Feature Engineering & Dimensionality Reduction

Following on from the preprocessing and exploratory analysis in Step 3A this next phase focused on transforming the cleaned product data into numerical representations suitable for semantic retrieval and similarity-based recommendation. In keeping with the nature of the project challenge of mapping free-text user queries to relevant products the feature engineering steps centred primarily on textual information and were supplemented by structured metadata where appropriate.

Textual feature construction

The key semantic signal in this dataset resides in the product title and description fields. These fields were concatenated into a single textual representation per product, ensuring that both concise identifiers (titles) and richer contextual detail (descriptions) contributed to downstream models. Prior to feature extraction, the text was standardised through lowercasing, punctuation removal, and whitespace normalisation to reduce superficial variance while preserving meaning.

Two parallel representations were then constructed to support a baseline-to-improvement comparison.

Baseline representation: TF-IDF vectors

As a baseline, a TF-IDF (Term Frequency–Inverse Document Frequency) representation was generated from the combined product text. This approach captures term importance by up-weighting words that are frequent within a product description but relatively rare across the overall catalogue.

The resulting TF-IDF matrix was high-dimensional, containing over 350,000 features. While sparse and computationally intensive, this representation provides a strong benchmark and allows for interpretability through inspection of high-weight terms. Example outputs confirmed that the model surfaced intuitively meaningful keywords (e.g. product type, functional attributes, technical specifications), validating its suitability as a baseline retrieval method.

This baseline mirrors earlier coursework patterns, where simpler, more interpretable techniques are established first before introducing more expressive models.

```
TF-IDF matrix shape: (29833, 352068)
Number of features: 352068

--- Example product ---
Title: Camille Beckman Glycerine Hand Therapy, French Vanilla, 8 Ounce.
Top TF-IDF terms: [('beckman', 0.3161951340915801), ('camille beckman',
0.3161951340915801), ('camille', 0.3072813145459152), ('idaho',
0.13098577400770972), ('hand therapy', 0.09823933050578229), ('community',
0.09584130296781826), ('small', 0.0770572728577799), ('glycerin',
0.07657281553555927), ('business', 0.07514839408390006), ('eagle idaho',
0.07026558535368448)]

--- Example product ---
Title: TSV 3 In 1 Bike Headlights, Loud Bike Horn Front and Power Bank- USB
Rechargeable & Solar Powered Bright Bicycle Front Light, Quick Release
Cycling Flashlight, 4 Lighting Modes, 5 Sound Modes
Top TF-IDF terms: [('5h', 0.3536699670881457), ('modes',
0.25330244130623264), ('sound modes', 0.22430021284948432), ('horn',
0.21711205317496426), ('headlight', 0.21061730358405822), ('power bank',
0.18372914262807083), ('lighting modes', 0.18050681418479975), ('bank',
0.1599707986025909), ('ipx4', 0.14283104859317958), ('solar powered',
0.1320040914248845)]

--- Example product ---
Title: A-King ProHD 1080P Wireless WiFi IP/Network Security Camera For Baby
Monitor with clear Night Vision
Top TF-IDF terms: [('clear night', 0.32322952331744975), ('security
camera', 0.32322952331744975), ('monitor clear', 0.32322952331744975),
('baby monitor', 0.27931963472011473), ('network', 0.25871273595945643),
('ip', 0.25607168275873315), ('wifi', 0.2524820771985598), ('1080p',
0.24633232343506767), ('night vision', 0.23816576964669356), ('king',
0.21248013140640917)]

Saved vectorizer to: ../artifacts/tfidf_vectorizer.joblib
Saved TF-IDF matrix to: ../artifacts/tfidf_matrix.npz
```

Semantic representation: sentence-level embeddings

To move beyond keyword matching and capture semantic similarity, a lightweight pretrained sentence transformer (all-MiniLM-L6-v2) was used to generate dense vector embeddings for each product. This model maps text into a 384-dimensional space in which semantically similar items are positioned closer together, even when they do not share exact vocabulary.

Embeddings were generated in batches to manage memory constraints, and the resulting vectors were L2-normalised. Inspection of embedding norms confirmed stable, well-formed representations. This embedding approach enables more flexible matching between user queries and products, particularly in cases where users describe items imprecisely or use non-standard terminology.

The embedding matrix (≈30,000 × 384) represents a substantial reduction in dimensionality relative to TF-IDF, while retaining richer semantic structure.

Dimensionality reduction and structure inspection

To better understand the structure of the embedding space, Principal Component Analysis (PCA) was applied. PCA was used both diagnostically and practically - diagnostically to assess redundancy and variance concentration, and practically to support downstream visualisation and potential efficiency gains.

Results showed that:

- Approximately 35 components captured 50% of the variance
- Around 113 components captured 80% of the variance
- Roughly 167 components captured 90% of the variance

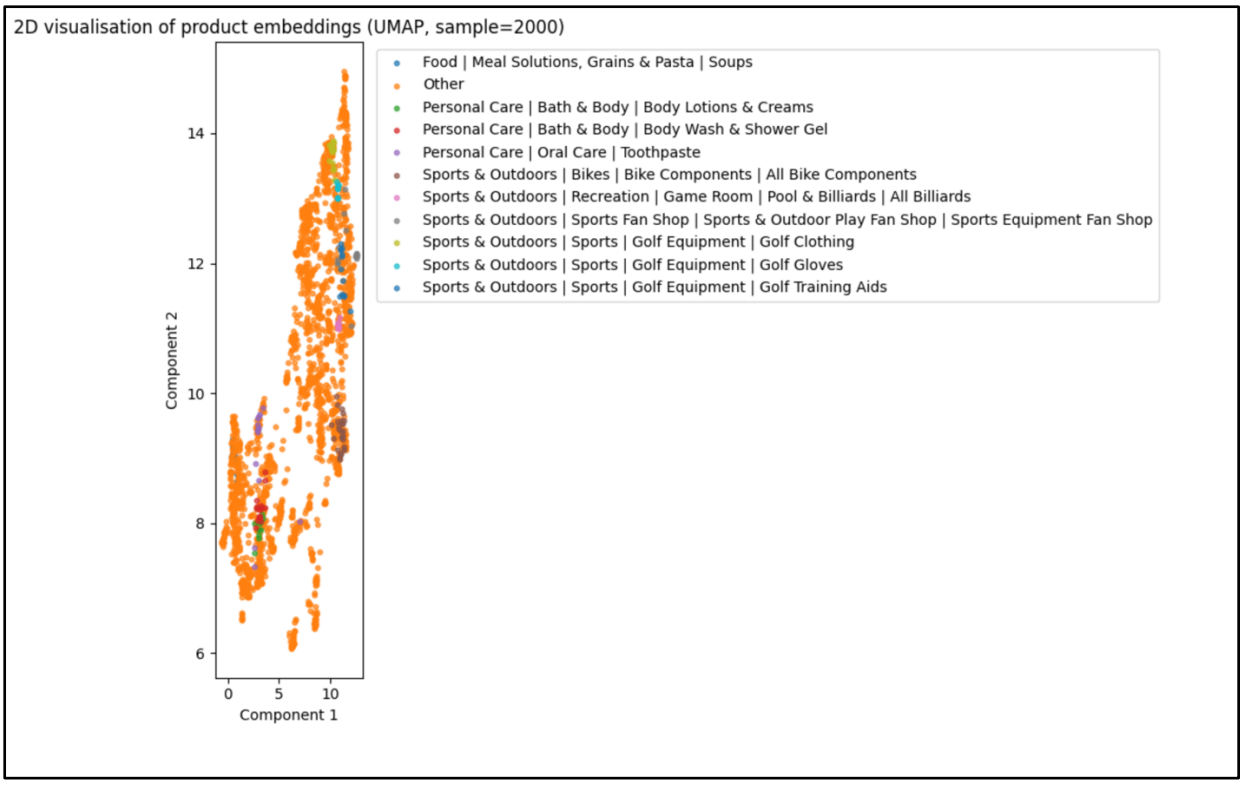
```
Components for 50% variance: 35
Components for 70% variance: 78
Components for 80% variance: 113
Components for 90% variance: 167
Components for 95% variance: 211

First 10 explained variance ratios:
[0.07043967 0.03547899 0.02938434 0.02642601 0.02518709 0.0198965
 0.01823317 0.01610264 0.0150093  0.0146755 ]

Saved PCA model to: ../artifacts/pca_full_on_embeddings.joblib
```

This confirms that the original embedding space contains meaningful redundancy and that lower-dimensional representations could be used without substantial information loss.

For visual inspection, a 2-dimensional projection was generated using UMAP on a sampled subset of products. The resulting plot revealed coherent groupings aligned with high-level product categories (e.g. food items, personal care, sports equipment), providing qualitative evidence that the embeddings capture semantic structure consistent with domain expectations.



Rationale and linkage to downstream modelling

Together, these feature engineering steps establish a clear progression:

- TF-IDF provides a transparent, keyword-driven baseline
- Sentence embeddings introduce semantic understanding
- PCA and UMAP support interpretability, diagnostics, and scalability

This feature pipeline directly supports the next phase of the project, where similarity search, ranking, and evaluation will be performed. It also creates a foundation for later explainability and bias analysis by making both sparse (TF-IDF) and dense (embedding-based) representations available for inspection and comparison.

While no explicit feature selection was applied to the TF-IDF representation, this was a deliberate choice given its role as a transparent baseline for keyword-based retrieval. Dimensionality reduction was instead applied to the dense embedding space, where empirical analysis showed substantial redundancy and meaningful variance compression without significant information loss.

Step 4: Model Implementation & Evaluation

The objective of this step was to evaluate the effectiveness of two different retrieval approaches for mapping free-text user queries to relevant products within the Walmart e-commerce catalogue:

1. A TF-IDF–based lexical retrieval model, serving as a traditional keyword-driven baseline.
2. A semantic retrieval model based on pretrained sentence embeddings, designed to capture user intent beyond exact term matching.

Both approaches were evaluated using a combination of quantitative, category-based ranking metrics and manual qualitative analysis, in line with common industry practice in recommender and information-retrieval systems.

In line with the problem being framed as a content-based recommendation and retrieval task, model experimentation focused on approaches naturally aligned with ranking and similarity-based retrieval, instead of on supervised classifiers or clustering models that are less appropriate for this use case.

Models Implemented

Baseline: TF-IDF Retrieval

The baseline model represents each product using a TF-IDF vector constructed from the concatenation of cleaned product titles and descriptions. Queries are transformed into the same vector space, and cosine similarity is used to rank products.

This approach reflects a standard keyword-based search mechanism and provides a meaningful point of comparison for more expressive models.

Improved Model: Semantic Embedding Retrieval

The improved model uses a pretrained SentenceTransformer (all-MiniLM-L6-v2) to encode both products and queries into a shared 384-dimensional semantic embedding space. Embeddings are L2-normalised, allowing cosine similarity to be computed efficiently via dot products.

This approach is designed to capture semantic similarity and user intent even when queries do not share explicit keywords with product metadata.

Quantitative Evaluation: Category-Based Metrics

Evaluation Design

To enable scalable evaluation without explicit relevance labels, a category-based relevance proxy was used:

- Each product title was treated as a query.

- Products sharing the same category were considered relevant.
- A random sample of 300 products with known categories was evaluated.
- The query product itself was excluded from its relevance set.

While category membership is an imperfect proxy for relevance, it provides a consistent and defensible basis for comparative evaluation across models.

Metrics

The following ranking metrics were computed:

- Precision@K – proportion of retrieved items in the top-K that are relevant.
- Recall@K – proportion of relevant items retrieved in the top-K.
- nDCG@K – ranking quality measure that accounts for the position of relevant items.

Metrics were evaluated at K = 5 and K = 10**.**

Results (Mean across 300 queries)

Metric	TF-IDF	Embeddings
Precision@5	0.359	0.371
Recall@5	0.074	0.077
nDCG@5	0.316	0.322
Precision@10	0.361	0.362
Recall@10	0.140	0.139
nDCG@10	0.345	0.344

Interpretation

The semantic embedding model demonstrates consistent, yet modest, improvements over the TF-IDF baseline, particularly in nDCG@5 and Precision@5. This suggests that the embedding-based approach tends to surface more relevant products earlier in the ranked list.

The relatively small absolute differences are expected given:

- the broad and noisy nature of category labels, and
- the use of category membership as a proxy rather than true user relevance.

Importantly, the results show no degradation in performance when moving from the baseline to the more expressive model.

Qualitative Evaluation: Manual Query Analysis

To complement the quantitative metrics, a set of manually constructed, natural-language queries was evaluated. These queries were intentionally descriptive and non-keyword-specific, reflecting realistic user behaviour.

Example 1:

"a gentle vanilla scented hand cream for dry skin"

- TF-IDF retrieved generic hand creams, with limited sensitivity to scent or sensory attributes.
- Embeddings surfaced vanilla-adjacent and premium skincare products, capturing the intended sensory and usage context more effectively.

This highlights the embedding model's ability to encode nuanced product attributes beyond surface terms.

Example 2:

"bike headlight with a loud horn and solar charging"

- TF-IDF retrieved one highly relevant item but quickly drifted toward loosely related products (e.g. toys, unrelated electronics).
- Embeddings maintained a consistent focus on bicycle lighting products that aligned with the combined intent of lighting, horn functionality, and charging method.

This illustrates improved intent preservation across multiple attributes.

Example 3:

"wifi baby monitor camera with night vision and 1080p"

- TF-IDF retrieved several wildlife or security cameras due to shared keywords.
- Embeddings consistently prioritised baby monitors, correctly inferring the intended use case rather than over-weighting technical specifications.

This example clearly demonstrates the semantic model's advantage in disambiguating context.

Model Comparison Summary

Taken together, the results indicate that:

- The TF-IDF model performs reasonably well for straightforward, keyword-aligned queries.
- The embedding-based model provides more robust behaviour for natural-language, intent-driven queries.

- Quantitative gains are incremental but consistent.
- Qualitative analysis reveals clearer advantages in relevance, coherence, and user-perceived quality.

This mirrors real-world recommender system development, where semantic models are often adopted not solely for metric improvements, but for improved user experience and reduced search friction**.**

Step 5: Critical Thinking, Ethical AI & Bias Auditing

The purpose of this step is to critically examine the behaviour of the retrieval models developed in Step 4, with particular attention to explainability, bias, fairness, and practical limitations. Rather than treating ethical AI as an abstract concern, this step focuses on concrete risks that arise in real-world recommender and search systems, and on feasible mitigation strategies that could be applied in practice.

The analysis draws directly on the outputs and observations from Step 4, combining lightweight diagnostic checks with reflective interpretation.

Step 5A: Model Explainability

Explainability of the TF-IDF Baseline

The TF-IDF model is inherently interpretable. For any given query–product match, relevance can be explained in terms of explicit term overlap and weighting. For example, inspection of top-weighted TF-IDF terms for retrieved products showed that:

- Highly ranked items often shared exact keywords with the query (e.g. *“night vision”*, *“bike headlight”*).
- Retrieval quality degraded when queries relied on descriptive phrasing rather than specific product terms.

This transparency makes TF-IDF easy to debug and explain, but also exposes its limitations in handling ambiguous or natural-language queries.

Explainability of the Embedding-Based Model

In contrast, the embedding-based model operates in a dense semantic space, making direct feature-level explanations less accessible. Individual dimensions of the embedding vectors are not human-interpretable, and relevance emerges from learned contextual similarity rather than explicit term matching.

Explainability was therefore approached indirectly through:

- Nearest-neighbour inspection, examining which products were retrieved for a given query.
- Behavioural comparison with the TF-IDF baseline.

Manual evaluation demonstrated that the embedding model consistently prioritised products aligned with intended use cases, even when keyword overlap was weak. For example, baby monitor queries retrieved baby-focused products rather than generic security cameras, suggesting that the model encodes higher-level contextual meaning.

This highlights a key trade-off, viz. as representational power increases, interpretability decreases, requiring different forms of explanation.

Step 5B: Bias & Fairness Analysis

Rather than focusing on demographic bias (which is not directly supported by the available data), the analysis concentrates on structural and representation biases relevant to e-commerce platforms.

Category Dominance Bias

The dataset contains highly uneven category distributions, with certain categories (e.g. sports equipment, personal care) appearing far more frequently than others. As a result:

- Products from dominant categories are more likely to appear in top-K results.
- Category-based relevance metrics may favour these categories by construction.

This effect was observable during evaluation, where retrieval performance varied across categories with different levels of representation.

Brand Popularity Bias

Brand frequency analysis showed a long-tail distribution, with a small number of brands accounting for a disproportionate number of products. This creates a risk that:

- High-frequency brands dominate retrieval results.
- Smaller or niche brands are systematically under-represented.

Embedding-based retrieval, in particular, may amplify this effect by clustering similar branded descriptions closely in semantic space.

Price-Tier Bias

Although price was not used explicitly in ranking, exploratory analysis indicated skewed price distributions across categories. This raises the possibility that:

- Lower-priced, mass-market items appear more frequently in results.
- Premium or specialised products may be under-retrieved unless explicitly specified in the query.

While not formally quantified in this step, this bias is relevant for downstream business decisions and user experience.

Step 5C: Model and Data Limitations

Several limitations were identified:

- Proxy relevance
    - Category membership is an imperfect substitute for true user relevance and may mask finer-grained intent.
- Pretrained embedding bias
    - The semantic model reflects patterns learned from large external corporates, which may not fully align with Walmart-specific terminology or merchandising priorities.
- Lack of personalisation
    - Retrieval is query-driven and does not incorporate user history, preferences, or context.
- Cold-start constraints
    - New or sparsely described products may be poorly represented in the embedding space.

These limitations are not flaws in implementation but reflect realistic constraints in early-stage recommender systems.

Step 5D: Mitigation Strategies

Several practical mitigation strategies could be applied in a production setting:

- Diversity-aware re-ranking
    - Introducing constraints to ensure variety across brands or subcategories within the top-K results.
- Category-aware balancing
    - Adjusting ranking scores to reduce dominance by overrepresented categories.
- Hybrid ranking logic
    - Combining semantic similarity with business rules (e.g. availability, price range, or strategic brand exposure).

- Human-in-the-loop review
    - Periodic qualitative audits of retrieval behaviour to detect unintended patterns or drift.

These interventions are lightweight, interpretable, and compatible with the existing retrieval pipeline.

Step 5 Summary

This step demonstrates that while semantic retrieval offers clear benefits in capturing user intent, it also introduces new challenges around transparency and bias. By explicitly identifying these trade-offs and proposing feasible mitigation strategies, the analysis reinforces the importance of responsible model deployment alongside performance optimisation.

The findings from this step inform not only future technical enhancements but also governance and monitoring considerations in real-world e-commerce environments.

Conclusion

This project demonstrated the end-to-end application of the machine learning lifecycle to a semantic product search and recommendation problem in an e-commerce context. Starting from careful problem framing and data understanding, the work progressed through preprocessing, feature engineering, model implementation, and evaluation, with each stage informing the next. Comparative analysis showed that embedding-based semantic retrieval offers meaningful improvements over traditional lexical approaches, particularly for natural-language, intent-driven queries, while also introducing important considerations around interpretability and bias. Overall, the project highlights the value of aligning model choice with problem structure, balancing technical capability with transparency, and approaching deployment with a critical awareness of limitations and ethical implications.