Automated Customer Reviews Using Natural Language Preprocessing (NLP)

A Business Case Report

Team Members:
Mehdar - Randa - Faisel

1. Introduction

1.1 Problem Statement

With e-commerce platforms expanding, customers are increasingly overwhelmed by the massive volume of product reviews, making it difficult to make informed purchasing decisions. This project addresses the problem by automating the review analysis process using Natural Language Processing (NLP). Resulting to generate clean, blog-style, human-readable product summaries to assist users in decision-making.

***Goal**: Develop a fully automated pipeline to automate the process of analyzing customer reviews.*

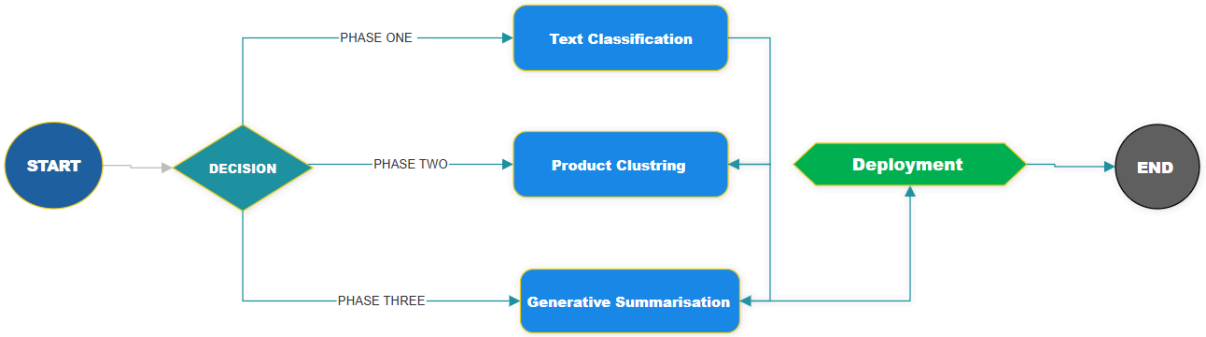
2. Scope of the Project

Our project workflow was strategically divided into three critical phases: Review Classification, Clustering, and Generative Summarization with GPT to ensure logical and efficient progress.

Our pipeline consists of three interconnected NLP phases:

1. Review Classification (sentiment-Analysis)
2. Product Category Clustering
3. Generative Summarization with GPT

2.1. Pipeline Overview



Each phase is designed to build logically on the previous phase, ensuring data consistency and contextual accuracy throughout the transformation process

.3. Phase 1: Review Classification (Sentiment Analysis) :

In this phase, we developed a sentiment classification model to categorize product reviews into positive, neutral, or negative. The sentiment labels were derived from star ratings:

- Ratings 1–2 were mapped to negative
- Rating 3 was mapped to neutral
- Ratings 4–5 were mapped to positive

We used a pre-trained DistilBERT model fine-tuned on our labeled dataset. During preprocessing, reviews were cleaned by lowercasing, truncating, and tokenizing. A major issue in the dataset was class imbalance, where most reviews were positive. To solve this, we applied oversampling to increase the number of neutral and negative examples.

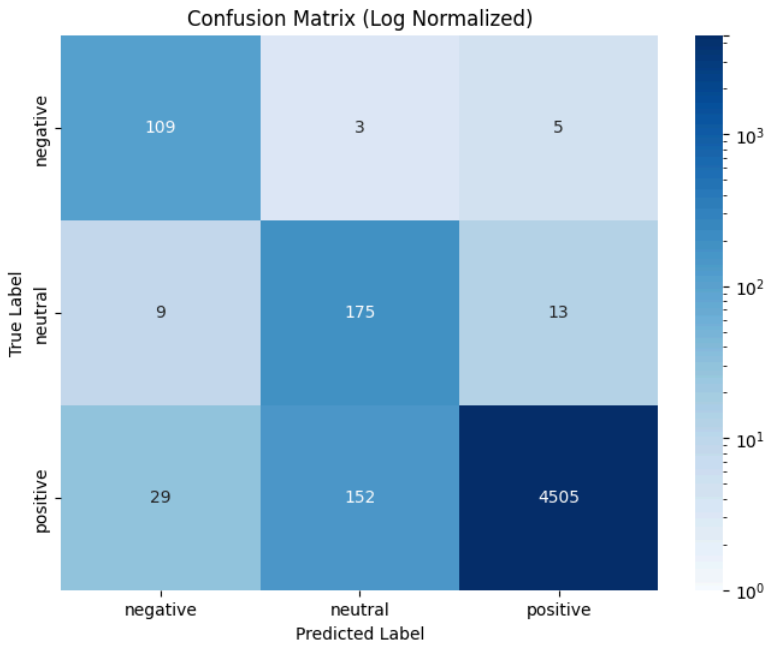
This helped the model learn to classify all three sentiments more effectively.

The model was then evaluated on a separate test set of 5000 samples. The true sentiment was derived from the "reviews.rating" column and compared to the predicted labels. Below is the classification report

	precision	recall	f1-score	support
negative	0.74	0.93	0.83	117
neutral	0.53	0.89	0.66	197
positive	1.00	0.96	0.98	4686
accuracy	-	-	0.96	5000
macro avg	0.76	0.93	0.82	5000
weighted avg	0.97	0.96	0.96	5000

The model achieved 96% accuracy. It performed very well on the positive class, and thanks to oversampling, it also handled neutral and negative classes much better than before.

4.1. Confusion Matrix:



This confusion matrix illustrates the **performance of the sentiment classification model** across three classes: **negative**, **neutral**, and **positive**.

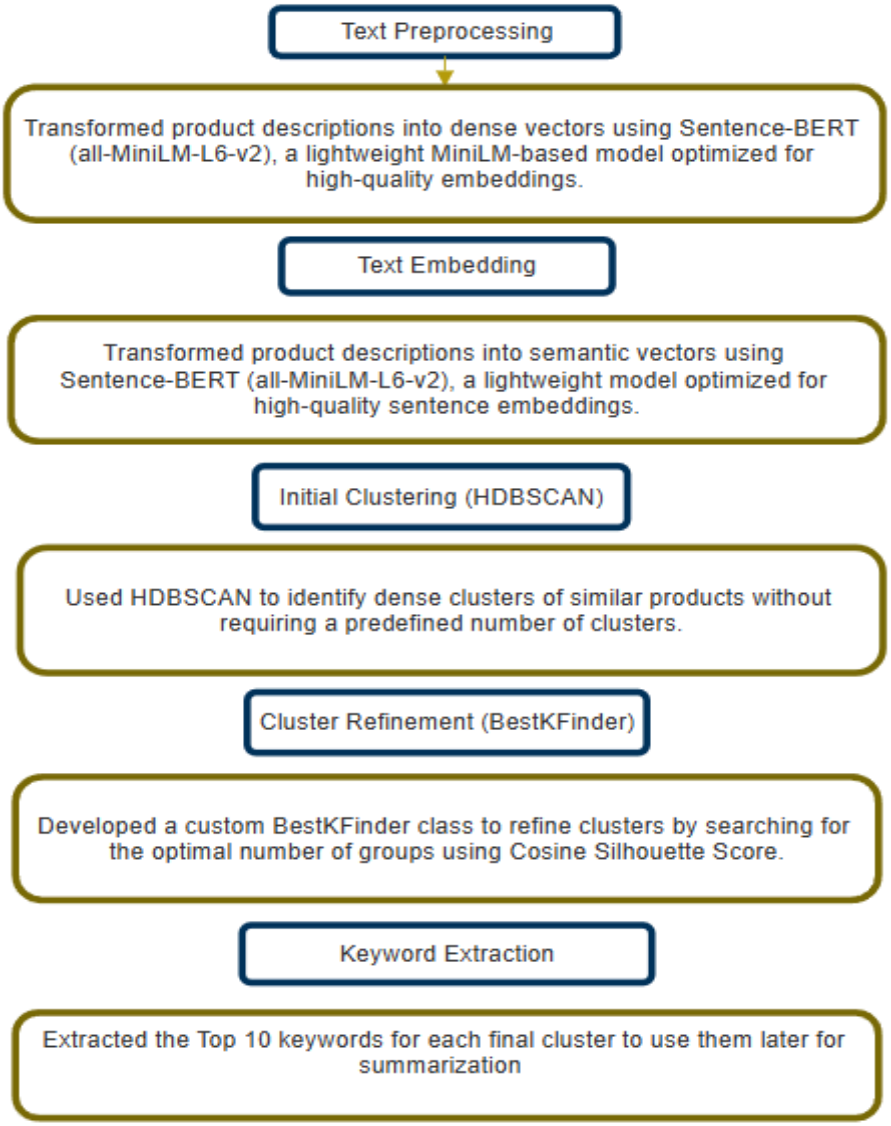
- The model performs very well on **positive reviews**, correctly identifying **4,505** out of the total, with minor confusion with neutral (152) and negative (29).
- **Neutral reviews** are also well predicted, with **175** correct predictions, though a small portion is misclassified as positive (13) or negative (9).
- For **negative reviews**, the model correctly predicted **109**, with very few misclassifications.

4. Phase 2: Product Category Clustering:

In this phase, we aimed to group products into meaningful categories based on their names and associated metadata. The goal was to cluster similar products together without predefined labels, enabling better organization and downstream processing for summarization. To achieve this, we used advanced Natural Language Processing (NLP) techniques combined with unsupervised clustering algorithms.

4.1 Methodology

The Product Category Clustering phase was completed in several steps:



4.2 Detailed Implementation

4.2.1. Text Cleaning:

- Removed frequent noise words (e.g., "wifi", "display", "32gb").
- Removed numbers and special characters.
- Applied lemmatization and customized stopword filtering.

4.2.2. Embedding Generation:

Each product was encoded into a 384-dimensional semantic vector using Sentence-BERT. And it provides an excellent balance between performance and computation cost, making it ideal for clustering large volumes of product data efficiently.

4.2.3. Experimental Trials: HDBSCAN Clustering

In order to determine the best input features for clustering, three different approaches were tested:

	Parameters	Silhouette Score	Initial output
Product Name + Categories	min_cluster_size = 150	99%	22 clusters identified
			709 noise points (outliers)
Reviews Text + Categories	min_cluster_size = 150	36%	4 clusters identified
			14070 noise points (outliers)
Categories	min_cluster_size = 150	99%	19 clusters identified
			538 noise points (outliers)

4.2.4. Selected Approach

Based on the comparative analysis of the different input combinations tested, the **Product Name + Categories** combination was selected as the final clustering approach.
This decision was supported by the following observations:

- **Highest Silhouette Score:** Achieved a 99% silhouette score, indicating highly coherent clusters.
- **Rich Semantic Context:** Combining product names with categories provided a deeper semantic understanding, resulting in more meaningful groupings.
- **Minimal Noise:** Only 709 noise points were detected, compared to significantly higher noise levels in other combinations.
- **Well-Formed Clusters:** Clear and distinct cluster structures were visible in the UMAP visualization, confirming effective separation.

While using **Categories only** also achieved a 99% silhouette score, it lacked semantic richness, often grouping dissimilar products under the same category.
On the other hand, Reviews Text + Categories introduced excessive noise and yielded poor clustering performance.

Thus, **Product Name + Categories** was adopted to ensure robust, interpretable, and scalable clustering outcomes for downstream analysis and summarization tasks.

4.3 Results and Observations

4.3.1 HDBSCAN Cluster Visualization

The figure below shows a two-dimensional UMAP projection of the product embeddings clustered using HDBSCAN.

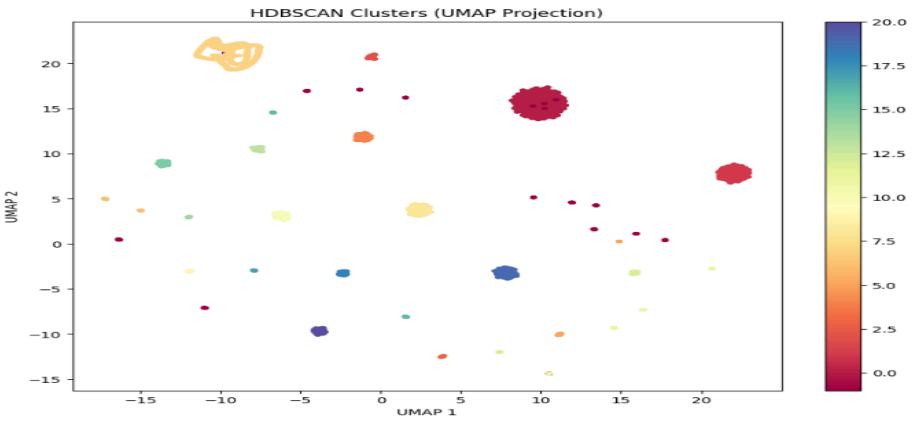


Figure 4.3.1.1: HDBSCAN Clusters (UMAP Projection) using Name + Categories

Observations:

1. Each point represents a product embedding.
2. Different colors represent different clusters identified by HDBSCAN.
3. The shape and spread of the clusters reflect the semantic similarity between product names and categories.
4. Outliers are scattered outside the dense groups.

Summary:
Multiple well-formed clusters appear, reflecting high semantic similarity among product groups, with expected noise due to variability in e-commerce product descriptions.

UMAP was used to reduce the high-dimensional Sentence-BERT embeddings (384 dimensions) into two dimensions for easier visualization without losing local structure.

UMAP was used to reduce the high-dimensional Sentence-BERT embeddings (384 dimensions) into two dimensions for easy visualization without losing the local structure of the data. After applying the BestKFinder refinement, the optimized clusters show improved separation as illustrated below.

4.4 Final Optimization and Observations:

After the initial HDBSCAN clustering, we refined the clusters using BestKFinder.

- Refinement Method: Custom BestKFinder class using Cosine Silhouette Score.
- Search Range: Between 2 and half the number of HDBSCAN clusters.
- Final Output: 7 optimized clusters.

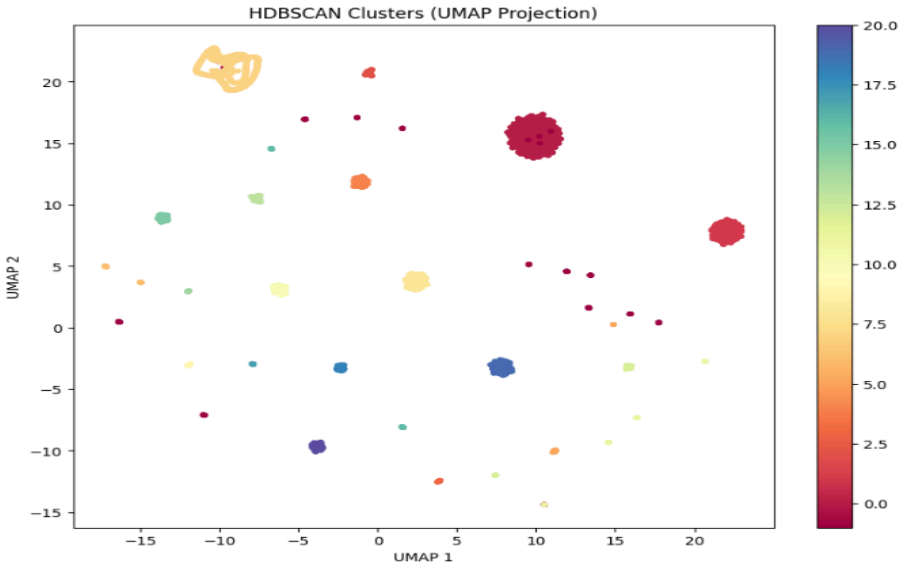


Figure 4.4.1: Optimized Clusters (KMeans BestKFinder output) - UMAP Projection

Observations:

- Clusters are now larger, denser, and better separated compared to the initial HDBSCAN output.
- Products within the same cluster are highly semantically similar.
- Final clusters provide a strong foundation for coherent product categorization.

4.5 Outputs

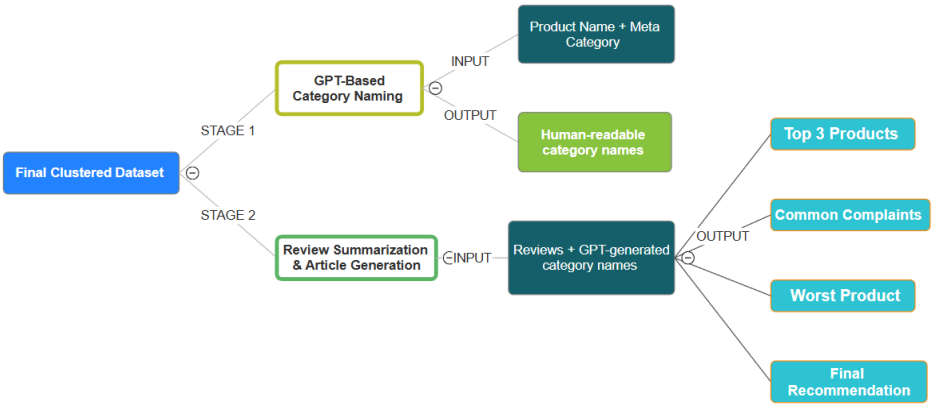
At the end of Phase 2, we produced a final clustered dataset containing:

5. Phase 3: AI-Powered Review Analysis & Summarization

This task was set to Provide detailed, blog-style review summary articles using generative AI. The core aim is to transform unstructured customer reviews collected across various product categories into well-organized, narrative-driven recommendation articles.

5.1 Smart Automation Pipeline Flowchart (Two Stages)

By using the clean, clustered data, structured, human-readable product recommendations for each category were generated. Following the pipeline below:



Output: Structured Blog Article Generation for users

5.2. Model Configuration

Model Chosen:
GPT-4 prompt generator

Parameter	Value
Model	GPT-4
Temperature	0.7
Max Tokens	1200
Input Format	JSON
Output Format	Markdown

Table 1. GPT4 Prompt used for blog-style summarization

5.3. Results & Evaluation

5.3.1. Prompt Evaluation Methodology:

To validate the effectiveness of our prompt design, we employed the OpenAI ChatGPT API as a benchmarking tool. The API was utilized to:

- Generate article-style outputs based on selected review inputs and category context.
- Assess the clarity, coherence, and informativeness of the generated summaries.

5.3.2. Outputs for Prompts Comparison:

Comparison of summarization quality between in-depth structured and short prompt designs for the user:

Prompt Type 1	Long-Form Structured Prompt (Used in GPT-4)

Model	GPT-4		
Temperature	0.7	Limit Tokens	1200
Prompt	prompt = f""" You are a senior product analyst with expertise in interpreting customer behavior across varied contexts. Given customer reviews for the category [{meta_category}], write a well-structured blog-style article that includes the following: --- 1. **Top 3 Highest-Rated Products** - List the top 3 products based on average ratings. - Summarize their key strengths in bullet points. - Highlight unique features or differences between them. - If applicable, explain how user context (e.g., frequent travelers, gamers, remote workers) influenced their positive experiences. 2. **Most Common Complaints per Product** - For each top product, summarize the most frequent issues or complaints mentioned. - Pay close attention to issues related to **environmental or contextual conditions** (e.g., hot or cold weather, indoor vs outdoor use, long vs short usage time), **explicitly mention that context**.		
Output	# Analyzing Amazon's Fire Tablet Range: A Comprehensive Product Review In this blog post, we are going to take a thorough look at Amazon's Fire Tablet range. Based on customer reviews, we'll highlight the top three highest-rated products, their strengths, and key differences. We'll also delve into the most common complaints for each product, and warn you about the worst-rated product and why it should be avoided. ## Top 3 Highest-Rated Products ### 1. Fire HD 10 Kids Edition The Fire HD 10 Kids Edition is highly rated by customers for its **excellent parental controls**, making it a fantastic choice for parents who want control over their children's screen time and content. It also boasts a **durable design** that can withstand the rigors of child use, and an impressive **battery life** that allows for extended use. ### 2. Fire HD 8 Tablet The Fire HD 8 Tablet is another popular choice, particularly for individuals looking for a budget-friendly tablet. It's **affordable** and offers a **decent performance for content streaming**, making it great for Netflix binges. Its **small form factor** also means it's highly portable and easy to hold for extended periods. ### 3. Fire 7 Tablet (2022) The Fire 7 Tablet (2022) is lightweight and compact, making it ideal for on-the-go use. It runs **basic apps without any issues**, and is generally seen as a **good tablet for children** due to its simplicity and ease of use.		

Table 2. An in depth analysis prompt before we compare it with a short prompt

From Table 2, the in-depth analysis using a structured storytelling prompt clearly demonstrates GPT-4's ability to build a cohesive narrative, present product arguments, and conclude with well-aligned, personalized recommendations. However, as shown in Table 4, the output generated from a short direct prompt—while concise and factual informative—lacks the richness of storytelling, product comparison context, and narrative continuity required for engaging consumer-oriented articles.

Prompt Type 1	Direct and minimal		
Model	GPT-4		
Temperature	0.7	Limit Tokens	1000
Prompt	You are a smart product review summarizer. Based on the following data from category "{category}", generate a short article: 1. Recommend the top 3 products: {', '.join(top3)} 2. Explain the main differences between them. 3. Summarize common complaints about each. 4. Warn about the worst product: {worst}, and explain why it should be avoided. Top reviews: {reviews_top} Worst reviews: {reviews_worst} Write the article in an informative, blog-like tone. """		
Output	### Top 3 Highest-Rated Products 1. Fire HD 10 Kids Edition: Excellent parental controls, durable design, long battery life. 2. Fire HD 8 Tablet: Budget-friendly, great for streaming, compact size. 3. Fire 7 Tablet: Affordable but limited storage, still suitable for light usage. ### Common Complaints - Slow charging speed - Limited app support - Occasional touchscreen lag ### Worst Product Fire 7 (older model): Frequent crashes, outdated OS, poor resolution. ### Final Recommendation For families: Fire HD 10 Kids For casual users: Fire HD 8 Avoid: Fire 7 (2019 version)		

Table 3. Short Direct prompt

This evaluation process allowed us to identify strengths and limitations in prompt structure and refine the design to ensure consistent, high-quality summarization across varied product categories.

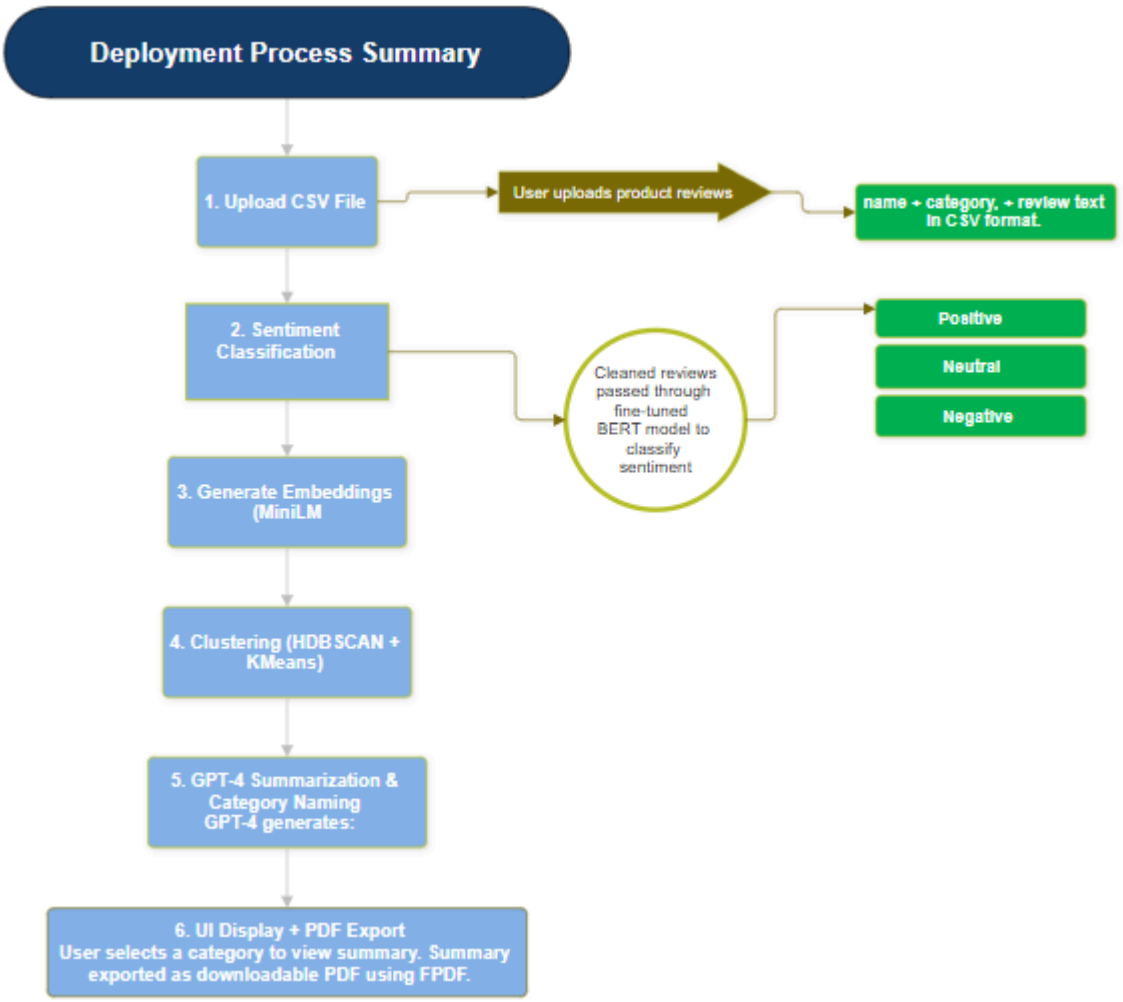
5.4. Feature-to-Benefit Table

Table 4. shows a benefit for each feature that has been used in GPT-4:

Feature	Benefit
GPT-powered Category Naming	No need to manually label clusters
Prompt-driven Summarization	Adaptable tone, content, and structure
Fully Automated pipeline	Scalable and reproducible for any domain
PDF Export	Ready for blog deployment

6. Deployment Strategy

The deployment process for the product review analyzer is structured into six streamlined steps, ensuring efficient transformation from raw user data into meaningful insights:



7. Conclusion

This project presents a scalable NLP pipeline that automates customer review analysis by classifying, clustering, and summarizing reviews using GPT-4. The unsupervised clustering effectively groups products into meaningful categories, the proposed pipeline not only classifies and clusters reviews

effectively but also utilizes GPT-4 to create clear and helpful blog-style product summaries. This structure enhances user experience, streamlines content creation, and can be adapted across domains such as Amazon, TripAdvisor, or Yelp.