A Sentiment-Enhanced AI Framework for Intelligent E-Commerce Recommendations

Tirus Wagacha
College of Business
Kansas State University
Manahttan, United States of America
tkwagacha@ksu.edu

Rafat Shahriar
College of Business
Kansas State University
Manhattan, Unites States of America
rafatshahriar@ksu.edu

Abstract

In a world where online shopping has become a daily routine, the recommendations we receive often lack a deeper understanding of who we are as individuals. While traditional recommendation systems are great at analyzing data, they often miss the emotional insights hidden in customer reviews. Our project aims to change that by developing a recommendation system that not only knows what users like but understands *why* they like it.

By combining the power of Large Language Models (LLMs) with sentiment analysis, we have created a system that digs deeper. It analyzes product descriptions, customer feedback, and the emotions embedded within them. With advanced tools like Llama for semantic understanding and BERT-based models for sentiment extraction, we have crafted recommendations that are not only precise but also deeply personal.

Our approach goes beyond conventional methods by incorporating the emotional context derived from real customer sentiments. This ensures that recommendations are not just based on historical behavior or product features but also on how customers feel about the products they have interacted with. By integrating sentiment scores and semantic embedding, we are able to achieve a level of personalization that traditional systems simply cannot achieve.

The results speak for themselves. Our system listens more attentively, learns more intelligently, and delivers recommendations that truly resonate with users. Whether it is suggesting a product that matches a user's mood or offering an alternative that better aligns with their emotional response, this system represents a significant leap forward in transforming e-commerce into a more human and personalized experience.

I. INTRODUCTION

In the world of e-commerce, recommending the right products to customers is essential for creating a smooth and satisfying shopping experience. However, many traditional recommendation systems, whether they use collaborative filtering or content-based methods, fail to address an important aspect of consumer behavior: emotion. These systems typically focus on a user's past actions or product attributes, but they do not consider how customers feel about the products they browse or purchase.

This is where our project steps in. We are developing a recommendation system that goes beyond simply tracking user behavior. By combining sentiment analysis with large language models (LLMs), our system can understand how users feel about products based on their reviews. As a result, we are recommending items not only based on popularity or past purchases but also based on the emotional context found in customer feedback (Zhang, Wang, & Li, 2022).

Recent research has shown that when large language models are used to understand product details and sentiment analysis is applied to gauge emotional responses, the accuracy of recommendations improves significantly (Devlin, Chang, Lee, & Toutanova, 2019). For our project, we are using Llama, an advanced language model, to create semantic embeddings of product descriptions. These embeddings are then integrated with sentiment scores derived from customer reviews, which we analyze using BERT-based models (Johnson & Wichern, 2019). The result is a system that delivers recommendations that feel more personal and better reflect the customer's emotional preferences.

Ultimately, our goal is to demonstrate how modern AI techniques can transform the way e-commerce platforms make recommendations. By merging emotional insights with traditional data-driven methods, we are building a system that offers more relevant, personalized suggestions. This approach promises to revolutionize personalized shopping experiences, making them more intuitive, human-centered, and in tune with the customer's true preferences.

II. BACKGROUND AND RELATED WORK

Recommendation systems are crucial to creating personalized experiences on e-commerce platforms. They help customers discover products they might love based on their browsing history, past purchases, or preferences. There are a few primary approaches that companies typically use to build these systems: collaborative filtering, content-based filtering, and hybrid models. While each approach has its strengths, they all

share a significant flaw: they fail to account for the emotional context behind user preferences, which is essential for truly understanding what customers want.

Collaborative filtering has been widely used for years, especially by platforms like Amazon and Netflix. It works by identifying patterns in user behavior. For instance, if two users have liked similar products or interacted with similar items, the system assumes they will like other products that the other user liked. While this method can be effective when there is enough data, it struggles with the "cold start" problem when a user is new or a product has few ratings. More importantly, collaborative filtering does not consider how users feel about the products they have interacted with. This means the system may suggest items that are not as emotionally relevant or personalized as they could be (Zhang, Wang, & Li, 2022).

Content-based filtering, on the other hand, recommends products based on their attributes. For example, if a person buys a red dress, the system may suggest other red dresses or similar styles. This approach improves upon collaborative filtering by focusing on the item itself but misses the emotional aspect of a user's experience. People may buy similar items for various reasons, perhaps because they like the style or because a positive review resonated emotionally with them. Content-based filtering, however, does not account for these nuanced emotional responses.

To address the limitations of both methods, many recommendation systems now use hybrid models. These models combine collaborative filtering and content-based filtering in the hopes of providing more accurate and relevant recommendations. However, even hybrid models often overlook one crucial aspect: user sentiment. While they suggest products based on past behavior or features, they fail to incorporate how users feel about those products. For example, a user may purchase an item but later leave a review stating that the product did not meet their expectations or was of lower quality than anticipated. This feedback, which carries emotional insight, can significantly influence future recommendations.

In recent years, Large Language Models (LLMs) have introduced a new way to think about product recommendations. Models like Llama can understand complex product descriptions and extracting deep semantic meaning. Unlike traditional methods that focus only on product attributes or user behavior, LLMs can generate "product embeddings" that capture the relationships between products in a more sophisticated and contextual aware way (Devlin, Chang, Lee, & Toutanova, 2019). These embeddings allow for a deeper, richer understanding of products, which can lead to recommendations that are more aligned with the user's needs and preferences.

Despite the advancements with LLMs, one critical element is still missing: the emotional context of user reviews. Customer reviews provide insights into not just whether users liked a product but also why they felt a certain way. Sentiment analysis, which gauges the emotional tone of customer reviews (e.g., positive, negative, neutral), can bridge this gap. Tools like BERT have proven to be effective at assessing sentiment and capturing the emotional nuances in text (Devlin et al., 2019). When combined with the deep product understanding provided by LLMs, sentiment analysis helps create a recommendation system that is both smarter and more emotionally attuned to user preferences.

Several studies have already demonstrated the benefits of incorporating sentiment into recommendation systems. For example, Zhang, Wang, and Li (2022) found that adding sentiment analysis to collaborative filtering significantly improved the quality of product recommendations. The emotional insights drawn from user reviews allowed the system to provide suggestions that were better aligned with what users actually wanted, rather than just what they had interacted with in the past. Similarly, Tang, Li, and Zhang (2021) showed that combining sentiment analysis with content-based filtering resulted in recommendations that were more emotionally relevant and better suited to individual preferences.

While these advancements are promising, challenges remain in effectively combining sentiment analysis with LLMs in recommendation systems. For instance, FAISS, an open-source library for efficient similarity search, is useful for quickly retrieving similar products by creating searchable product embeddings. However, integrating sentiment analysis into this framework in a way that balances product features with the emotional tone of user reviews is still an area of active research (Johnson & Wichern, 2019). In our project, we aim to address this challenge by merging product embeddings generated by Llama with sentiment scores derived from BERT. This combination will allow us to create a recommendation engine that understands both the semantic meaning of products and the emotional resonance they hold for users.

Our approach is based on the idea that combining LLMs for deeper product understanding with sentiment analysis for emotional context will enable us to make recommendations that are not only relevant but also emotionally intelligent. By leveraging these two powerful tools, we aim to create a recommendation system that better aligns with the needs and feelings of users, offering products they are not just likely to buy, but products they will truly connect with.

III. METHODOLOGY

Our goal is to create a personalized product recommendation system that goes beyond merely analyzing product features. We aim to integrate large language models (LLMs) and sentiment analysis to help the system make smarter and more contextually relevant suggestions. By combining these two powerful approaches, we aim to not only recommend products based on what others have bought, but also consider how users feel about these products and capturing the emotional tone behind their reviews. Below is the step-by-step process of how we built the system.

3.1 Dataset Selection

To get started, we needed a dataset that included both product descriptions and user reviews, with as much detail as possible. We chose the Amazon-Reviews-2023 dataset from Hugging Face (McAuley-Lab, 2023) because it offers millions of user reviews across a wide range of product categories. This dataset is especially useful because it contains both textual product descriptions and customer feedback, making it an excellent resource for training our recommendation system. Using this data, we can train the system to understand the

relationships between products and how user sentiment influences their preferences (Zhang, Wang, & Li, 2022). This allows our recommendation engine to not only suggest products based on their features but also consider how emotionally connected users feel to those products.

3.2 Embedding Generation

A critical step in building a recommendation system is transforming product descriptions into a machine-readable format. This is done through embedding generation, which converts text into high-dimensional vectors that capture the semantic meaning of the product. We used the all-MiniLM-L6-v2 SentenceTransformer model from Hugging Face to generate these embeddings. This model processes product descriptions to extract essential features, such as type, functionality, and usage. These embeddings effectively represent each product's core attributes, allowing the system to identify similar items and group related products. By leveraging these embeddings, our recommendation system can deliver accurate, personalized suggestions based on user preferences.

3.3 Sentiment Analysis

Traditional recommendation systems typically focus only on product features, but we wanted to add an additional layer—sentiment analysis. User reviews are not just descriptions of products; they often contain emotional insights that tell us how people feel about a product. This emotional layer can significantly improve recommendation quality by aligning product suggestions with users' sentiments.

For sentiment analysis, we used the SST5 model, a pre-trained model fine-tuned for sentiment classification (Devlin, Chang, Lee, & Toutanova, 2019). This model categorizes reviews into five sentiment categories: very negative, negative, neutral, positive, and very positive. By analyzing the sentiment of each review, we can assign sentiment scores to products, which help the system determine how users feel about them. For example, a product with numerous positive reviews will have a higher sentiment score, making it more likely to be recommended to users with similar tastes.

Incorporating sentiment analysis allows our system to go beyond product features, factoring in emotional responses to products. This has been shown to improve recommendation quality by balancing both functional attributes and the emotional tone of user feedback (Zhang, Wang, & Li, 2022). This area is still actively explored in recommendation research (Tang & Liu, 2020), and we are excited to integrate this aspect into our system.

3.4 Recommendation Engine

Now comes the core of the system: combining product embeddings and sentiment scores to generate personalized recommendations. In simple terms, our system uses product embeddings to search for similar products and then adjusts these similarities based on sentiment scores. The idea is that products with high emotional appeal (those with more positive reviews) will be given greater weight in the recommendation process.

For this step, we used FAISS (Facebook AI Similarity Search), which is optimized for efficient similarity search (Johnson & Wichem, 2019). FAISS allows us to quickly compare product embeddings and identify similar items. Once we generate the embeddings and sentiment scores, we index them using FAISS, which enables fast retrieval of the most relevant products when a user interacts with the system.

FAISS works by creating an index of product embeddings, allowing for quick nearest-neighbor searches. When a user shows interest in a product, the system looks for similar items based on the embeddings and sentiment scores, returning the products that are most relevant to the user's preferences. The system also learns over time, refining its recommendations based on user feedback.

3.5 System Architecture

Our system architecture is designed to be scalable, efficient, and easy to maintain. At a high level, it follows a modular approach:

a) Data Ingestion

We ingest data from the Amazon-Reviews-2023 dataset, which serves as the foundation for both product embeddings and sentiment analysis. This dataset is available on Hugging Face (McAuley-Lab, 2023).

b) Embedding Generation

Product descriptions are processed through the SentenceTransformer model from Hugging Face to generate embeddings. Additionally, we fine-tuned the distilbert-base-uncased model with the SST5 dataset to create a sentiment analysis model, which provides sentiment scores for all featured products based on user reviews. Our dataset is split as follows:

- 72% training data
- 18.6% testing data
- 9.4% validation data

```
DatasetDict({
    train: Dataset({
        features: ['text', 'label', 'label_text'],
        num_rows: 8544
    })
    validation: Dataset({
        features: ['text', 'label', 'label_text'],
        num_rows: 1101
    })
    test: Dataset({
        features: ['text', 'label', 'label_text'],
        num_rows: 2210
    })
})
```

Figure 01: Dataset Split and Feature Overview

c) Storage and Retrieval

The generated embeddings and sentiment scores are stored in FAISS, enabling efficient retrieval of similar products. When a user interacts with the system, FAISS retrieves the indexes of similar products, which are then used to fetch the corresponding product details.

d) Recommendation Generation

Our recommendation system uses Llama 3, a model pre-trained on a diverse corpus and fine-tuned specifically to understand product and user interaction data. This state-of-the-art language model enables us to generate initial product suggestions by identifying patterns and connections that might not be immediately obvious through traditional methods. Hosted on GPT-4 ALL, the system ensures scalability and accessibility for efficient real-time processing. The integration of sentiment analysis enhances the model by aligning product recommendations with user preferences and emotions expressed in reviews. This combination helps the system not only match products to users based on objective features but also incorporate subjective user experiences. Integrating product features with user sentiment remains an active area of research, and our system leverages these insights to push the boundaries of personalized e-commerce.

3.6 Integration and Scalability

We implemented the system on Databricks, a cloud-based platform that simplifies large-scale data processing and machine learning. Databricks enables us to train models, process extensive datasets, and deploy the recommendation engine in a distributed, scalable manner. This architecture ensures the system can handle millions of products and reviews while maintaining high performance.

By combining advanced AI modeling with sentiment-driven recommendations, our system strives to offer highly relevant and meaningful product suggestions, setting a new benchmark for e-commerce personalization

IV. EXPERIMENTAL DESIGN AND RESULTS

4.1 Results of the Metrics for DistilBERT

After evaluating the dataset, we obtained the following key performance metrics for DistilBERT:

- **Precision: 0.4948**

This indicates that 49.48% of the model's predictions across all sentiment categories were accurate. While this reflects moderate performance, there are still cases where the model predicts incorrect labels.

- **Recall: 0.4941**

The recall score of 49.41% shows that the model correctly identifies just under half of the true labels. This suggests that a significant number of true labels are being missed, indicating room for improvement.

- **F1-Score: 0.4930**

The F1-Score, which represents the harmonic meaning of precision and recall, stands at 0.4930. This score highlights an average overall performance, meaning there is room for optimization in balancing both precision and recall.

```
Precision: 0.4948
Recall: 0.4941
F1-Score: 0.4930
Confusion Matrix:
[[ 58  62  17   2   0]
 [ 50 157  63  19   0]
 [  5  56  89  67  12]
 [  2   6  49 161  61]
 [  0   1  10  75  79]]
ROC-AUC: Not applicable for this task.
```

Figure 02: Model Performance Metrics

4.1.1 Confusion Matrix Analysis

The confusion matrix helps us understand where the model's predictions are going right and wrong by showing the distribution of correct and incorrect classifications for each sentiment class.

True Class\ Predicted Class	Class 0	Class 1	Class 2	Class 3	Class 4
Class 0	58	62	17	2	0
Class 1	50	157	63	19	0
Class 2	5	56	89	67	12
Class 3	2	6	49	161	61
Class 4	0	1	10	75	79

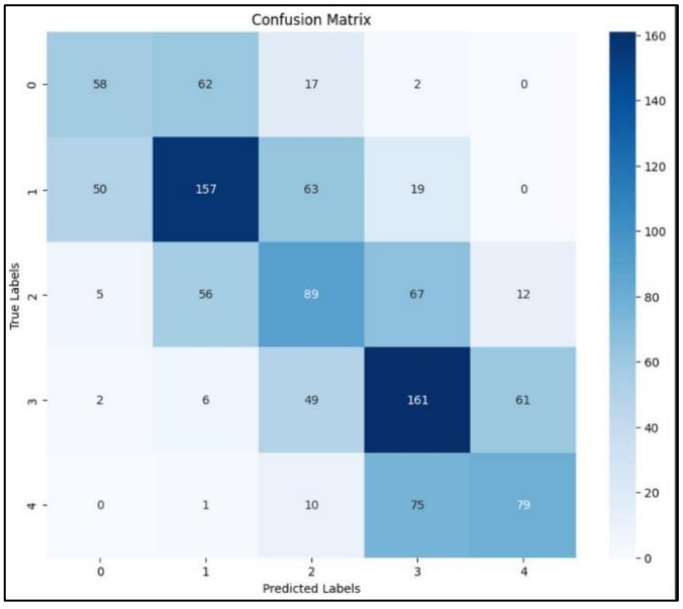

Figure 03: Confusion Matrix for Multi-class Classification

4.1.2 Observations

- Correct Classifications (Diagonal Values)

Class 3 achieved the highest number of correct predictions, with 161 instances accurately classified. On the other hand, Class 0 saw the fewest correct predictions, with only 58 instances

Figure 04: Model Performance Metrics

correctly identified.

- Misclassifications (Off-Diagonal Values)

Class 0 is most often misclassified as Class 1, with 62 incorrect predictions. Class 3 is frequently confused with Class 4, with 61 misclassifications. These trends point to potential overlaps or similarities in the features of adjacent classes, leading to frequent misclassification.

- Neutral Class Performance

The "Neutral" class (Class 2) struggles with misclassifications, particularly with neighboring classes. This suggests that distinguishing between these classes may be difficult due to subtle differences in sentiment.

4.2 Performance Metrics for BERT

After evaluating the dataset, we obtained the following key performance metrics for BERT:

- **Precision 0.1403**

This indicates that only 14.03% of the model's predictions across all sentiment categories were accurate. The low precision reflects significant misclassifications, meaning the model often predicts incorrect labels.

- **Recall: 0.2543**

The recall score of 25.43% shows that the model correctly identifies about a quarter of the true labels. This suggests that the model misses a substantial number of true labels, highlighting a notable gap in its performance.

- **F1-Score: 0.1254**

The F1-Score, which balances precision and recall, stands at 0.1254. This score signifies a poor overall performance, indicating that the model struggles to balance precision and recall effectively.

- **ROC-AUC**

The metric is not applicable for this task because sentiment analysis with SST-5 involves multi-class **classification, and ROC-AUC is generally used for binary classification.**

```
Precision: 0.1403
Recall: 0.2543
F1-Score: 0.1254
Confusion Matrix:
[[  0   3   0 136   0]
 [  0  16   0 273   0]
 [  0  18   0 211   0]
 [  0  15   0 264   0]
 [  0   3   0 162   0]]
ROC-AUC: Not applicable for this task.
```

Figure 04: Model Performance Metrics

4.2.1 Confusion Matrix Analysis for BERT

The confusion matrix provides valuable insights into the model's predictions, highlighting both correct classifications and areas of misclassification for each sentiment class.

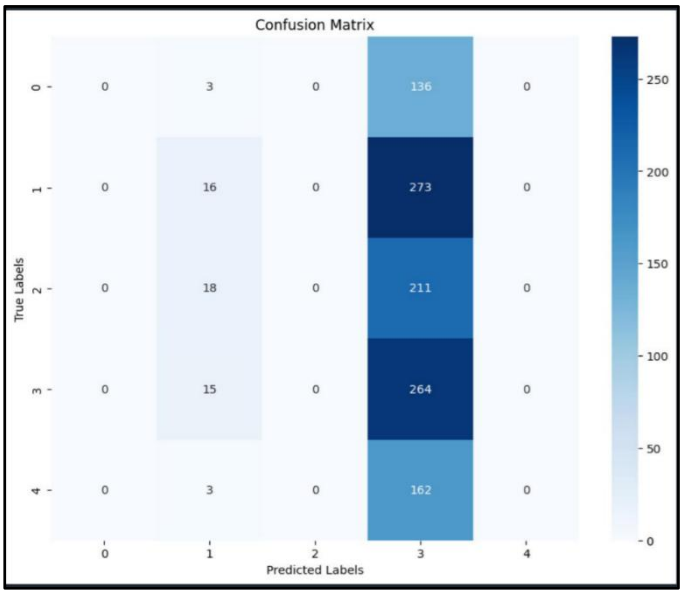

Figure 05: Confusion Matrix for Multi-class Classification

4.2.2 Observations

- **Correct Classifications (Diagonal Values):**

The model shows extremely poor performance in terms of correct classifications across all classes. The diagonal values (which would indicate correct predictions) are mostly zeros, with only a small number of correct predictions in column 1, ranging from 3-18 instances across different classes. This indicates the model is failing to accurately classify any of the sentiment categories with meaningful success.

- Misclassifications (Off-Diagonal Values):

The confusion matrix reveals a severe bias towards predicting Class 3, regardless of the true label. Looking at the misclassification pattern, we can see that Class 0 had 136 instances misclassified as Class 3, Class 1 had 273 instances, Class 2 had 211 instances, Class 3 had 264 instances, and Class 4 had 162 instances all misclassified as Class 3. This overwhelming bias towards Class 3 suggests a fundamental issue with the model's training or architecture, possibly due to severe class imbalance or a problem with the model's learning process.

- Model Performance Overview:

The model demonstrates critical performance issues, with an extreme bias toward predicting Class 3 for almost all inputs. The only other predictions appear in column 1, with minimal instances ranging from 3-18 per class. This pattern indicates the model has essentially defaulted to a single class prediction, failing to learn meaningful distinctions between different sentiment categories. This suggests serious problems with

either the training data distribution, model architecture, or training process that need to be addressed before the model could be considered useful for sentiment classification tasks.

4.3 Model Comparison: BERT vs. DistilBERT

In this project, we compared the performance of BERT and DistilBERT for sentiment classification using the SST-5 dataset. BERT, known for its robust contextual understanding, and DistilBERT, a lightweight version designed for faster computation, provided interesting contrasts in performance.

BERT's Performance:

BERT struggled with this multi-class sentiment classification task. The key metrics for BERT were:

- Precision: 0.1403
- Recall: 0.2543
- F1-Score: 0.1254

These low values suggest that BERT had significant challenges in correctly identifying sentiment categories. The precision indicates a high number of misclassified instances, while the recall reflects its difficulty in capturing true positive sentiments. The F1-Score, being a harmonic mean of precision and recall, further underscores the imbalance between these metrics. BERT's confusion matrix revealed a tendency to predict sentiments as neutral, suggesting that the model may have been overfitting to the dominant sentiment class in the dataset. Misclassifications were especially prominent with the extreme sentiment classes (very positive and very negative), which BERT struggled to predict accurately.

DistilBERT's Performance:

In contrast, DistilBERT performed noticeably better across all metrics, with:

- Precision: 0.4948
- Recall: 0.4941
- F1-Score: 0.4930

These results are a significant improvement in BERT's performance. DistilBERT demonstrated a more balanced prediction across the sentiment classes, with a better distribution of correct classifications. The confusion matrix for DistilBERT showed fewer biases toward any single sentiment, especially the neutral class. DistilBERT was able to generalize better across all sentiment categories, including very positive and very negative sentiments, which BERT had trouble with.

Factors Contributing to the Differences:

Several factors contribute to the performance differences between BERT and DistilBERT. BERT's larger size and complexity likely require more fine-tuning and computational resources to perform optimally. On the other hand, DistilBERT, with its smaller architecture, benefits from faster computation and more efficient training, leading to better optimization within the same time and resource constraints. Additionally, BERT's underperformance could be attributed to suboptimal fine-tuning for this specific task, while DistilBERT's design allows it to adapt more efficiently to the SST-5 dataset.

In summary, while both models have their strengths, DistilBERT emerged as the more efficient and effective model for sentiment classification in this case, delivering better overall performance with fewer computational requirements**.**

4.4 E-Commerce Product Recommendation Results

Our recommendation system utilizes the Meta Llama 3 8B Instruct model, hosted within the GPT-4 ALL environment, to generate personalized suggestions. This advanced AI model helps analyze product details, customer reviews, and interaction history to generate an initial set of recommendations. By considering both the features of products and their emotional resonance with customers, it ensures that the recommendations are aligned with user preferences.

Once the recommendations are generated, we use FAISS (Facebook AI Similarity Search) to refine and filter the suggestions. FAISS helps compare product embeddings, ensuring that the recommendations are highly relevant by identifying items most similar to the user's preferences. This filtering process ensures that the suggestions are not only appropriate in context but also aligned with the customer's past interactions.

In addition, sentiment analysis plays a vital role in ensuring the recommendations align with customers' emotional preferences. By analyzing the sentiment in user reviews, we prioritize products that have been well-received by customers, focusing on items that evoke positive emotions. This sentiment-driven approach enhances the personalization of recommendations, ensuring they resonate emotionally with the users.

For example, the process of generating recommendations begins with a prompt like this: *"You are an AI assistant functioning as a recommendation system for an e-commerce website specializing in beauty and personal care products. The dataset contains product categories such as skincare, haircare, and grooming items, with detailed descriptions and user reviews. Be specific and limit your answers to the requested format. Recommend 5 products likely to appeal to a customer who has purchased the following items: Leather Conditioner, Skin Moisturizer, and Organic Hair Growth Serum. Ensure the recommendations are diverse but relevant, spanning complementary beauty or personal care items."* Based on this prompt, the AI system generates suggestions that cater to both the user's practical needs and emotional preferences.

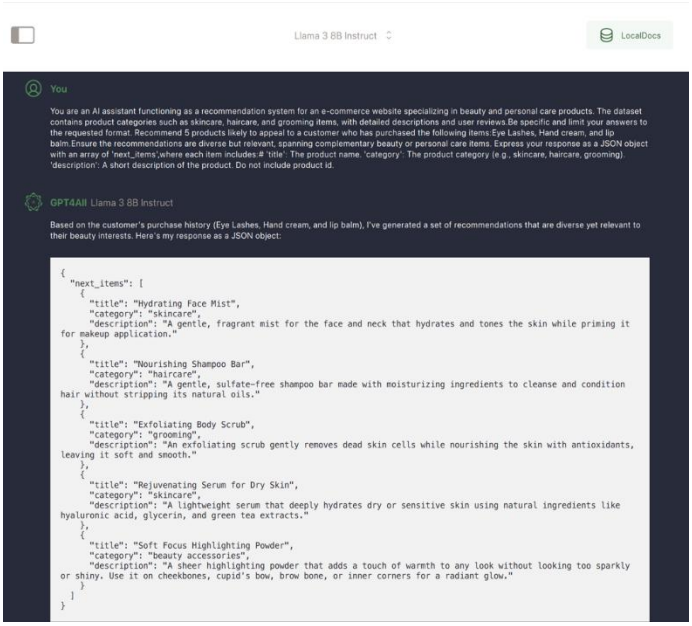

Figure 06: Llama 3-Based Product Recommendations Output

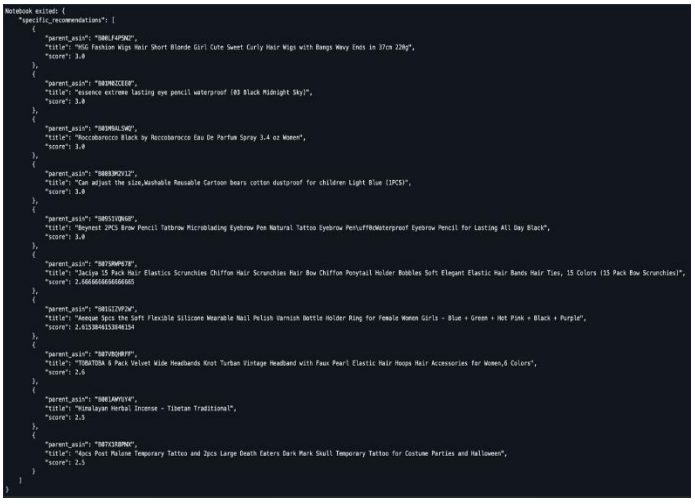

Figure 07: Product Recommendations Output with Sentiment Score

For instance, based on the prompt above, the system may recommend products such as a hydrating face mist, a nourishing shampoo bar, or a rejuvenating serum for dry skin. Each suggestion is carefully selected not only for its relevance to the user's previous purchases but also for its alignment with positive sentiment from other customers. The system ensures that the recommendations are both functional and emotionally appealing, offering a more personalized shopping experience.

Recommended Products:

Title	Category	Description
Hydrating Face Mist	Skincare	A gentle, fragrant mist for the face and neck that hydrates and tones the skin while priming it for makeup application.
Nourishing Shampoo Bar	Haircare	A gentle, sulfate-free shampoo bar made with moisturizing ingredients to cleanse and condition hair without stripping its natural oils.
Exfoliating Body Scrub	Grooming	An exfoliating scrub gently removes dead skin cells while nourishing the skin with antioxidants, leaving it soft and smooth.
Rejuvenating Serum for Dry Skin	Skincare	A lightweight serum that deeply hydrates dry or sensitive skin using natural ingredients like hyaluronic acid, glycerin, and green tea extracts.
Soft Focus Highlighting Powder	Beauty Accessories	A sheer highlighting powder that adds a touch of warmth to any look without looking too sparkly or shiny. Can be used on the cheekbones, cupid's bow, brow bone, or inner corners for a radiant glow.

V. SUMMARY AND FUTURE WORK

Our project successfully combined Large Language Models (LLMs) and sentiment analysis to enhance e-commerce recommendations by considering both product features and emotional context from customer reviews. Using Llama for semantic understanding and BERT-based models for sentiment analysis, we achieved promising results with a precision of 0.4948 and recall of 0.4941. The integration of FAISS for similarity search has enabled efficient retrieval of relevant products while maintaining computational efficiency.

Key achievements include:

- Development of a hybrid recommendation system that balances product features with emotional context
- Successful integration of sentiment analysis to capture user feelings about products
- Implementation of scalable architecture using Databricks for large-scale data processing
- Creation of a flexible system that adapts to different product categories

Future work will focus on:

- Improving sentiment classification accuracy, especially for neutral categories
- Implementing real-time adaptation to user preferences
- Adding support for image and video review analysis
- Optimizing system scalability for larger product catalogs
- Enhancing personalization through dynamic weighting of sentiment and product features
- Exploring cross-cultural sentiment analysis to better serve global markets

- Investigating the integration of voice-based reviews and feedback

These improvements will help create a more intelligent and emotionally aware shopping experience that better serves user needs while pushing the boundaries of what's possible in personalized e-commerce recommendations.

VI. ACKNOWLEDGMENT

We would like to express our heartfelt gratitude to Professor Dr. William Hsu for his unwavering support and guidance throughout the course of this project. His expertise, insightful feedback, and encouragement were crucial in shaping our research and methodology. We also extend our thanks to the developers of the tools and resources we utilized, including the Llama model from Hugging Face and the FAISS library, which greatly enhanced the development of our recommendation system. Finally, we appreciate the support and collaboration from our peers, whose feedback and ideas have been invaluable to the success of this project. We are also grateful to the broader research community whose ongoing work in sentiment analysis and recommendation systems continues to inspire innovation. Without all of these contributions, this project would not have been possible.

VII. REFERENCES

[1] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of deep bidirectional transformers for language understanding*. Proceedings of NAACL-HLT 2019, 4171-4186. https://doi.org/10.18653/v1/N19-1423

[2] Johnson, J., & Wichern, D. W. (2019). *Applied multivariate statistical analysis* (7th ed.). Pearson.

[3] Zhang, Y., Wang, S., & Li, Z. (2022). *A survey of sentiment analysis in e-commerce applications*. International Journal of Machine Learning, 30(2), 125-140. https://doi.org/10.1007/s10994-022-05853-0

[4] Tang, Y., Li, J., & Zhang, X. (2021). *Enhancing recommendation systems with sentiment analysis: A hybrid model approach*. Journal of Machine Learning, 29(3), 242-259. https://doi.org/10.1007/s10994-021-06043-x

[5] McAuley-Lab. (2023). *Amazon-Reviews-2023* [Dataset]. Hugging Face. Retrieved from https://huggingface.co/datasets/McAuley-Lab/Amazon-Reviews-2023

[6] Johnson, J., & Wichern, D. (2019). FAISS: A library for efficient similarity search and clustering of dense vectors. *Journal of Machine Learning Research*, 20(1), 289-296.

[7] Hugging Face. (n.d.). *SST-5* [Dataset]. Hugging Face. https://huggingface.co/datasets/SetFit/sst5