**Paulina Thrasher**

**Data Engineering Assignment 2**

**27/03/2025**

**Introduction**

Understanding how products relate to each other through the lens of customer reviews is a useful application of Natural Language Processing. This project develops a system to compare products from Amazon based on reviews left by customers. The data is from the Amazon Reviews 2023 dataset which contains information on different products (Amazon Science, 2023). Developed by McAuley Lab, it hosts multiple files for different product categories (Electronics, fashion, beauty, books, etc.). Each file is large with millions of rows containing user reviews, item information, and links.

This type of work can be valuable in a business context for companies looking to improve their recommendation systems or investigating customer segmentation. For instance, Amazon could use this type of analysis to suggest related products that customers could potentially be interested in. The goal of this project is to compare products by generating vector-based representations of customer reviews called embeddings and determining the similarity between them.

**Methodology**

1. **Gather Reviews for Specific Products**

Two JSONL files were loaded first: Handmade\_Products.jsonl contains review text, rating, and product identifiers while meta\_Handmade\_Products.jsonl includes product metadata such as the product title and parent\_asin, which is an id given to group products under the same family. Only products with at least 20 reviews were included to ensure that each product had enough information to analyze. The first 10 parent\_asin’s that qualified under this condition were included. Additionally, review data was merged with metadata to associate each with its corresponding. Below is a screenshot showing the average rating and number of reviews for each product in this filtered dataset. The second screenshot also shows a distribution of ratings per product. Notably, each product has the majority of ratings in the 5/5 category which is an important factor to keep in mind and is discussed further in the results and limitations of this project.

A screenshot of a computer

AI-generated content may be incorrect.

A graph with different colored bars

AI-generated content may be incorrect.

1. **Create Product Embeddings**

A sentence embedding model from Hugging face “all-MiniLm-L6-v2” was used to convert each review into a 384-dimensional vector. This model can be used to capture semantic meaning or for clustering. It is typically utilized for projects that encode short sentences or paragraphs, making it appropriate for Amazon reviews (Reimers & Gurevych). For each of the 10 products selected, embeddings (numerical representations of text) were generated for all its reviews.

Once all review embeddings were created for each product, they were aggregated into a single product level embedding which determines how much each review contributes to the final representation of the product. For this project, three different aggregation methods were explored. The Simple Average technique ensures that all reviews contribute equally to the final product embedding. The Rating-Weighted Average specifies that reviews are weighted based on their star rating (1-5), which means that higher rated reviews have more influence. Finally, the Helpfulness-Weighted Average method uses reviews that were rated “helpful” and weights them more.

These methods led to slightly different results when comparing products. The rating-weighted method seemed to give more weight to positive reviews which sometimes changed how similar two products appeared to be when they shared a lot of enthusiastic reviews, even if they were different in their function. The helpfulness-weighted method emphasized reviews that were considered useful by others, which highlighted the more detailed reviews. More discussion of these results can be found below in the ‘Key Findings’ sub-section of this report.

Review length was not included in the final method, but it is an important factor to take into consideration as longer reviews could contain more details and could be more influential. In future work, it would be advisable to include review length into aggregation either as its own method or in combination with rating or helpfulness.

1. **Finding Similar Products**

To find the top three most similar products for each item, I used both cosine similarity and Euclidean distance. Euclidean distance is the distance between two points in Euclidean space. In other words, it is the length of the line segment that connects two points (GeeksforGeeks, 2021). Cosine similarity measures the similarity between two vectors by taking the cosine of the angle between them (GeeksforGeeks, 2025).

In the code, I created functions called find\_similar\_products and find\_similar\_products\_euclidean with similar structures. First, the function takes the list of product ids from the embedding dictionary that was created previously. Next, the embedding dictionary is converted into a 2d matrix of shape (num\_products, 384), this is the dimensionality of the model all-MiniLM-L6-v2’s output (Reimers & Gurevych). Then, it can be passed through the cosine\_similarity or Euclidean\_distance functions and subsequently a similarity matrix and distance matrix are calculated. These matrices hold the cosine similarity and Euclidean distance values when comparing product i and j. For each product, a for loop is then used to get the name of the product and then get its row from the similarity/distance matrix which tells us how far it is from all other products. These calculations are sorted in increasing order and the indices of the 3 closest products is pulled, excluding the anchor product itself. Finally, this is run for each aggregation method (simple average, rating-weighted, and helpfulness-weighted).

**Key Findings**

As mentioned above, the results for this project evaluate 3 different aggregation methods (simple average, rating-weighted, and helpfulness-weighted) and 2 different measures of similarity (cosine similarity and Euclidean distance). The aggregation methods were used to weight reviews in using different techniques and the similarity concepts were used to group similar products based on these reviews. Overall, the three aggregation methods captured logical groupings such as personalized accessories or home decorations. The rating-weighted method tends to emphasize the voice of happy customers, leading to stronger clustering among items that were rated highly. As noted earlier, most reviews fell in the 5/5 rating category for each item. However, the results showed that products with moderate or even lower than average ratings still appeared highly similar. For example, it more clearly aligned the 16k Gold Name Bar Bracelet (avg. rating 3.45) and the Baby Name ID Bracelet (avg rating 3.55), suggesting that similarity scores were driven by consistent language or words used, which makes sense given that they are both bracelets. The rating-weighted average likely just emphasized similarities between products that already had overlapping themes.

Products similar to:

16K Gold Your Name Bar Bracelet - Personalized gift Gold Plated bar Delicate Hand Stamp Best bridesmaid Wedding Graduation Gift

→ Baby Name Bar id Bracelet Baby Gift Personalized gift 16k Gold Plated Dainty Hand Stamp Your Baby Name Customized New Born to Children First Birthday Great Gift: similarity = 0.9610

→ Personalized Customizable Necklace Women Mothers Day Gift for Mom Silver Name Bar Custom Engraved Gold Necklace Bridesmaid Engagement Gift for Her Bestie Jewelry for Girls Unique New Mom Gifts - 4N: similarity = 0.9110

→ Triple Protection Bracelet - For Protection - Bring Luck And Prosperity - Hematite - Black Obsidian - Tiger Eye - Stone Bracelet: similarity = 0.9001

On the other hand, the helpfulness-weighted method gave more influence to longer, more detailed reviews. These reviews were often marked as ‘useful’ by other users which sometimes resulted in more unique product groupings based on user’s experiences.

In the results, there were a few pairings where the mathematical similarity that the model found did not match real-world expectations. For instance, one product was the Benshot Bullet Rocks Glass (a glass drinking cup for cocktails) that was matched with custom wall art and a custom necklace. These items have very different functions but shared keywords like “unique”, “gift”, and “family” which likely led to similar language used in the reviews, skewing the embeddings. The following results were produced when using cosine similarity and a simple average. The results were consistent across aggregation methods and distance calculations, the following output is used as an example.

Products similar to:

The Original BenShot Bullet Rocks Glass with Real .308 Bullet - 11oz | Made in the USA

→ Custom Metal Signs | Metal Name Sign | Metal Wall Art | Split Letter Monogram Wall Decor | Metal Wall Art Last Name Sign | Family Name Sign | Personalized Wedding Gift | Metal Art | Outdoor Metal Sign: similarity = 0.8326

→ Custom Star Map - Personalized Star Map (Multiple Sizes - Unframed Star Prints, Star Constellation Map Wall Art, Great Gift - Special Occasion, Engagement Gift, Wedding Gift, Anniversary Gift): similarity = 0.7971

→ Personalized Customizable Necklace Women Mothers Day Gift for Mom Silver Name Bar Custom Engraved Gold Necklace Bridesmaid Engagement Gift for Her Bestie Jewelry for Girls Unique New Mom Gifts - 4N: similarity = 0.7743

Looking at cosine similarity and Euclidean distance, there were also some differences. Cosine similarity is sensitive to the direction of the embedding vectors, which makes it good for comparing the overall tone of reviews (GeeksforGeeks, 2025). In contrast, Euclidean distance considers direction and magnitude (GeeksforGeeks, 2021) which can cause different results for products with shorter vs longer reviews.

To show a case where we can see magnitude-sensitive differences when looking at results, I created some code to show the average review length between products. The AirPod case had an average review length of 21.4 words, while the products like the Custom Star Map and the Wooden name sign had an average length of 37.4 and 30 words respectively. Despite this difference, the AirPod case was paired with those items when Euclidean distance was used. For example, with helpfulness-weighted Euclidean distance, the AirPod Case’s top 3 similar products were the Custom Metal Sign, personalized Necklace, and Star Map which are different in their respective uses, but the results indicate that they are similar in the embedding space. Using cosine similarity showed slightly different results. The AirPod Case was not included in the top 3 similarities for any of those products. This could be an example that shows how cosine similarity was more sensitive to the overall tone of the reviews whereas Euclidean distance allowed repetitive and shorter reviews to appear closer to longer reviews that had similar words. This shows how the aggregation process and similarity metric can have a significant effect on how the relationship between products is interpreted.

**Visualizations**

Taking inspiration from the examples provided, the visuals that were created for this report were 3 different t-SNE 2d Embedding visualization. It is a 2d scatterplot that shows product embeddings and can be used to show how similar products cluster together. Below is an example, showing the relationships between the sample of products based on helpfulness-weighted reviews. T-SNE is a dimensionality reduction technique that takes high-dimensionality embeddings and projects them into 2d space, it has its own pairwise distance approach to show relationships between data points (DataCamp). The reduction in dimensionality done by this method can change how relationships between products are interpretted. Products that seem far away on the 2d plot can actually be relatively close in the original high-dimensional space and vice versa. This means that this visual should not necessarily be used as a precise tool to measure similarity, but more of an overall view of some patterns.

In the below plot, we can see that the Triple Protection Bracelet and 16K Gold Name Bracelet are clustered together which shows a strong overlap in the theme, tone, and words used in reviews. Both products had reviews that talked about the aesthetic of the product, gifting it to friends/family, and personal touches. For example, the reviews for the Triple Protection bracelet included phrases like “Love my new bracelet!” and” Loved it , wll buy more for gifts” while the 16k Gold bracelet had reviews like “This was a gift and the recipient was so pleased.” and “This bracelet was a early valentines gift from my husband and it was perfect!”. Reviews like this likely caused the embeddings to look similar in the reduced dimensionality space of the scatterplot.

One outlier that seems to appear is the Original Benshot Bullet Rocks Glass. The reviews seemed to be slightly more niche compared to the others. While they did mention similar keywords to othe reviews, reviews like “Bought this for a gun owner friend of mine as a thank you gift - he LOVED it!” and “Quality glass. Great addition for someone's Man Cave.” Showed the unique characteristics of this product as it appeals to men, gun culture, and is more of a novelty gift.

A graph with blue dots

AI-generated content may be incorrect.

**Limitations and Possible Improvements**

There are some limitations that are important to mention regarding the embedding based approach used in this project. Firstly, I limited the analysis to products with more than 20 reviews. While this gives more data to work with, it also may not reflect the broader dataset as some products could have less reviews. Additionally, there are more complex and advanced aggregation methods that could be used to improve the product embeddings. For example, attention-based pooling could improve this analysis because it assigns more weight to more informative parts of reviews rather than assuming all reviews contribute equally or using weights like rating or helpfulness (Mukherjee, 2025).

When analyzing rating distributions, I found that most reviews across all products were five-star ratings. This pattern likely created a sentiment-skew that probably affected the review embeddings, particularly under the rating-weighted aggregation. Handmade and personalized products likely were used as gifts which often means that buyers rate based on how excited the recipient was, or they feel more obligated to rate the items of smaller sellers more highly. As a result, it’s important to call out that methods like rating-weighted aggregation may have reinforced already strong similarities in products with overlapping themes even when actual feedback was less varied.

In terms of visualizations, as discussed, the t-SNE visualization can distort some relationships, so it would be interesting to include other visuals like a heatmap or a bar chart illustrating review length could provide more context to the analysis. Another possible addition that could be included is incorporating more of the data on price, category, or star rating alongside review text to provide even greater context.

**Citations**

Amazon Science. (2023). Amazon Reviews 2023 Dataset [Data set]. Hugging Face. <https://huggingface.co/datasets/amazon/amazon-reviews-2023>

DataCamp. (n.d.). Introduction to t-SNE in Python: A Tutorial. <https://www.datacamp.com/tutorial/introduction-t-sne>

GeeksforGeeks. (2021, July 14). Euclidean distance. GeeksforGeeks. <https://www.geeksforgeeks.org/euclidean-distance/>

GeeksforGeeks. (2025, March 1). Cosine Similarity. GeeksforGeeks. <https://www.geeksforgeeks.org/cosine-similarity/>

Mukherjee, Tanmoy. (2025, March 26).  *Understanding Attention Mechanisms* [PowerPoint slides]. Faculty of Business Economics, University of Antwerp. <https://lms.uantwerpen.be/ultra/courses/_111486_1/outline/file/_6133967_1>

Reimers, N., & Gurevych, I. (n.d.). *all-MiniLM-L6-v2* [Pretrained model]. Hugging Face. <https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2>