

Iteration 04

Jinyu Li, Richard Chen, Yichen Zhang

November 2025

1 Dataset Description

- Link to dataset <https://www.kaggle.com/datasets/karkavelrajaj/amazon-sales-dataset>
- Introduction to dataset

The Amazon Sales Dataset is a publicly available CSV dataset on Kaggle. Its author is KARKAVELRAJA J, and it was updated 3 years ago. This dataset contains more than 1000 Amazon products' ratings and product reviews with details listed on the official website of Amazon. The dataset includes 16 fields: product, product_name, category, discounted_price, actual_price, discount_percentage, rating, rating_count, about_product, user_id, user_name, review_id, review_title, review_content, img_link, and product_link. Some of them are in numerical form, and others are in textual form. In our project, we are making a chatbot that automatically retrieves and organizes user reviews and ratings from this dataset.

- Explanation Our team aims to build a chatbot that retrieves, organizes, and analyzes user reviews through automated backend pipelines, SQL query handling, and a web-based interface. This dataset is highly suitable for our project because it provides well-structured data. It contains 1K+ product records, each paired with reviews, ratings, product metadata, and also user identifier. Additionally, the dataset reflects real consumer review behavior from a platform similar to the environment our chatbot is designed to interact with, making it ideal for building, testing, and demonstrating the end-to-end functionality described in our project scope.

2 Tools and Methodologies

- Python
 1. Used for data ingestion, cleaning, and preprocessing.
 2. Chosen for its strong libraries (pandas, numpy) and ability to efficiently handle both tabular data and text.
- SQL
 1. Used for structured storage of products, users, and reviews.
 2. Supports reliable querying, indexing, and aggregation essential for chatbot retrieval.
 3. Because relational databases provide strong consistency, reliable indexing, and efficient retrieval of review information, which are essential for chatbot responses.
- React
 1. Used to build the interactive web UI.
 2. Component-based design and strong ecosystem allow fast, scalable frontend development.

3 Preliminary Timeline

See Figure 1

Task ID	Task Description	Assigned to	Status of Completion	Due
1.1	Define topic & objectives	All	100%	9/26/2025
1.2	Define roles	All	100%	10/26/2025
1.3	Confirm tools & environment	All	100%	
2.1	Confirm data source license & citation	All	100%	10/30/2025
2.2	Download dataset and organize	All	100%	10/30/2025
2.3	Review data briefly	All	100%	10/30/2025
3.1	Remove duplicates	Jinyu & Richard	100%	11/2/2025
3.2	Standardize data types	Jinyu & Richard	100%	11/2/2025
3.3	Text normalization	Jinyu & Richard	100%	11/2/2025
3.4	Generate cleaned dataset	Jinyu & Richard	100%	11/2/2025
4.1	Draw ER diagram	All	100%	11/8/2025
4.2	Design schema	All	100%	11/8/2025
4.3	Constraint & index design	All	100%	11/8/2025
5.1	Create database	Jinyu		11/14/2025
5.2	Load data into SQL schema	Jinyu		11/14/2025
5.3	Validate referential integrity	Jinyu		11/14/2025
6.1	Define 8 analysis questions	Jinyu & Yichen		11/21/2025
6.2	Implement queries #1-4	Jinyu & Yichen		11/22/2025
6.3	Implement queries #5-8	Jinyu & Yichen		11/22/2025
6.4	Check performance	Jinyu & Yichen		11/22/2025
7.1	Init backend data pipelines	Richard		11/25/2025
7.2	DB connection	Richard		11/25/2025
7.3	Build endpoints for all SQL queries	Richard		11/25/2025
7.4	API unit & integration tests	Richard		11/25/2025
8.1	Front-end and UI design	Yichen		11/28/2025
8.2	Set up UI style & components	Yichen		11/28/2025
8.3	Build Overview dashboard	Yichen		11/28/2025
8.4	API integration & error handling	Yichen		11/28/2025
9.1	Define chatbot capabilities	Richard		11/30/2025
9.2	Intent routing & backend logic	Richard		11/30/2025
9.3	Front-end chat UI	Yichen		11/30/2025
10	Final Report	All		12/1/2025

Figure 1: Timeline

4 Team Member Contributions

- **Jinyu Li** Leads data preprocessing and SQL design. Cleaned the dataset, created schema and analytical queries, and coordinated overall progress tracking. Will continue managing database optimization and reporting.
- **Yichen Zhang** Responsible for front-end development. Designed the interface layout, prepared React components, and planned database integration through REST API. Next, will connect API endpoints and deploy the web app.
- **Richard Chen** Built the Python backend and planned chatbot logic. Handled data ingestion scripts and tested Flask connectivity. Will implement sentiment analysis and integrate chatbot functions into the deployed system.
- **Team Collaboration** The team collaborates weekly through GitHub commits and progress-tracking spreadsheets. Tasks are clearly divided but reviewed jointly. In the next phase, roles will overlap more during integration, debugging, and presentation to ensure end-to-end system consistency.

5 Progress and Next Steps

Current Progress: So far, our team has completed the dataset review and is in the process of finalizing the required columns while mapping the Excel fields to the corresponding tables and columns in the SQL database. We are also developing a Python automation pipeline to clean the datasets and load them into the SQL database. Overall, the project is progressing smoothly with no major challenges encountered. However, we are taking extra care to ensure that the pipeline correctly categorizes the data and that our SQL queries retrieve the intended information accurately.

Next Steps: In the upcoming week, our team will focus on finalizing the Python automation pipeline and testing it with sample datasets to verify that data is being cleaned, transformed, and loaded into the SQL database accurately. We will also begin validating the integrity of the data within the database, ensuring that all mappings between Excel and SQL columns are consistent. Once testing is complete, we plan to document the full ETL workflow and make any necessary adjustments to improve efficiency or error handling. At this stage, no major changes to our original plan are anticipated, but minor refinements may be made based on the results of our initial test runs.