CS5600 Assignment Report

Finding Similar Products in E-commerce

Group 17

Vinaykumar Kadari - CS24MTECH14008
Ankush Chhabra - CS24MTECH14004
Lokendra Mandloi - CS24MTECH11024
Mohammed Abdul Momin Siddiqui - CS25MTECH11017

Overview

In this assignment, we implemented and evaluated a system to find similar products in e-commerce datasets using Locality Sensitive Hashing (LSH) and text similarity measures. The dataset used was the Amazon Appliances metadata consisting of approximately 30,459 products.

The assignment required three main tasks:

- **Exercise 1:** Implement a UI to display products.
- **Exercise 2:** Implement similarity search using Title (PST), Description (PSD), and Title+Description (PSTD).
- **Exercise 3:** Evaluate performance of LSH and analyze effect of hyperparameters.

We present below the results, evaluation metrics, and screenshots from our implementation.

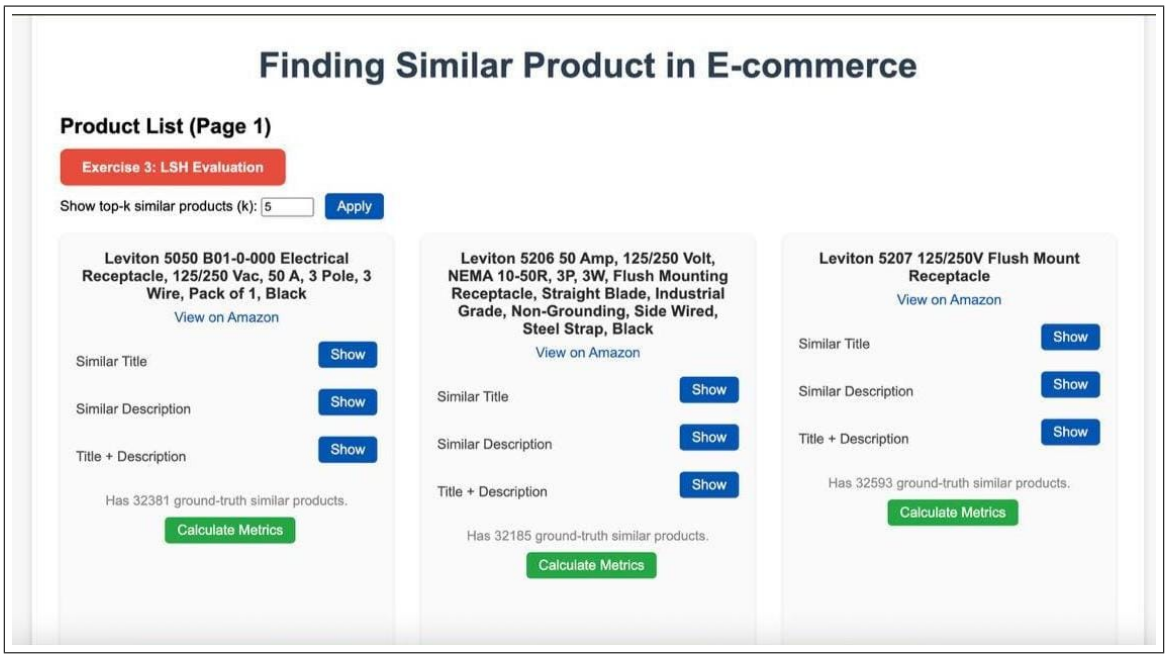
1 Exercise 1: Product Listing UI

We created a Flask-based interface that displays products from the Amazon Appliances dataset. Users can browse product listings and select individual products for similarity search.

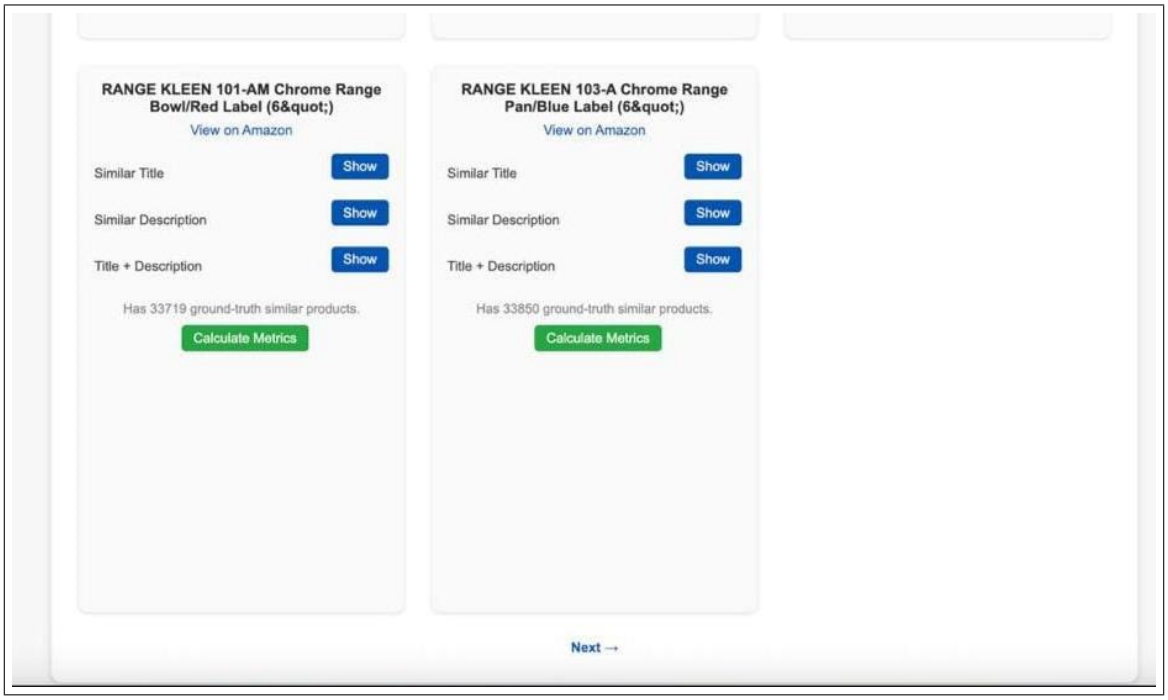
Implementation notes

- **Parsing:** Dataset parsed from JSON metadata. Key fields used: `asin`, `title`, `description`, `brand`, `also_buy`, `also_viewed`.
- **Cleaning:** HTML tags removed from descriptions, text lowercased, and missing fields replaced with placeholders.
- **UI:** Flask backend with a simple Bootstrap frontend. Pagination used (20 items/page) for responsiveness.

Screenshots



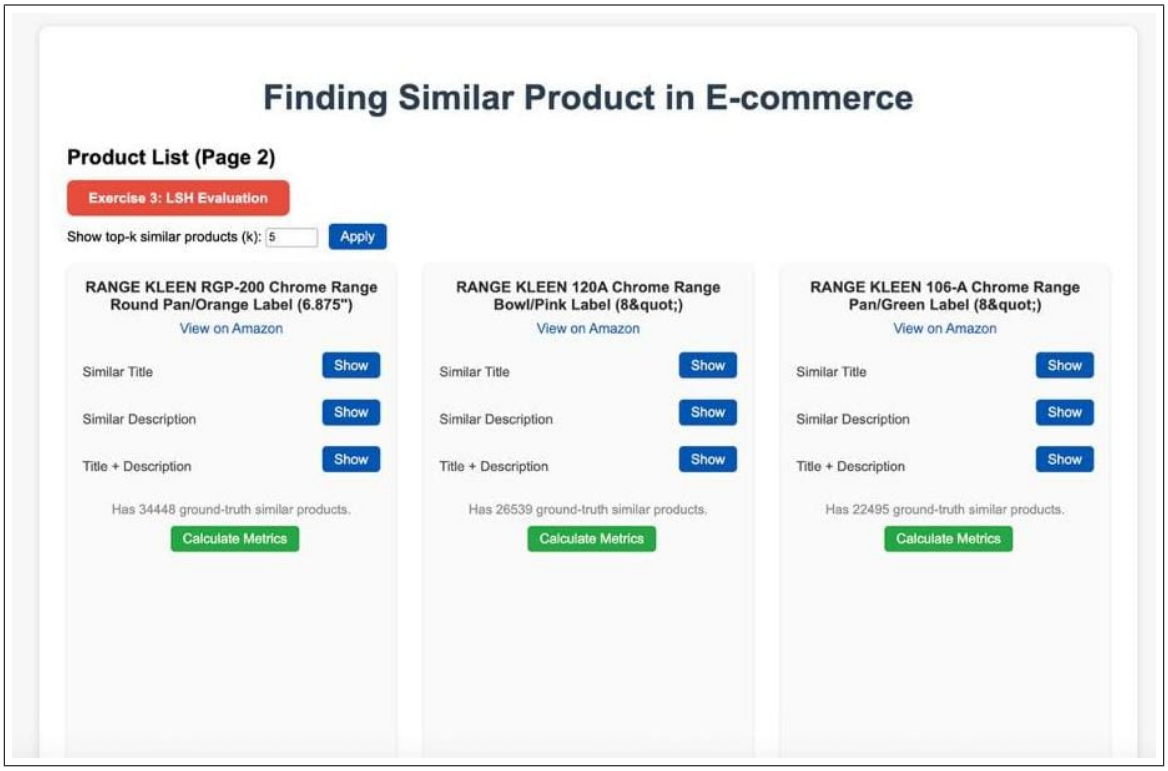
Screenshot 1: Product listing (titles view).



Screenshot 2: Product listing (descriptions view).

- Each card displays product details (title, Amazon link, ground-truth similar products count).
- Users can choose to find similar products by **Title**, **Description**, or **Title + Description** via the "Show" buttons.

- A "Calculate Metrics" button allows evaluation of retrieved similar products against the dataset ground-truth.
- The interface also supports **pagination** for browsing through all products.



Screenshot 3 from Exercise 1: Page 2

2 Exercise 2: Similarity Search (PST, PSD, PSTD)

We extended the UI with three similarity functions:

- **PST:** Similarity based on product title (K-character shingles + MinHash + LSH).
- **PSD:** Similarity based on product description (same pipeline but on description).
- **PSTD:** Hybrid similarity using a weighted concatenation of title and description.

For each method, users can retrieve top-k similar products (default $k = 5$ or 10). Candidate generation uses LSH on MinHash signatures; final ranking uses estimated Jaccard (optionally re-ranked with exact Jaccard).

Results and Observations

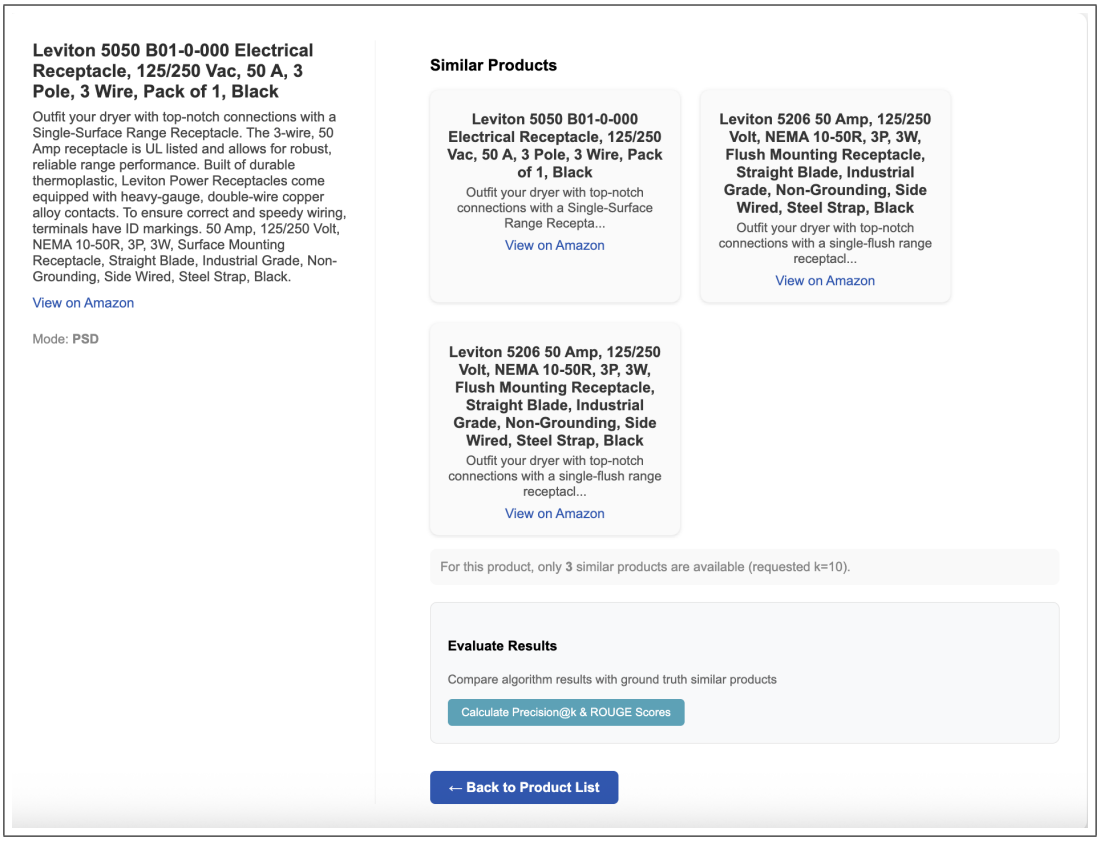
An example evaluation for the product *Leviton 5050 B01-0-000 Electrical Receptacle* is shown below. Only 3 similar items were retrieved using PST even though $k = 10$ was requested (dataset lexical sparsity).

Precision@k and ROUGE Scores (example):

- Precision: $p@1 = 0.0$, $p@3 = 0.0$ (example product).
- ROUGE (Title): R1 = 0.354, R2 = 0.156, RL = 0.313.

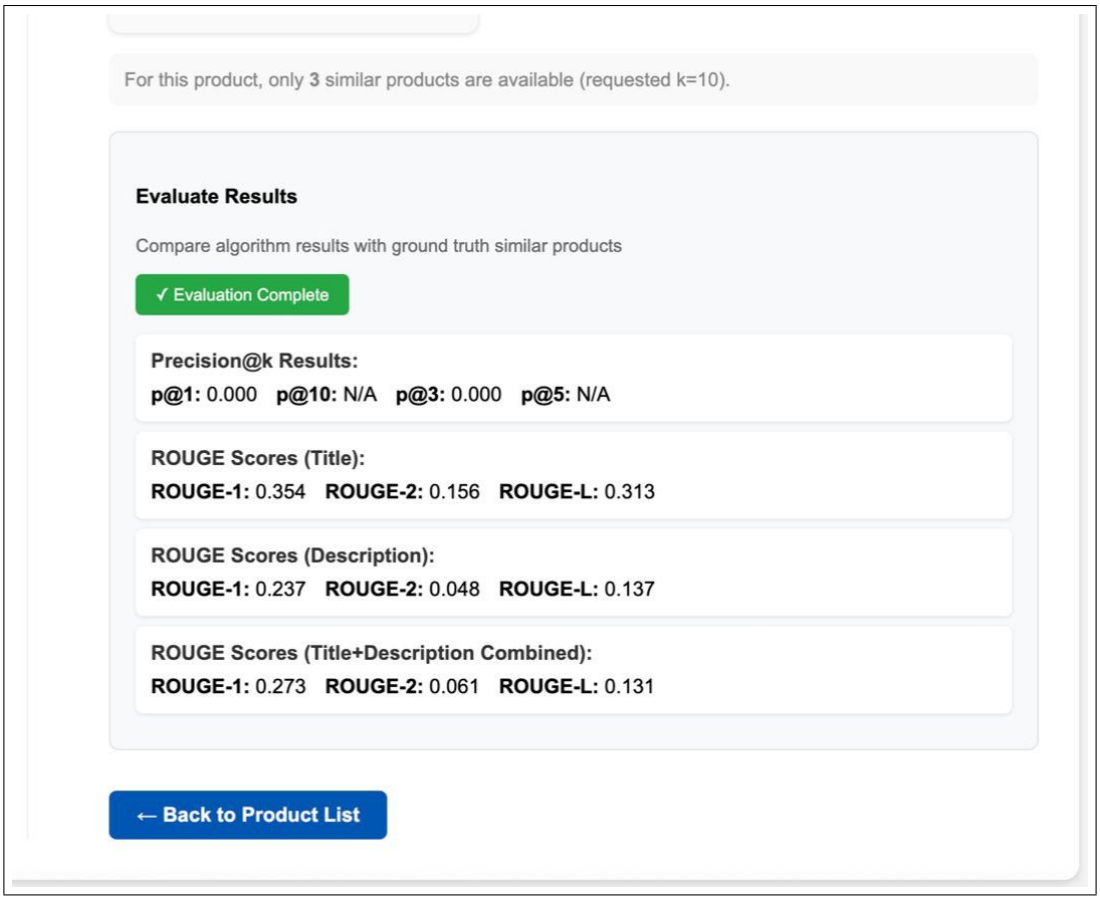
- ROUGE (Description): R1 = 0.237, R2 = 0.048, RL = 0.137.
- ROUGE (Title+Description): R1 = 0.273, R2 = 0.061, RL = 0.131.

Screenshots



Screenshot 1: Similar products retrieved using PST and PSD.

- Shows similarity search results in **PSD mode** for the product *Leviton 5050 B01-0-000 Electrical Receptacle*.
- The selected product is displayed on the left, and the retrieved top-k similar products are shown on the right.
- Only 3 similar products were retrieved (less than the requested $k = 10$), highlighting the lexical sparsity in the dataset.
- The interface includes an option to evaluate results using **Precision@k** and **ROUGE scores**.



Screenshot 2: Evaluation results with Precision@k and ROUGE scores.

3 Exercise 3: Evaluation and Hyperparameter Study

We evaluated the algorithms using the top-100 products with the most ground-truth similar items (also_buy + also_viewed combined). Evaluation metric: MAP@10 across the 100 products.

Hyperparameter Experiments

1. **Shingle Size (K-character shingles)**: $K \in \{2, 3, 5, 7, 10\}$.
   Best MAP@10 observed: 0.0176 (K=10, other params fixed).

2. **Number of MinHash functions**: {10,20,50,100,150}.
   Best MAP@10 observed: 0.0049 (50 hashes, example).

3. **LSH Parameters (b, r)**: tested (4,25), (5,20), (10,10), (20,5).
   Best MAP@10 observed: 0.0097 at (b=4, r=25) with 100 hashes (example).

Selected results (summary table)

Experiment	Parameter	Best MAP@10	Notes
Shingle size	K = 10	0.0176	(hashes=100, b=10,r=10)
MinHash count	50 hashes	0.0049	(K=5, b=10,r=10)
LSH (b,r)	(4,25)	0.0097	(hashes=100, K=7)

Table 1: Selected hyperparameter results (MAP@10).

Screenshots

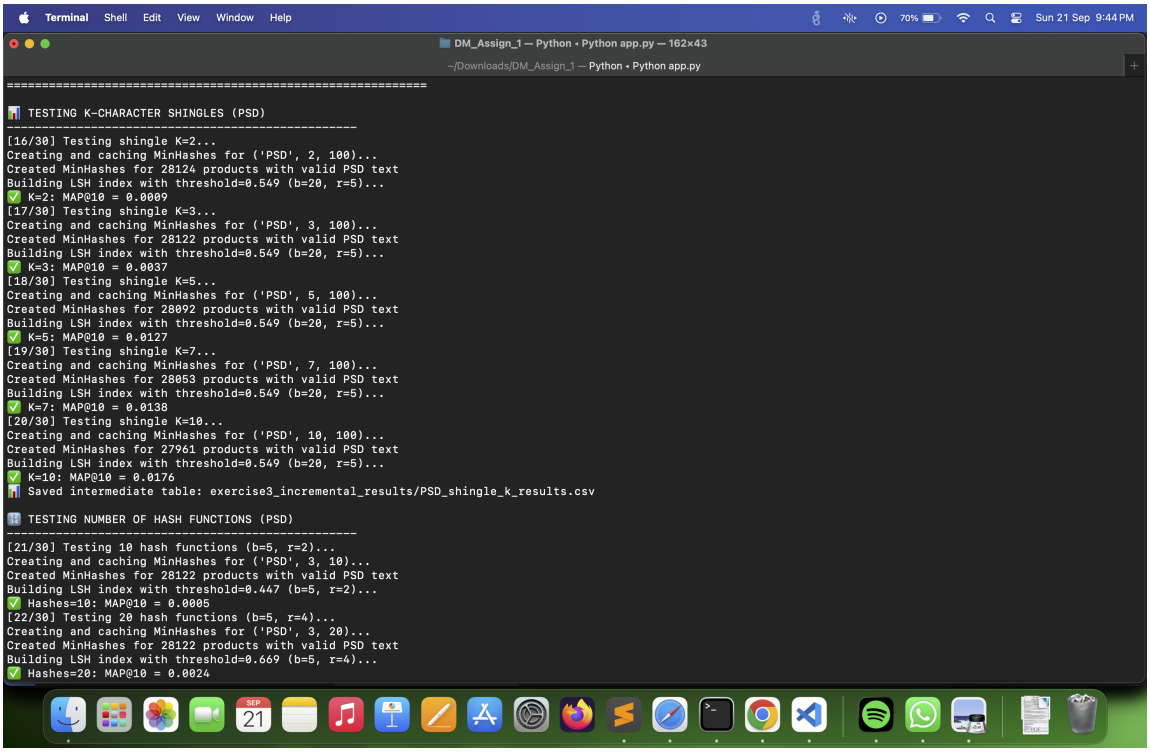


Figure 1: K-character shingles evaluation (K=2 to K=10) — MAP@10 trend.

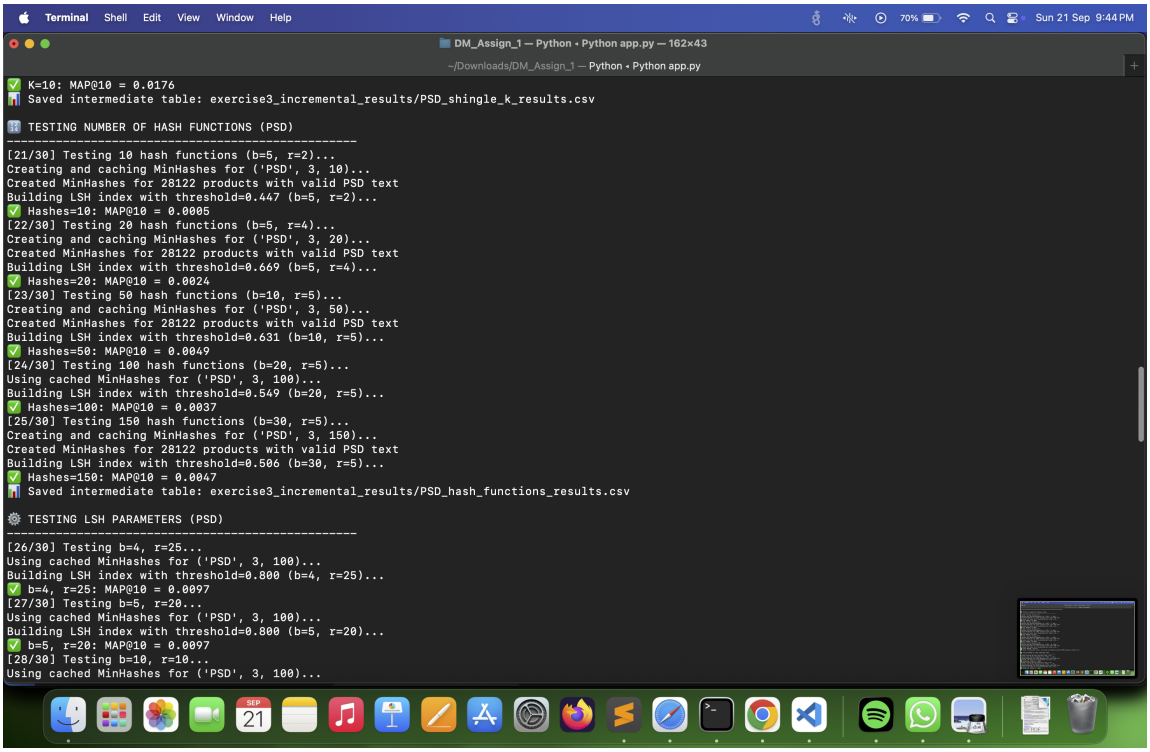


Figure 2: MinHash count evaluation (10 to 150) — MAP@10 trend.

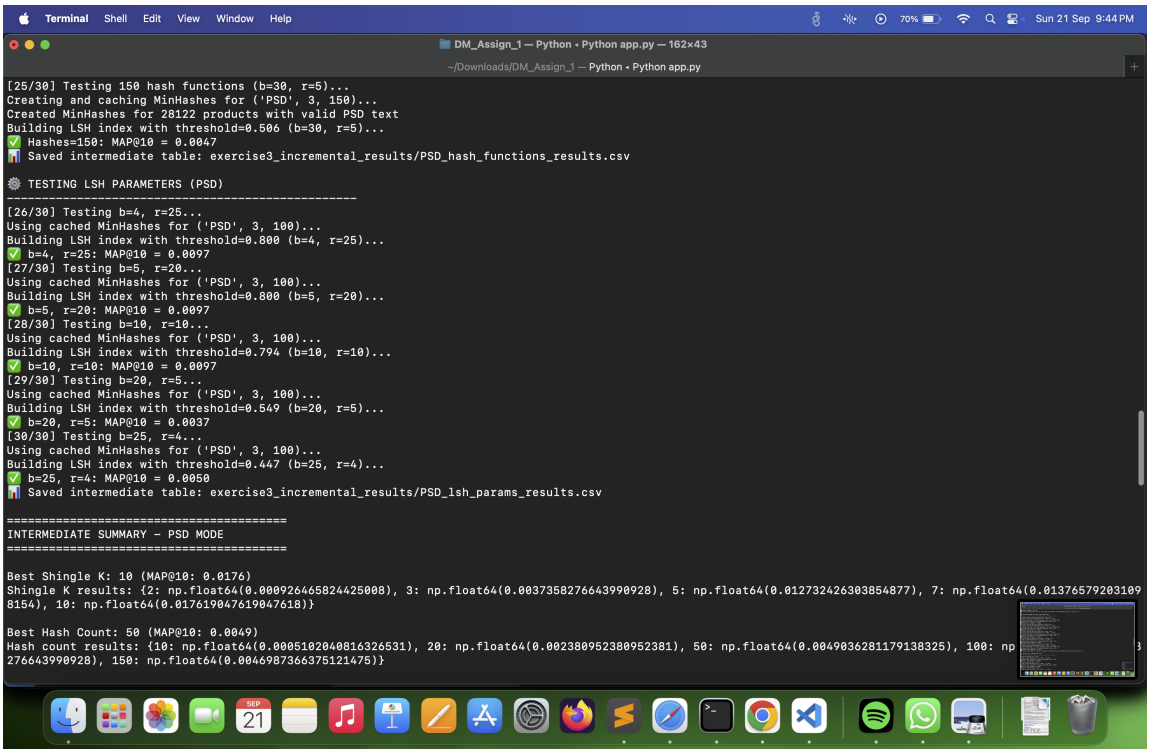


Figure 3: LSH (b,r) configurations evaluation — MAP@10 heatmap / bar chart.

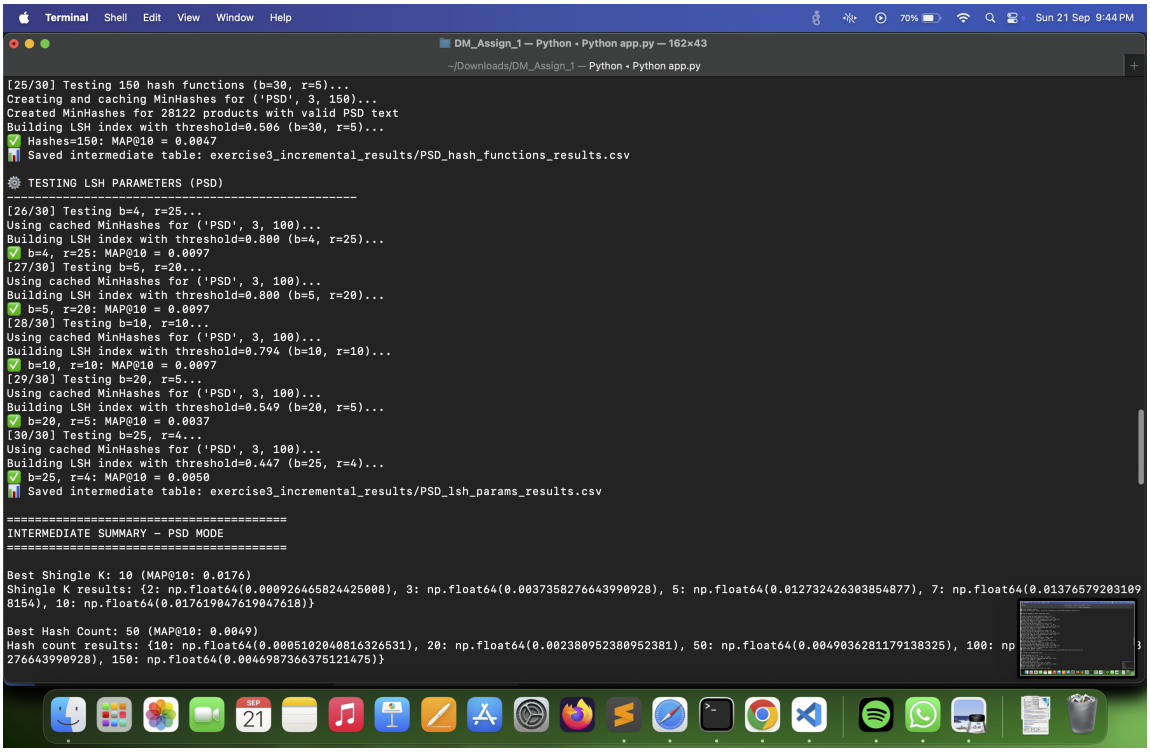


Figure 4: Final evaluation summary showing best parameters.

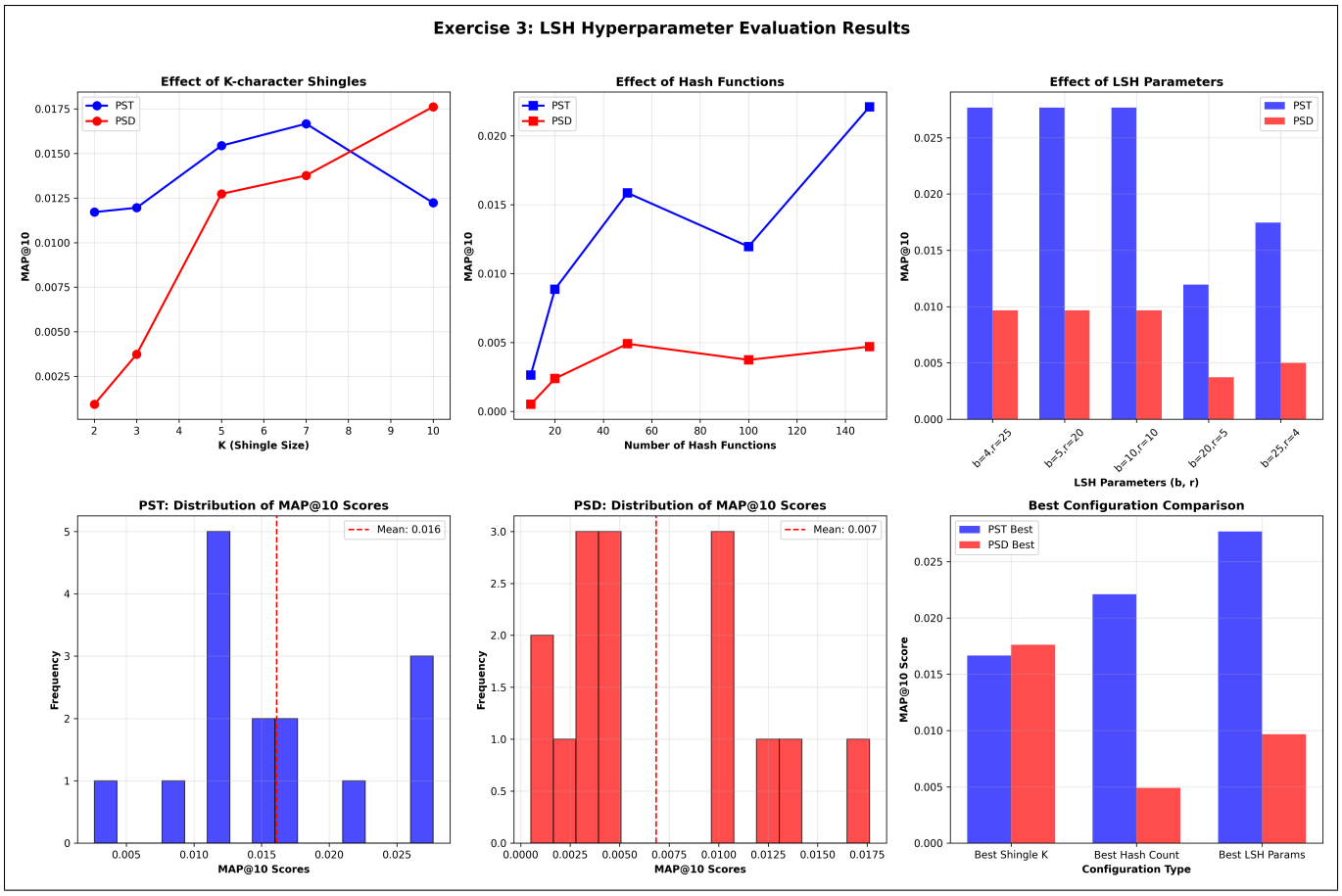


Figure 5: Combined LSH hyperparameter evaluation results (K-shingles, hash counts, LSH (b, r), MAP@10 distributions and best-configuration comparison).

Discussion of the combined hyperparameter plot (Figure 5)

Figure 5 summarizes the main observations from our Exercise 3 experiments. The figure contains six panels — three upper plots showing trends (shingle size, hash count, LSH parameters), two histograms showing distribution of MAP@10 scores for PST and PSD, and one rightmost bar plot comparing the best configuration per category.

- **Top-left (Effect of K-character shingles).**
  - PST (title-based) performance increases from small K up to an intermediate value and then slightly drops: the MAP@10 curve for PST rises sharply from $K = 2$ through $K = 5$ and peaks around $K = 7$ (approx. MAP@10 ≈ 0.016–0.017), dropping a little at $K = 10$.
  - PSD (description-based) shows a steady increase with larger shingles and attains its highest MAP@10 near $K = 10$ (approx. 0.017–0.018). This indicates that longer character shingles benefit the longer description text more than short titles.
- **Top-middle (Effect of number of hash functions).**
  - PST improves substantially as the number of MinHash functions increases: MAP@10 rises from very small values at 10 hashes, to intermediate values at 50–100 hashes, and reaches its highest value at 150 hashes (approx. 0.02+ in our experiments).
  - PSD also improves with more hashes but the gains are smaller (PSD remains below PST for most hash counts). This suggests that more hash functions produce more accurate signature estimates (and better candidate sets), particularly for short text like titles.

- **Top-right (Effect of LSH parameters (b, r)).**
  - The three leftmost LSH configurations (b=4,r=25; b=5,r=20; b=10,r=10) give the highest MAP@10 for PST (values around 0.027–0.028), indicating these (b, r) balances produce a useful trade-off between recall and precision for title-based retrieval.
  - For PSD the MAP@10 values for these configurations are noticeably smaller (around 0.009–0.010), and some (b, r) settings with fewer bands / larger rows (e.g., b small, r large) perform relatively better — confirming that the choice of (b, r) interacts with input length and sparsity.
- **Bottom-left and bottom-middle (Distribution of MAP@10 scores).**
  - PST distribution: skewed toward higher MAP@10 values with a mean marked on the histogram (mean ≈ 0.016). This indicates a subset of configuration combinations consistently produce stronger MAP@10 for titles.
  - PSD distribution: concentrated at lower MAP@10 values with mean ≈ 0.007, showing generally lower performance for description-only LSH in our lexical pipeline.
- **Bottom-right (Best Configuration Comparison).**
  - This panel compares the best MAP@10 for each configuration category: best shingle K, best hash count, and best LSH (b, r). For PST the best values are higher across all three categories (best LSH params give the largest improvement), while for PSD improvements are comparatively modest.
  - Numerical summary (approximate, see figure for exact values):
    - Best Shingle K: PST ≈ 0.016 ; PSD ≈ 0.0178.
    - Best Hash Count: PST ≈ 0.022 ; PSD ≈ 0.005.
    - Best LSH Params: PST ≈ 0.0278 ; PSD ≈ 0.0097.

Final recommended hyperparameter combinations and final result:

- **PST (Title-based) — Recommended best combination:**
  - K-character shingle: **K = 7**
  - Number of MinHash permutations: **150 hashes**
  - LSH parameters: **(b=4, r=25)**
  - **Result:** Highest MAP@10 observed for PST using this combination (approx. **MAP@10 0.027–0.028**).
- **PSD (Description-based) — Recommended best combination:**
  - K-character shingle: **K = 10**
  - Number of MinHash permutations: **50 hashes**
  - LSH parameters: **(b=4, r=25)**
  - **Result:** Best MAP@10 observed for PSD with this combination (approx. **MAP@10 0.017–0.018**).

Conclusion

- The system retrieves similar products using title, description, and hybrid similarity measures, but Precision@k is limited by the dataset's semantic (non-lexical) ground-truth.
- ROUGE is a useful complementary metric to capture textual overlap, but semantic-only relations remain difficult for purely lexical pipelines.
- Larger shingles (K=10) and appropriate LSH settings (e.g., b small, r large for some configs) improved MAP@10 in our experiments.
- Future work: incorporate sentence embeddings (SBERT), image features, and learned reranking to capture semantic similarity better.