

AI-Powered Merchandising Agent for SKINSEOUL

Case Study Submission

Chetan Sonawane

 [Chetansonawane7](#)

 [chetan-sonawane](#)

 sonawanechetan@gmail.com

This document outlines the design and logic for a fully automated merchandising system for SKINSEOUL. This AI-powered agent dynamically ranks products for key website touchpoints to boost conversion rates, increase Average Order Value (AOV), and enhance product discovery, while significantly reducing manual curation effort.

1. Strategic Touchpoint: Homepage "Best Sellers"

The primary touchpoint selected for automation is the **Homepage "Best Sellers" Carousel**.

1.1. Rationale

This carousel is the most strategic merchandising element due to its high visibility and its role as the first point of contact for most users. A dynamic, data-driven "Best Sellers" list immediately engages customers with proven, high-demand products, shaping a positive user journey from the first click and driving qualified traffic to high-conversion product pages.

1.2. Design Goals & Constraints

- **Goals:** Increase conversion, boost AOV, enhance product discovery.
- **Constraints:** Ensure product visibility is tied to sufficient inventory to avoid customer frustration. The system must be robust, reliable, and refresh on a regular cadence (e.g., every 24 hours).

2. Data-Driven Scoring Engine

The core of the agent is a weighted scoring function that balances multiple business objectives.

2.1. Data Inputs

A variety of data points from the provided mock dataset are used to calculate a product's relevance score.

Field	Purpose in Logic
Volume Sold Last Month	Primary indicator of sales velocity and popularity.
Views Last Month	Proxy for user interest and engagement.
Brand Tier (A, B, C)	Aligns merchandising with strategic brand partnerships.
Days of Inventory	Penalizes slow-moving products to optimize stock turn.
Units in Stock	Used as a hard filter to exclude out-of-stock items.

2.2. Scoring Function

The final score for each product is calculated using the following normalized and weighted formula:

$$\begin{aligned} \text{Score} = & (w_1 \cdot \text{Normalized_Volume}) + (w_2 \cdot \text{Normalized_Views}) \\ & + (w_3 \cdot \text{Brand_Tier_Bonus}) - (w_4 \cdot \text{Inventory_Penalty}) \end{aligned} \quad (1)$$

2.3. Weight Justification

The weights (w_1, w_2, \dots) are configurable parameters designed for optimization. The initial proposed weights are:

- $w_1 = 0.4$ **(Volume)**: Heavily prioritizes products that are actually selling.
- $w_2 = 0.3$ **(Views)**: Rewards trending products that capture user attention.
- $w_3 = 0.2$ **(Brand)**: Provides a strategic lever to boost partner brands.
- $w_4 = 0.1$ **(Inventory)**: Slightly deprioritizes items with excessive inventory.

These weights can be fine-tuned via A/B testing to maximize performance against specific KPIs.

3. Automated Process Flow Map

The end-to-end process is designed for full automation, from data ingestion to frontend rendering, while retaining a crucial layer for manual business control.

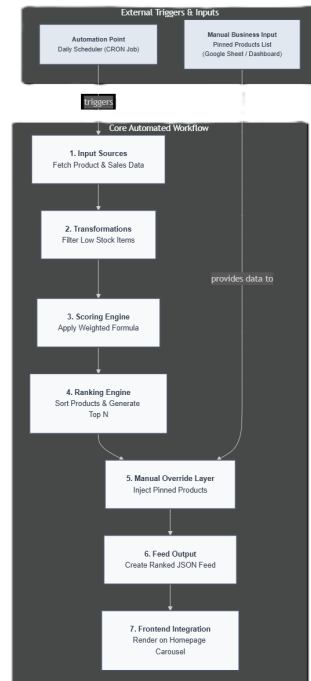


Figure 1: End-to-end automated merchandising workflow.

4. Step-by-Step Workflow Description

The process visualized in Figure 1 follows a logical, sequential execution:

1. **Data Ingestion:** A scheduled job (e.g., a daily CRON script) pulls the latest product, sales, and inventory data from internal databases or APIs.
2. **Data Pre-processing:** The system first applies a hard filter, removing any products with low stock (e.g., 'Units in Stock < 10') to prevent showing items that a customer cannot purchase.
3. **Scoring Engine:** The agent iterates through the filtered products, calculating a dynamic score for each one using the weighted formula from Equation 1.
4. **Ranking Algorithm:** All scored products are sorted in descending order to create a preliminary ranked list.
5. **Manual Override Layer:** Before finalizing, the system checks a separate control source (e.g., a Google Sheet or an internal dashboard). If the merchandising team has "pinned" any Product IDs, those products are moved to the top of the list in the specified order. This provides essential business control for promotions or strategic placements.

6. **Output Generation:** The final ranked list of Product IDs is formatted into a clean, easily digestible format, such as JSON, and pushed to a location accessible by the frontend.
7. **Frontend Integration:** The SKINSEOUL website's homepage carousel component fetches this JSON feed and dynamically renders the top N products, ensuring the "Best Sellers" are always current, relevant, and optimized for business goals.