Retail Promotions Analysis: Courts vs Harvey Norman

Introduction

This project focuses on web scraping product data from two of Singapore's most popular electronics and home appliance retailers: Courts and Harvey Norman. With the upcoming Hari Raya celebrations in 2025, both retailers have launched special promotional campaigns offering significant discounts across various product categories.

The primary objectives of this web scraping project are to:

1. Extract comprehensive product information from both Courts and Harvey Norman websites
2. Identify the top discounted products from each retailer's Hari Raya promotions
3. Compare identical or similar products across both retailers to find the most affordable options
4. Analyze pricing trends and discount patterns to help consumers make informed purchasing decisions

By systematically collecting and analyzing this data, the project aims to provide valuable insights for consumers looking to maximize savings during the Hari Raya promotional period, while also demonstrating effective web scraping techniques for retail price comparison.

Project Overview

The Harvey Norman Product Data Scraper is a Python-based web scraping tool that extracts detailed product information from Harvey Norman Singapore's website. It specifically targets product listings and extracts data such as:

- Product ID
- Product name
- Brand name
- Primary category
- Additional categories
- Price information

Requirements

- Python 3.7+
- Beautiful Soup 4
- Requests
- Pandas
- Regular expressions (re)

Methodology

This project employed a systematic approach to extract promotional product data from Courts and Harvey Norman websites. The methodology focused on efficiently gathering comprehensive product information while respecting website structures and ethical scraping practices.

Technology Stack

The web scraping implementation relied on the following technologies:

- **Python 3.7+**: Core programming language for the project
- **Beautiful Soup 4**: HTML parsing library to navigate and extract data from web page structures
- **Requests**: HTTP library to fetch web pages and handle sessions
- **Pandas**: Data manipulation library for organizing and analyzing the extracted data
- **Regular expressions (re)**: Pattern matching for extracting specific data formats and cleaning text

Scraping Approach

1. Initial Research and Website Analysis

Before writing any code, I conducted a thorough analysis of both websites:

- Manually navigated through the Hari Raya promotional pages on both Courts and Harvey Norman
- Identified common product information patterns and data structures
- Examined the HTML structure and JavaScript data layers using browser developer tools
- Checked for any rate limiting, anti-scraping measures, or robots.txt restrictions
- Documented the URL patterns for promotional pages and pagination systems

2. Courts Website Scraping Strategy

For the Courts website:

1. **Entry Point Identification**: Located the main Hari Raya promotion landing page and mapped the category structure
2. **Pagination Handling**: Implemented logic to navigate through multiple pages of product listings
3. **Data Extraction Method**:
    - Primary approach: Located embedded JavaScript objects (dataLayer or similar) containing structured product data
    - Secondary approach: Direct HTML parsing of product grid elements when JavaScript data wasn't available
4. **Session Management**: Maintained consistent session cookies to simulate normal browsing behavior

3. Harvey Norman Website Scraping Strategy

For the Harvey Norman website:

1. **Promotion Page Navigation**: Located and mapped the Hari Raya promotional sections
2. **JavaScript Data Extraction**:
    - Identified script tags containing product arrays with item details
    - Used regular expressions to extract and convert JavaScript objects to valid JSON
    - Parsed structured data to extract comprehensive product information
3. **HTML Parsing Fallback**: Implemented secondary extraction logic using BeautifulSoup to parse product elements directly from the DOM
4. **Category and Brand Extraction**: Special focus on extracting hierarchical category information and normalizing brand names

4. Data Extraction Workflow

The general extraction workflow for both websites followed these steps:

1. Send HTTP requests with appropriate headers (User-Agent, Accept, etc.) to avoid blocking
2. Handle response status codes and implement retry logic for failed requests
3. Parse the HTML content using BeautifulSoup
4. Extract product data using dual approaches:

    - Search for JavaScript data structures containing structured product information
    - Parse HTML elements containing product details as fallback
5. Extract key product attributes:
    - Product ID and SKU
    - Product name
    - Brand information
    - Price (original and discounted)
    - Discount percentage/amount
    - Product categories
    - Product specifications
6. Store extracted data in structured Python dictionaries
7. Implement appropriate delays between requests to respect server resources

5. Parallel Processing

To improve efficiency while maintaining ethical scraping practices:

- Implemented throttled parallel processing for handling multiple pages
- Used thread pools with controlled concurrency (max 3-5 concurrent requests)
- Maintained sufficient delays between requests to the same domain
- Implemented exponential backoff for retry attempts

Data Extraction

The extraction process focused on gathering comprehensive product information from both Courts and Harvey Norman websites during their Hari Raya 2025 promotional campaigns.

Courts Data Extraction

For the Courts website, I implemented a structured approach to navigate through their promotional pages:

1. **Initial Page Access**: Started with the main Hari Raya promotional landing page to access all featured products.

2. **Navigation Through Categories**: Systematically accessed each product category section to ensure comprehensive coverage of all promotional items.

3. **Product Information Extraction**: For each product listing, I extracted:

    - Full product name (which typically contained brand and model information)
    - Current promotional price
    - Original price (when available)
    - Product URL for reference

4. **Pagination Handling**: Implemented logic to navigate through multiple pages of results within each category section to capture all available products.

Harvey Norman Data Extraction

The Harvey Norman website required a different approach due to its unique structure:

JavaScript Data Targeting: Identified that product information was primarily stored in JavaScript variables within the page source. For example:

var items = [{ 'item_id': '85044', 'item_name': 'Samsung 530L 2 Door Fridge - Refined Inox (RT53DG7A6CS9SS)', 'item_brand': 'S', "item_category" : 'Fridges', ...}];

1.
2. **Regular Expression Extraction**: Used regex patterns to locate and extract these JavaScript data structures that contained rich product information.

3. **Data Conversion**: Transformed the extracted JavaScript objects into structured Python dictionaries for further processing.

4. **HTML Fallback Method**: In cases where JavaScript data wasn't available, implemented a secondary extraction method that directly parsed the HTML elements containing product information.

5. **Multi-Page Extraction**: Systematically worked through all pages of the Harvey Norman Hari Raya promotional section to ensure complete data collection.

Handling Website Variations

Both websites presented unique challenges that required adaptive extraction techniques:

- **Dynamic Content**: Some promotional sections loaded products dynamically with JavaScript, requiring session management and request headers that mimicked browser behavior.

- **Inconsistent Data Structures**: Product information wasn't always presented in a consistent format, requiring flexible parsing logic.

- **Rate Limiting Consideration**: Implemented appropriate delays between requests to respect server resources and avoid IP blocking.

Data Cleaning

After extraction, the raw data required several cleaning steps to make it suitable for analysis and comparison:

1. Brand Extraction

The combined product information often included brand names embedded within the full product name:

- **Brand Identification**: Extracted brand names from the beginning of product names where they typically appeared (e.g., "Samsung 530L 2 Door Fridge" → brand: "Samsung").

- **Brand Normalization**: Standardized brand representations across both retailers. For example, "S" was mapped to "Samsung" to ensure consistent comparison.

2. Price Cleaning

Price data required significant processing:

- **Currency Symbol Removal**: Removed the "$" sign from all price values to convert them to numeric format.

- **String to Numeric Conversion**: Transformed price strings into float values for mathematical operations.

- **Missing Price Handling**: Implemented logic to handle cases where original prices were not explicitly listed.

3. Product Type & Category Classification

Product categorization was derived from analyzing the full product names:

- **Keyword Extraction**: Identified key product type indicators in names (e.g., "Fridge", "TV", "Washing Machine").

- **Hierarchical Categorization**: Created a category hierarchy by grouping products into major categories (e.g., "Kitchen Appliances", "Home Entertainment").

- **Category Standardization**: Normalized category names across retailers to enable direct comparison of similar product types.

4. Discount Calculation

Created additional data points to facilitate deal comparison:

Percentage Discount: Calculated discount percentage using the formula:

```
discount_percentage = ((original_price - current_price) / original_price) * 100
```

Absolute Discount Amount: Calculated the actual dollar savings:

```
discount_amount = original_price - current_price
```

5. Data Structuring

The final cleaned data was structured into a comprehensive CSV file with the following columns:

- `product_name`: Full product name as displayed on the website
- `brand`: Extracted and normalized brand name
- `product_type`: Primary product type (e.g., TV, Refrigerator)
- `category`: Broader product category (e.g., Electronics, Home Appliances)
- `original_price`: Original price before discount
- `current_price`: Current promotional price
- `discount_amount`: Dollar amount saved
- `discount_percentage`: Percentage discount offered

This structured dataset provided the foundation for subsequent analysis to identify the best deals across both retailers during the Hari Raya promotional period.