SRH University

Big Data Systems for Fashion E-Commerce

Presented by :

Anusha Srinivasulu	- 100003468
Ashwin Jayan	- 100002367
Sumit Patil	- 100003409

Big Data Infrastructure | 05-02-2025

Table of Contents

- Introduction
- Technology Stack
- Data Collection
- Data Preparation & Enrichment
- Data Storage in HDFS
- SQL Database Integration
- Results & Challenges
- Future Scope
- Conclusion

Introduction

This project aims to streamline the extraction of fashion product details such as pricing, descriptions, and images from various e-commerce platforms using Selenium & BeautifulSoup for web scraping. The collected data will be stored in HDFS for scalable and distributed storage, ensuring efficient handling of large datasets. A MySQL database will then be used to organize and structure the data, enabling advanced querying and analysis. The final goal is to provide actionable insights into trends, pricing strategies, and inventory management, empowering businesses to make data-driven decisions in the competitive fashion industry. Additionally, the system will be designed to support real-time updates and integration with analytics tools for dynamic reporting.

Technology Stack

This project integrates various technologies to ensure efficient data collection, storage, and retrieval:

- Scraping Tools: Selenium, BeautifulSoup for dynamic website parsing.
- Storage & Processing: HDFS on Docker Swarm (distributed architecture).
- Database: MySQL for structured data storage.
- Orchestration & Connectivity: Docker, Hadoop, Tailscale for networking and distributed processing.

Data Collection (Scraping Process)

- H&M: Extracts product details using BeautifulSoup and Selenium.
- Zara: Handles dynamic content loading with incremental scrolling.
- Superdry: Utilizes element selection and structured data extraction.
- Cookies, pop-ups, and pagination are managed to ensure complete data extraction
- All data is stored in a JSON file before enrichment.

Data Preparation & Enrichment

- The scraped data is merged into a single dataset from all three sources.
- Enrichment Process: Additional attributes such as gender, clothing type, fit type, and season are extracted based on predefined keyword matching from product title.
- This ensures that data is properly categorized for meaningful insights.
- The final dataset is stored as enriched_fashion_products.json.

Data Storage in HDFS

- The enriched dataset is split into five tables:
    - Products (product details)
    - Pricing (price information)
    - Images (image URLs)
    - Links (product links)
    - Inventory (stock availability)
- Data is uploaded to HDFS running on a Docker Swarm cluster (2 workers, 1 master).
- Redundancy Test: A worker is shut down to verify data fault tolerance.

SQL Database Integration

- Extracted data from HDFS is inserted into MySQL tables.
- Five relational tables store product attributes, ensuring structured queries and efficient retrieval.
- Queries executed include:

  Find most common price point for products.
  Total products available in each category.
  Products name along with their inventory price.

Results & Challenges

✅ Achievements:

- Successfully extracted and structured thousands of products.
- Built a scalable, fault-tolerant data pipeline with HDFS & MySQL.
- Enabled structured SQL queries for insights.

❌ Challenges:

- Handling dynamic pages in web scraping.
- Ensuring consistency in extracted data.
- Managing large datasets efficiently.

Future Scope

- Machine Learning Integration: Predict pricing trends and demand.
- Automated API Scraping: Reduce reliance on Selenium for faster data collection.
- Performance Optimization: Improve query efficiency using indexing and caching.

These improvements will enhance the scalability and accuracy of the pipeline.

Conclusion

This project demonstrates a complete data pipeline, from scraping to storage to analysis, leveraging Selenium for data extraction, HDFS for distributed storage, and MySQL for structured querying. The use of distributed systems and SQL databases ensures a scalable and efficient solution, capable of handling large volumes of data while maintaining performance and reliability. Additionally, the pipeline is designed to be modular, allowing for easy integration of new data sources or analytical tools in the future.

Thank You