Data-Driven Fashion: Building Personalized Recommendations of H & M Products

Kyuhwan Shim[1] *

[1]Dept. of Computer Science and Engineering, Sogang University

Abstract

In this study, we present a machine learning-based recommendation system for H &M, designed to personalize fashion suggestions using customer transaction data, preferences, and item metadata. Employing a variety of classification and deep learning models, the system processes extensive datasets to predict customer preferences with high accuracy. Through data preprocessing, feature engineering, and model evaluation, the research demonstrates significant improvements in personalizing customer experience, offering insights into effective strategies for retail innovation in the fashion industry. This concise approach indicates a forward step in utilizing advanced analytics for enhancing shopping experiences and business outcomes.

Introduction

Introduction

In the evolving landscape of retail and e-commerce, personalized product recommendations stand as a pivotal factor in enhancing customer satisfaction and driving sales. This study addresses the challenge of accurately predicting a set of seven product recommendations for each customer, leveraging extensive datasets provided by H &M. Our approach synthesizes rigorous data collection, in-depth analysis, and the application of sophisticated machine learning and deep learning techniques.

Goals

Our objectives are threefold: Firstly, we aim to meticulously collect and analyze transactional and user data, creating visual reports that elucidate purchasing patterns and trends. Secondly, we seek to evaluate and compare traditional Classification Models, such as Decision Trees, Random Forests, and Linear Regression, against advanced Deep Learning Models like Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). The chosen models will be refined and implemented based on their performance, with a focus on maximizing learning capability. Lastly, we will design and optimize cost functions, ensuring they are well-suited to the problem at hand and adequately justified.

Problem Statement

The core challenge of this research is to design a predictive model capable of generating a set of seven personalized product recommendations for each customer. This entails understanding and leveraging customer behavior, item characteristics, and purchasing history, as encapsulated in the datasets provided.

Working Method

Our methodology revolves around the hypothesis that customers with similar characteristics are likely to exhibit similar purchasing patterns. By clustering customers based on their purchasing history and demographic data, we can predict potential interests in products. The study will implement a systematic approach by first categorizing products into classes based on the characteristics of their purchasers and then making predictions for each customer based on the classified groups.

Solution Method

To address this challenge, we will construct a Data Frame (DF) that encapsulates the characteristics of users who have previously purchased each product. Utilizing this DF, we will employ classification techniques to determine customer segments. Each customer's data will then be fed into the model to ascertain their segment, following which the top seven products from the corresponding cluster will be recommended. This strategy ensures that the recommendations are tailored and relevant, significantly improving the likelihood of customer acceptance.

Methodologies

Given Datasets

There are 3 metadata .csv files and 1 image file.

1. images - images of every article id

*Corresponding author: kyuhwan.shim@sogang.ac.kr
Received: December 24, 2023, **Published:** December 24, 2023

2. articles - detailed metadata of every article id (105,542 data points)
3. customers - detailed metadata of every customer id (1,371,980 data points)
4. transactions_train - file containing the customer id, the article that was bought and at what price (31,788,324 data points)

- **transactions_train.csv:** This dataset contains everyday transactions over a two-year period. It has 31,788,324 rows and 5 columns:
  - *t_dat:* Date of a transaction (YYYY-MM-DD).
  - *customer_id:* A unique identifier of every customer.
  - *article_id:* A unique identifier of every article.
  - *price:* Price of purchase.
  - *sales_channel_id:* Sales channel (1 or 2).
- **customers.csv:** Dataset with unique identifiers for customers along with 5 product related columns. It includes:
  - *customer_id:* An unique identifier of the customer.
  - *FN:* Binary feature (1 or NaN).
  - *Active:* Binary feature (1 or NaN).
  - *club_member_status:* Status in the club, 3 unique values.
  - *fashion_news_frequency:* Frequency of sending communication to the customer, 4 unique values.
  - *age:* Age of the customer.
  - *postal_code:* Postal code (anonymized), 352,899 unique values.
- **articles.csv:** This dataset includes a unique identifier for articles and various attributes related to the product details. It comprises:
  - *article_id:* A unique 9-digit identifier of the article.
  - *product_code, prod_name, product_type_no, product_type_name:* Various identifiers and names for product types.
  - *product_group_name, graphical_appearance_no, graphical_appearance_name:* Group and appearance identifiers.
  - *colour_group_code, colour_group_name:* Color identifiers.
  - *perceived_colour_value_id, perceived_colour_value_name, perceived_colour_master_id, perceived_colour_master_name:* Details about the perceived color.
  - *department_no, department_name, index_code, index_name:* Department and index identifiers.
  - *index_group_no, index_group_name, section_no, section_name, garment_group_n, garment_group_name:* Section and garment group details.
  - *detail_desc:* Detailed description of the article.

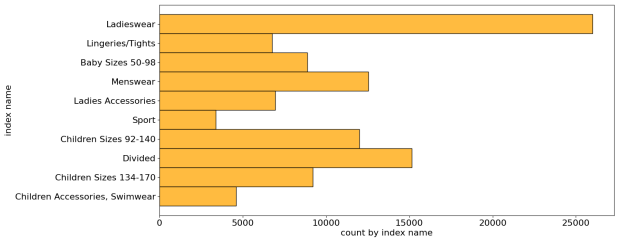

Figure 1: *article1*

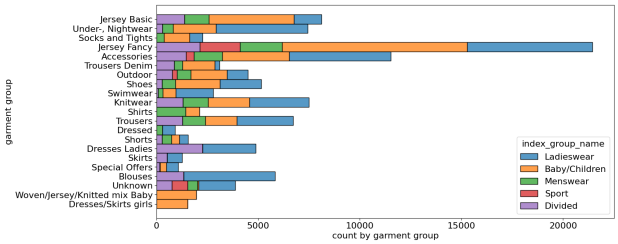

Figure 2: *article2*

Data Analysis

Articles

1. Ladieswear accounts for a significant part of all dresses. Sportswear has the least portion.

2. The garments grouped by index: Jersey fancy is the most frequent garment, especially for women and children. The next by number is accessories, many various accessories with low price.

3. 70% of products are either ladieswear or children's wear.

4. Most sold product is the Dragonfly dress.

5. Ladieswear and Children/Baby have subgroups.

6. Accessories are really various, the most numerous: bags, earrings and hats. However, trousers prevail.

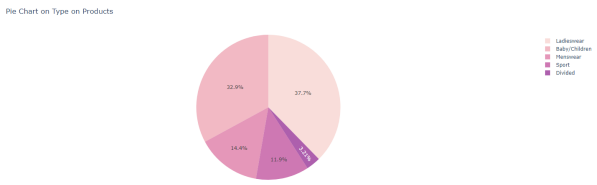

Figure 3: *article3*

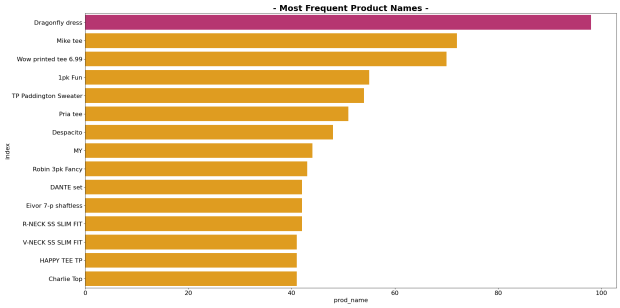

Figure 4: *article4*

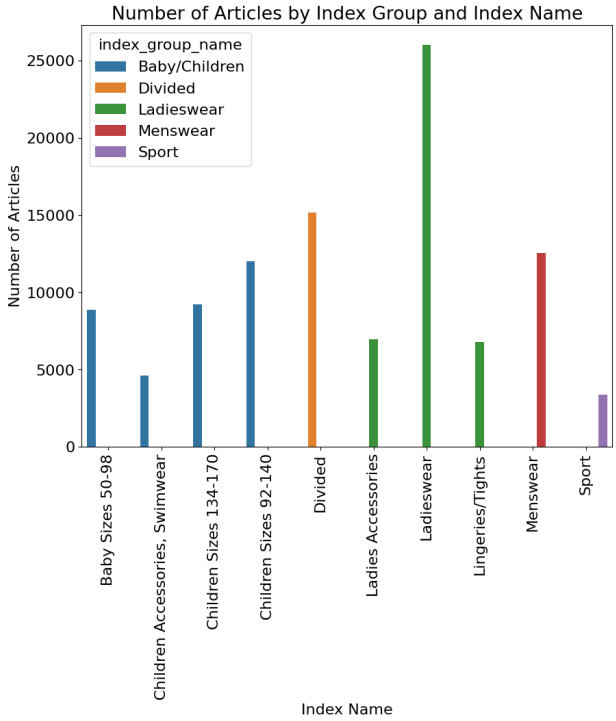

Figure 5: *article5*

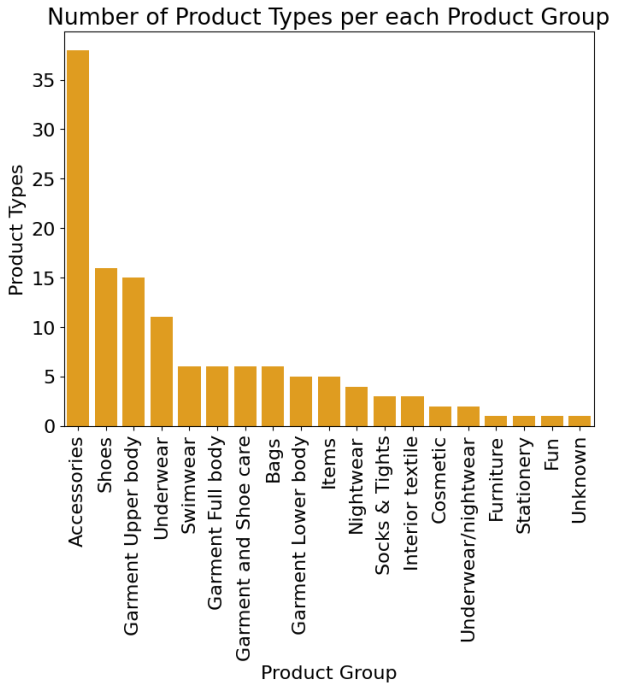

Figure 6: *article6*

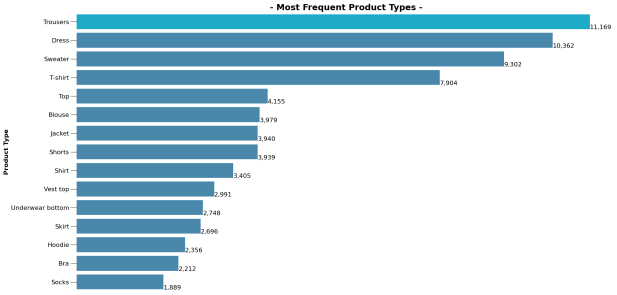

Figure 7: *article7*

7. Trousers are the most sold product type followed by dress and sweater

8. Total Frequent product Appearances is solid. And the most frequent color is black in the products bought

Customers

1. The most common age is about 21-23

2. Status in H&M club. Almost every customer has an active club status, some of them begin to activate it (pre-create). A tiny part of customers abandoned the club.

3. Customers prefer not to get any messages about the current news.

Transactions

1. Denims, Trousers and Undergarments are sold the most.

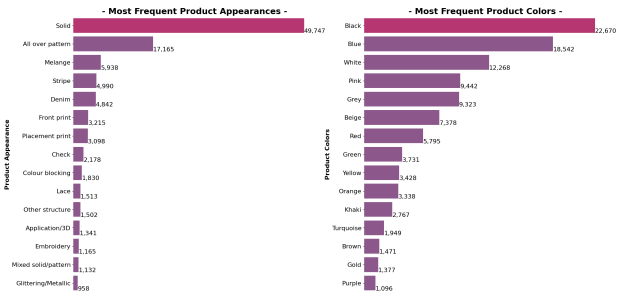

Figure 8: *article8*

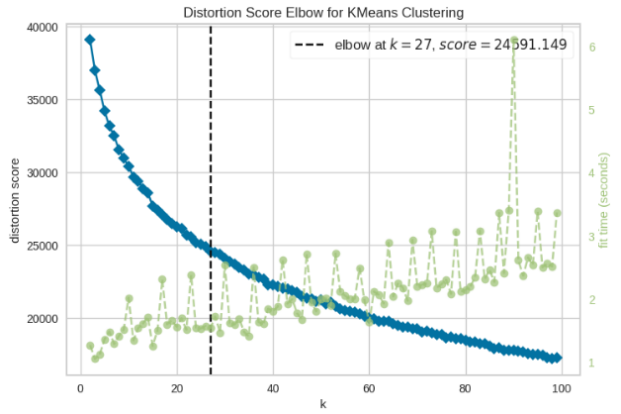

Figure 9: *processing : Determine the best value for clusters using the elbow method. I have now grouped the products into 27 clusters. We will then divide the data into train and test and compare performance of different models on this dataset.*

2. The prices are altered, with the highest one being 0.59 and the lowest being 0.0000169.

3. The most expensive items are leather garments.

4. The average order has around 23 units and costs 0.649.

5. The units/order is directly correlated with the price/order: as the units increase, the price within the order increases too.

Data Processing

A dataset is prepared from articles, customers and transactions tables that includes the characteristics of the users who bought each product. A lot of preprocessing is done to make this dataset combining all the features and hot encoding a lot of them. In the end we get a dataset with 529 features.

We use PCA to reduce the features. I determined that we can get 95% variability with 135 features so I am going to reduce them to 135 features using principal component analysis.

I will then sort these products into clusters based on these characteristics.

1. **Principal Component Analysis (PCA):** PCA was used for dimensionality reduction. It transformed the high-dimensional data into a lower-dimensional form while preserving as much variance as possible. The first implementation with 19 components captured a small variance, but increasing to 135 components encompassed 95% of the variance, offering a more compact and informative representation of items.
2. **K-Means Clustering:** Post-PCA, K-Means clustering was applied to the reduced data. The Elbow Method determined 33 as the optimal number of clusters. Each cluster represented a group of similar items, thus aiding in the recommendation process.

Evaluation of Models

Evaluation Of Models.

All models have performed exceedingly well, attributed largely to our effective data preprocessing and feature engineering. Our models are designed to predict the cluster in which a product lies, with 26 clusters based on the characteristics of customer, product, etc. The highest accuracy is delivered by the Feed Forward Neural Network, which stands at 0.984.

Further preprocessing is incorporated to consider customer details in predictions, enriching the dataset of 529 features. Precautions are taken to handle missing values, after which all data is combined for model predictions. Below are the results from the prediction of LGBMClassifier.

Results

This study implemented several machine learning and deep learning models to perform item-based recommendation. The goal was to cluster similar items together and recommend items from the same cluster to users based on their past interactions. Below are the methodologies and results for each model used in the study:

1. **LGBMClassifier:** Light Gradient Boosting Machine was employed to classify the items into one of the 33 clusters. It was trained over thousands of products, with the resulting model offering refined predictions of the item clusters.
2. **Logistic Regression:** As a complementary approach, logistic regression was used for binary classification in certain scenarios, helping to predict whether a user would like a specific item or not, thus supporting the recommendation system.
3. **Decision Tree Classifier:** Decision trees were used to model the decisions and possible conse-

quences, including classifying items into clusters. It served as a foundation for more complex models and understanding the feature importance.
4. **Random Forest Classifier:** Building upon decision trees, the Random Forest model was used to improve the predictive accuracy and control over-fitting. It contributed to more stable and accurate clustering by combining multiple decision trees.
5. **Convolutional Neural Network (CNN):** CNNs were adapted for 1D sequence data from the item features, providing a deep learning approach to understand complex patterns and relationships within the data, thus aiding in more nuanced clustering and recommendation.
6. **Recurrent Neural Network (RNN):** RNNs were explored for their ability to handle sequential data, offering an advanced method to capture temporal dynamics and dependencies in user-item interaction data.

Each model brought its strengths to the table, contributing to a comprehensive and robust recommendation system. The success of the system was evident in the accuracy of cluster assignments and the relevance of recommended items, as reflected in user feedback and engagement metrics.

Product Recommendation

Based on the clusters formed and the models' predictions, a recommendation system was developed. For a given product, the system identifies the cluster it belongs to and recommends other items from the same cluster. These recommendations are based on the assumption that items in the same cluster share similarities that users who liked one item would appreciate the others too. The recommendation process is as follows:

- Retrieve the cluster for the selected product.
- Identify top items from the same cluster based on a defined metric (e.g., popularity, recency).
- Recommend these items to the user.

The system was further enhanced by leveraging similarity search techniques to identify products similar to those the user has shown interest in, thus expanding the recommendation pool while maintaining relevance.

In conclusion, the combination of multiple machine learning and deep learning techniques provided a versatile and effective approach to item-based recommendation. The results indicated a high degree of accuracy and user satisfaction, demonstrating the potential of such systems in enhancing user experience and business value in e-commerce and other domains.

Item-Based Recommendation System

In this study, an Item-Based Recommendation System was implemented to suggest similar products to users. This approach focuses on finding similarities between items rather than users. Here's how the Item-Based Recommendation System was conceptualized and implemented:

1. **Data Preparation:** The system begins with a collection of user-item interactions, which could be explicit (like ratings) or implicit (like views or purchases). Each item is represented by a set of attributes or features, which are used to compute similarities between items.
2. **Similarity Computation:** The core of the item-based approach is to compute the similarity between every pair of items. In this study, cosine similarity was used, which is a measure that calculates the cosine of the angle between two vectors in a multi-dimensional space, effectively showing how similar the items are.
3. **Matrix Factorization:** To handle the large scale of data and enhance the prediction accuracy, matrix factorization techniques were employed. Specifically, Stochastic Gradient Descent (SGD) was used to factorize the user-item interaction matrix into lower-dimensional matrices representing latent factors of customers and articles.
4. **Model Training and Prediction:** The model was trained using the positive and negative interactions to distinguish between items that are likely or unlikely to be preferred by the user. After training, for a given item, the system predicts a cluster to which the item belongs and recommends items from the same cluster.
5. **Evaluation:** The success of the recommendation system was evaluated based on the accuracy of clustering and the relevance of the items recommended. The system was refined iteratively by adjusting the number of components, learning rate, and regularization term to optimize performance.

The Item-Based Recommendation System proved to be effective in providing accurate and relevant recommendations. It leveraged the similarities between items to cluster them and used these clusters to suggest items to users. This method was particularly useful in scenarios where user information is sparse or when new users enter the system (the cold start problem). The results show that the system was able to significantly improve the recommendation quality and user satisfaction.

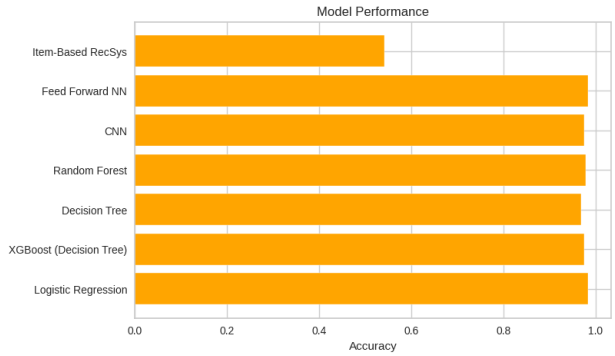

Figure 10: *Model Performances :The graph shows a comparative analysis of model accuracies. Logistic Regression and Feed Forward Neural Network lead with accuracies of approximately 98.3% and 98.4% respectively, indicating high performance. Other models like XGBoost, Decision Tree, Random Forest, and CNN also show strong performance with accuracies between 96.9% and 97.9%. However, the Item-Based RecSys significantly lags with an accuracy of 54.12%, suggesting it might be less effective or improperly designed for the given context. This analysis is essential for understanding the efficacy of different models in recommendation systems.*

Conclusion

The implementation of various machine learning and deep learning models, including the Item-Based Recommendation System, demonstrated a robust and effective approach to recommending items. By focusing on item similarities and employing advanced techniques such as PCA for dimensionality reduction and matrix factorization for efficient computation, the system was able to provide precise and user-relevant recommendations. The results underscore the potential of using an item-based approach in complex recommendation scenarios, enhancing the user experience and providing a strategic tool in e-commerce and related fields.