H&M Personalized Fashion Recommendations


## ABSTRACT

This paper presents a comprehensive experimental study of different recommendation system approaches implemented on H&M's fashion retail dataset. The research compares item-based and user-based collaborative filtering, content-based filtering using product metadata, K-means clustering based on RFM analysis, and ALS matrix factorization. Each method was independently implemented and assessed using Mean Average Precision at K=12. ALS Collaborative Filtering performed the best, with the highest MAP@12 score of 0.00786. This method works well in fashion retail because it captures hidden user preferences and provides more personalized, accurate recommendations. The hybrid methods and advanced machine learning techniques will be applied to improve performance.



**Keywords**: Fashion Recommendation system, Collaborative Filtering, Content-Based Filtering, Matrix Factorization, RFM Analysis, Customer Segmentation, K-means Clustering


## INTRODUCTION

Most customers enjoy online shopping, but they often face an overwhelming number of product choices, making the experience time-consuming and frustrating. This challenge is particularly significant for H&M Group, which operates in 53 online markets with approximately 4,850 stores, offering an extensive product selection. With the global e-commerce fashion industry projected to reach $1.0 trillion by 2025, effective recommendation systems have become crucial not only for customer satisfaction and business success but also for environmental sustainability. Accurate recommendations can significantly reduce product returns and associated transportation emissions, aligning with H&M's commitment to sustainable fashion retail.

The primary goal of this research is to create a recommendation system that predicts which products customers will purchase in the 7-day period immediately after the training data ends. Using H&M transaction data from 09-19-2018 to 09-21-2020, along with customer and product metadata, we aim to develop a recommendation system that not only helps shoppers find suitable items more efficiently but also increases customer satisfaction and drives revenue growth. The findings from this study will contribute to both academic understanding of recommendation systems and practical applications in the fashion e-commerce industry.

This research addresses three key questions:

1. Which recommendation algorithm performs best for fashion retail in terms of prediction accuracy and user relevance?
2. How effectively can RFM analysis enhance recommendation accuracy by identifying distinct customer purchasing patterns?
3. What role does product metadata play in improving recommendation accuracy?

The study evaluates multiple recommendation algorithms using MAP@12 as the performance metric, with the goal of identifying the most effective approach for fashion e-commerce recommendations.

## LITERATURE REVIEW

Recent developments in recommendation systems have significantly evolved to address the complexities of e-commerce personalization. Roy and Dutta [1] provide a fundamental framework by categorizing recommendation systems into three main types: content-based, collaborative, and hybrid systems, while emphasizing persistent challenges including cold-start problems, data sparsity, and scalability issues.

**Collaborative Filtering Approaches:** Research in collaborative filtering has demonstrated both promise and limitations. Wang et al. [2,16] implemented user-based collaborative filtering utilizing cosine similarity and KNN algorithms, showing effectiveness for personalized recommendations. However, their findings revealed significant computational complexity challenges with large-scale datasets. Their later work on combining user-based and item-based approaches showed improved accuracy in handling data sparsity, though scalability remained a concern.

**Advanced Clustering and Hybrid Methods:** Several researchers have explored clustering-based approaches specifically for fashion retail. Bellini et al. [3] pioneered a multi-clustering approach combining online and physical store data, achieving a notable 3.48% increase in purchase rates. Building on this, Yıldız et al. [4] integrated RFM analysis with geographical data and K-means clustering, though their approach was constrained by limited sample size and persistent cold-start issues. Hwangbo et al. [18] advanced this field with K-RecSys, successfully combining online click behavior with offline purchase data, demonstrating improved recommendation accuracy in real-world retail applications.

**Matrix Factorization and Deep Learning:** Matrix factorization techniques have emerged as particularly promising. Loukili et al. [5] and Koren et al. [6] demonstrated the superiority of matrix factorization approaches, especially ALS, over traditional methods in handling sparse datasets. The field has further evolved with deep learning applications, where Kandoi et al. [7] and Wang et al. [8] achieved remarkable results using VGG16, MobileNet, and LSTM-CNN architectures, reporting 94-100% similarity scores for top 5 recommendations. These advances suggest the potential for deep learning in capturing complex fashion preferences.

**Content-Based and Hybrid Approaches:**Content-based filtering has shown particular strength in addressing specific recommendation challenges. Suvarna and Balakrishna [9] achieved 89.02% accuracy using deep CNN models, while Chia and Najafabadi [10] effectively addressed the cold-start problem through content-based approaches. Hybrid systems have further advanced the field, with Geetha et al. [15] and Tu and Dong [17] developing sophisticated systems that integrate multimedia mining and dynamic user preference alignment.

A critical analysis of the existing literature reveals several significant gaps in recommendation system research for fashion retail. While individual studies have demonstrated success with specific techniques, such as Wang et al. [2] with user-based collaborative filtering and Loukili et al. [5] with ALS matrix factorization, there is a notable absence of comprehensive studies comparing multiple approaches within a single fashion retail context. Most research has focused on isolated techniques, making it difficult to understand relative performance under consistent conditions. Furthermore, the variation in evaluation metrics and datasets across studies, as noted by Chakraborty et al. [20], hinders direct comparisons and practical insights for large-scale implementations.

To address these gaps, our research implements and evaluates seven distinct recommendation approaches: user-based collaborative filtering, content-based filtering with both cosine similarity and KNN, item-based collaborative filtering, RFM analysis with K-means clustering, and ALS matrix factorization. Each approach is independently implemented and assessed using MAP@12 as a consistent evaluation metric, enabling direct performance comparisons. This comprehensive methodology allows us to evaluate the strengths and limitations of each approach within the fashion retail context while providing practical insights for large-scale e-commerce applications.

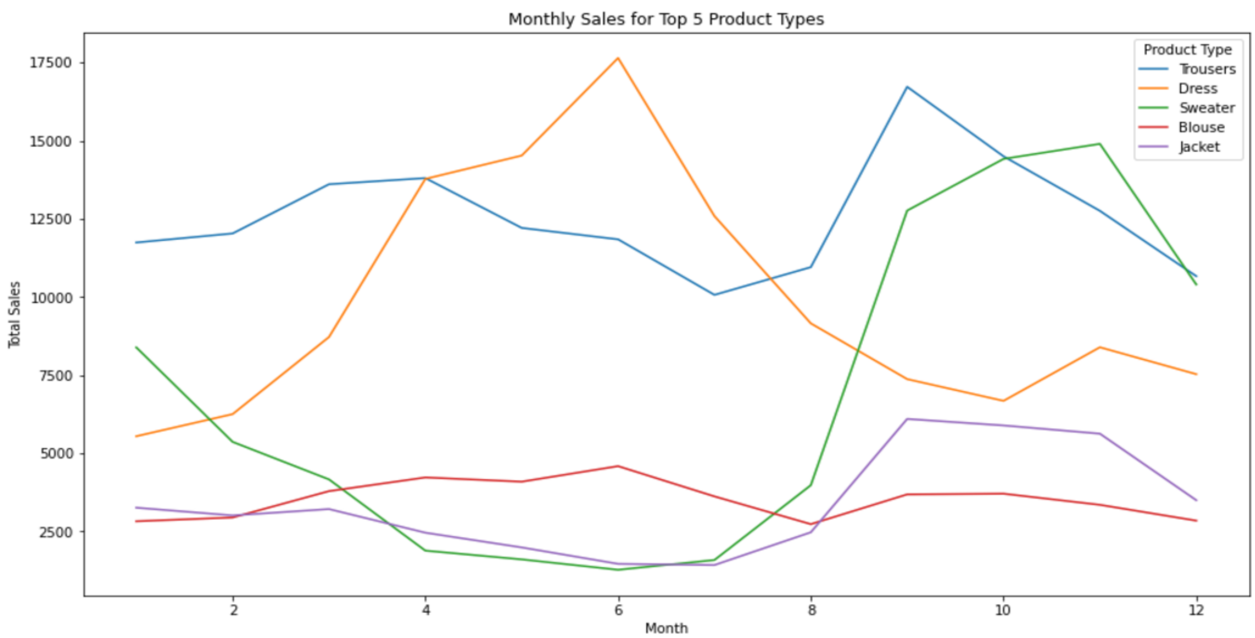
## DATA

The dataset used in the analysis was obtained from the H&M Personalized Fashion Recommendations competition hosted on Kaggle [23].

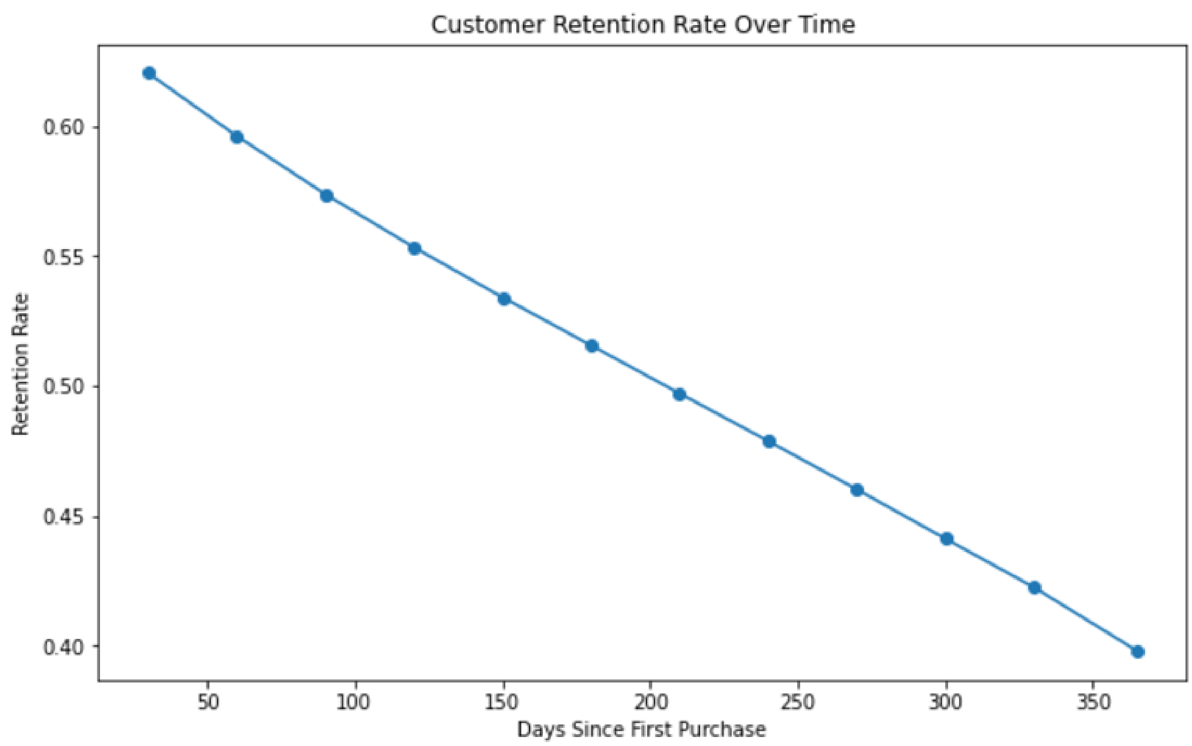
1. Articles: There are 105,542 rows and 25 columns. article_id is the unique identifier of every article. Customers: there are 1,371,980 rows and 7 columns. customer_id is the unique identifier of every customer.
2. Transactions_train: There are 31,788,324 rows and 7 columns.
3. The customer_id and article_id are in this table so we can merge the subset from articles or subset from customers with Transactions_train table to combine the three data tables.
4. We explored each feature and found below results:

| | ***article_id frequency*** | ***age*** | ***customer_id frequency*** | ***transaction frequency per date*** | ***price*** |
|---|---|---|---|---|---|
| ***count*** | 104547 | 31648066 | 1362281 | 734 | 31788324 |
| ***mean*** | 304.057735 | 36 | 23 | 43308 | 0.027829 |
| ***std*** | 791.26606 | 13 | 39 | 17075 | 0.019181 |
| ***min*** | 1 | 16 | 1 | 12760 | 0.000017 |
| ***0.25*** | 14 | 25 | 3 | 33936 | 0.015814 |
| ***0.5*** | 65 | 31 | 9 | 39516 | 0.025407 |
| ***0.75*** | 286 | 47 | 27 | 47448 | 0.033881 |
| ***max*** | 50287 | 99 | 1895 | 198622 | 0.591525 |

5. Based on the figure, most items are purchased less than 286 times, and 75% of transactions are made by customers younger than 47 years old. Most of the customs made purchases less than 27 times from 2018-09-20 to 2020-09-22. The average frequency purchased made per day is 43,308. 75% of items purchased are less than 0.033881.
6. The figure below shows the data shows strong seasonality, with dresses peaking around month 6 (summer), while sweaters and trousers show increased sales starting from month 8 (fall/winter), indicating clear weather-dependent buying patterns.

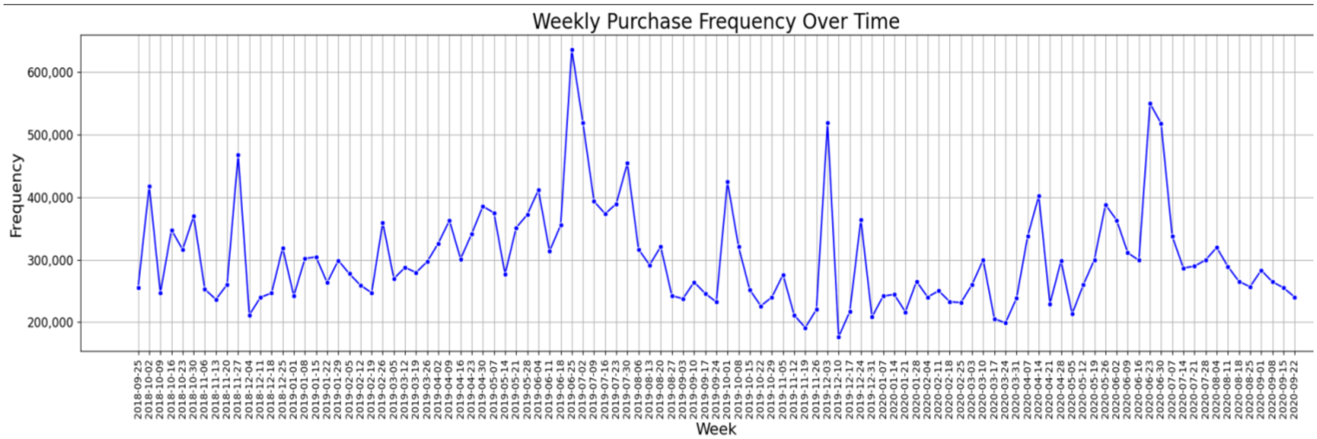



7. The customer retention rate figure below shows a clear downward trend, dropping from about 62% at day 50 to 40% by day 350, suggesting H&M needs to focus on strategies to maintain long-term customer engagement and loyalty.




8. Below is the purchase frequency by week from 2018-09-20 to 2020-09-22. According to the pattern from the figure, we can see the purchase frequency has seasonality. Based on the graphs, there are consistent spikes during November and December, suggesting

a strong seasonal trend associated with holiday shopping. There are also some noticeable rises in purchase activities around March, April,and July, likely from seasonal promotions. The main downtrends in purchase frequency occur in January-February, following the holiday season, and in August, likely due to vacation and back-to-school periods.




9. Since most of H&M's customers are young and the prices of items selling are cheap, the preference of the customers changes fast and has seasonality. In addition, the dataset is too large, and we may need to try down sampling. Therefore, it may be worth trying to use the last 5-weeks data as train dataset and last week as valid dataset.

## METHODOLOGY

Data Sampling

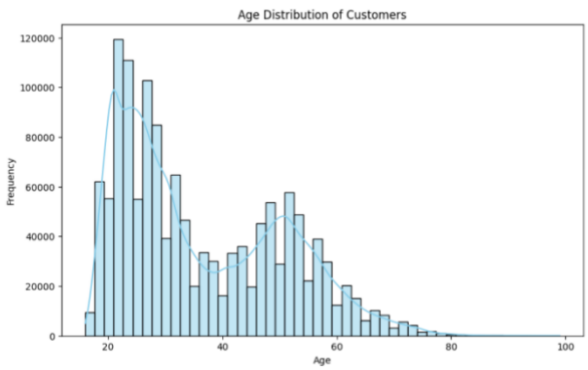
Given the huge size of the dataset - with articles having 105,542 rows, transactions totaling 28,813,419 rows, and customers numbering 1,371,980 - we decided to perform down-sampling. We selected only the most recent six weeks of data, as H&M operates in a fast fashion, where trends are seasonal and rapidly changing, which allowed us to focus on relevant data for our analysis.

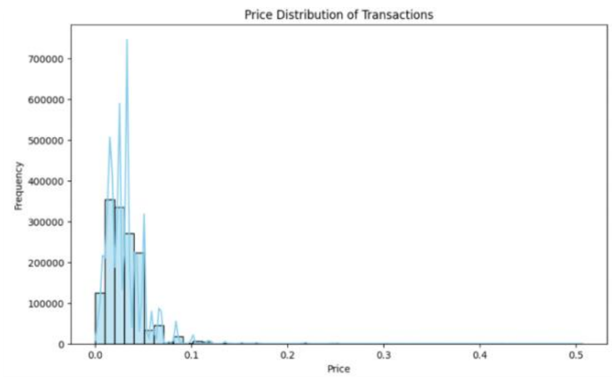
We took the most recent 6 weeks of transaction data for our analysis and divided our dataset into five-sixths for training and one-sixth for testing. We used the transactions from 2020-08-12 to 2020-09-15 as our training data. For our valid set, we used transactions from the last week, from 2020-09-16 to 2020-09-22. This is to evaluate the performance.

Model 1 – Content Based Filtering (Approach 1)

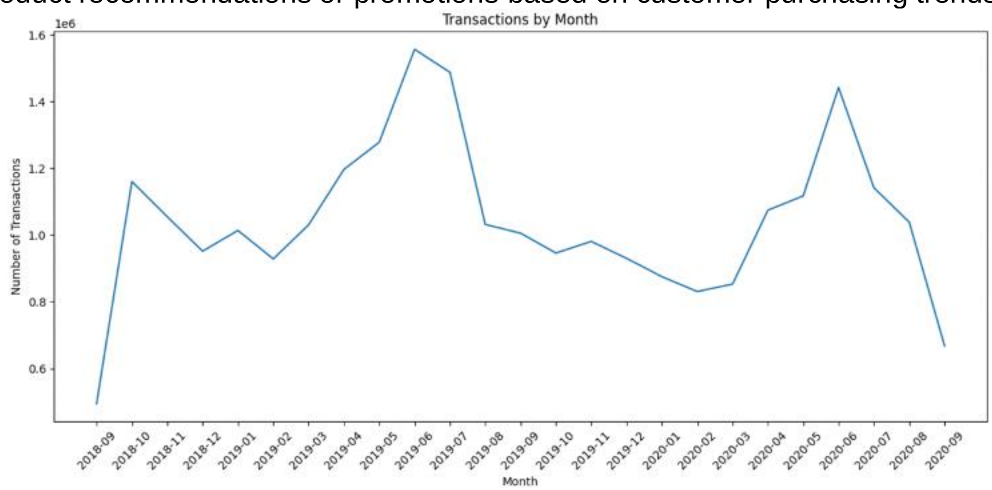
Thie idea of using Content-based filtering initially was because, with only six weeks of data, customer purchase behavior might be too limited to find meaningful comparisons between customers. Some customers may not purchase frequently enough during this period, which could make collaborative filtering less effective. By focusing on product attributes, content-based filtering can still provide personalized recommendations even with limited customer data.

1. Exploratory Analysis: To understand the dataset and get meaningful insights, we performed exploratory data analysis to explore key variables and their relationships through visualizations as below.

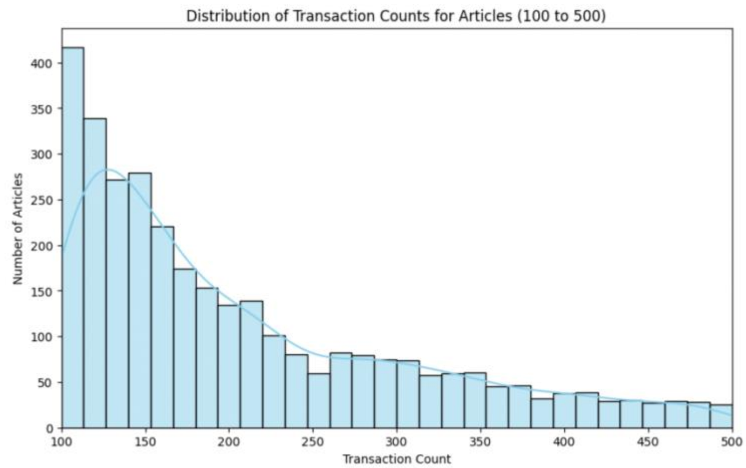


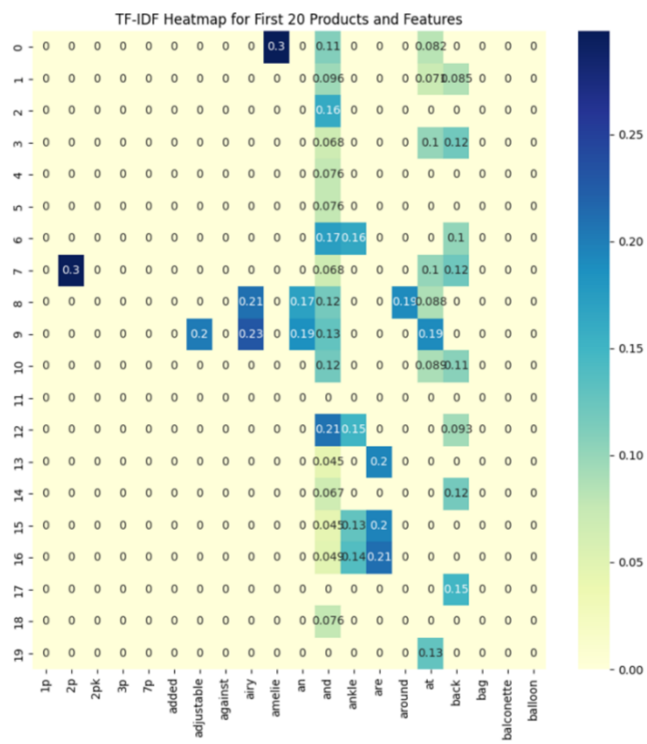

From the analysis, we observed notable spikes in transactions that happened during October and June. This purchasing behavior could be leveraged for creating seasonal product recommendations or promotions based on customer purchasing trends.




2. Data Preprocessing**:** Filtered the transactions to focus on a 6-week time frame, from 2020-08-12 to 2020-09-22, and performed data cleaning by dropping n/a and duplicates to ensure data quality.
3. Calculated the transaction count for each article and set a threshold of 100 to identify top-selling items, ensuring that we focus on products with sufficient sales volume rather than items with little or no sales.
4. Merged the transactions, articles, and customer data into a dataframe, and filtered for the top-selling products. (The threshold was set to 100, see the below figure.) This created a comprehensive view of the data relevant to our recommendation task.




5. Feature Engineering:
   - Categorical features, such as product type, color and other product attributes, were encoded using OneHotEncoder to convert them into binary format for model training. This is important because machine learning algorithms require numerical input, and categorical features must be converted into numerical format for the model to process.
   - Textual features, such as prod_name and detail_desc, were combined to create a new feature for text analysis applying TF-IDF to represent the text data in a meaningful way. This transformation shows how important each word is in the product description compared to other descriptions in the dataset, which allows the model to understand the textual content meaningfully and match customer preferences with article features.




   - Numerical features, such as price, were normalized using StandardScaler to ensure that all features contribute similarly to the model. Normalization is important as it can confound other features like product categories or textual information. Therefore, it helped to ensure that all features are in a comparable scale and prevented any feature from influencing the model.

6. All engineered features were merged into a final data frame for further analysis, and the final dataset was split into training and testing sets, using the majority of the data (approximately 83%) for training and the last week's transactions for testing to evaluate the model's performance.
7. Calculated customer preferences by grouping the training data by customer_id and calculating the mean of relevant features. This helped us get information including their preferred product types and budget ranges for each customer based on their purchasing history.
8. Extracted article features from the dataset and calculated the cosine similarity between customer preferences and article features. The cosine similarity measures the angle distance between customer preferences and product features, which show how related each customer is to the articles. It is used often for evaluating similarity in recommendation systems as it works well when the data is sparse, in this case, many products are not purchased by all customers.
9. For each customer, we generated recommendations by selecting the top K articles based on similarity scores and creating a list of recommended articles for each customer.
10. Evaluate: MAP@12= 0.00132. The result showed that there is significant room for improvement. While the model captures some of the customer preferences, it is better to refine the feature selections or perform different feature engineering to get better performance.

Model 2 – Content Based Filtering (Approach 2)

We developed multiple content-based recommendation systems utilizing product descriptions and customer transaction data. Here we present two approaches that differ primarily in their similarity computation methods.

1. Data Preparation:
   The preprocessing pipeline transforms product descriptions through lowercase conversion, special character removal, and whitespace normalization. These cleaned descriptions are then converted into numerical vectors using TF-IDF vectorization, limited to 5,000 features with English stop words removed.
2. System Implementation:
   Approach 1 - Cosine Similarity: Our first recommendation engine generates personalized product suggestions using cosine similarity. It constructs a similarity matrix using preprocessed product descriptions, where each product's TF-IDF vector is compared against all others using cosine similarity. This matrix serves as the foundation for content-based recommendations.
   Approach 2 - K-Nearest Neighbors: Our second implementation uses KNN to identify similar products. After the TF-IDF transformation, the system finds the k-nearest neighbors for each product based on the distance between their feature vectors. This approach provides a more localized similarity measure compared to the cosine similarity method.
3. The recommendation generation process for both approaches differs based on customer purchase history:
   For customers with existing purchases:
   1. The system retrieves the customer's unique purchase history
   2. For each previously purchased item, it calculates similarity scores with all other products in the catalog (using either cosine similarity or KNN)
   3. These similarity scores are averaged across all customer purchases to create a final similarity vector
   4. Products are ranked by their similarity scores, excluding items already purchased
   5. The top 12 highest-scoring products are selected as recommendations

For new customers (cold start):

1. The system calculates purchase frequency for all products in the training period
2. Products are ranked by their popularity (purchase count)
3. The top 12 most popular items are recommended, ensuring diverse product types

3. Evaluate:
   - Cosine Similarity Approach: MAP@12 = 0.0048
   - KNN Approach: MAP@12 = 0.0065

Model 3 – User-Based Collaborative Filtering

1. Train data and Valid data Feature Engineering:
   Take the last 6-week transactions data: the transactions data for last week (2020-09-16 to 2020-09-22) as valid data and the rest of the 5-weeks (2020-08-12 to 2020-09-15) transactions data as train data
   Train data and Valid data Engineering:
   First combine the transaction dataset with customer dataset, calculate the purchase frequency by zip code, the average purchase price by zip code, and the average purchase price by customer, regroup the dataset by customer ID and merge the three columns mentioned above, drop zip code column to avoid creating too many columns when encoding it to calculate distance and lastly fill in null value.
2. Calculate Gower Distance
   Gower distance [21] is a similarity measure that can be used in recommendation systems, especially when dealing with mixed data types. It is well-suited for calculating similarity between items or users when you have a heterogeneous dataset (containing different types of features), which is well-suited for the dataset I am using here.
   Key Characteristics:
   - Handles Mixed Data Types: Unlike traditional distance measures like Euclidean or Manhattan distance, Gower distance can handle a combination of:
     - Numerical features (e.g., age, rating score) ----To be normalize the numerical features
     - Categorical features (e.g., genre, product category) ---One-hot encode categorical features
     - Binary features (e.g., like/dislike)---One-hot encode categorical features
   - Scales Each Feature: Gower distance scales each feature independently to ensure that no single feature dominates the distance calculation. This is particularly important when features have different ranges or scales.
   - Value Range: Gower distance ranges from 0 (most similar) to 1 (most dissimilar).

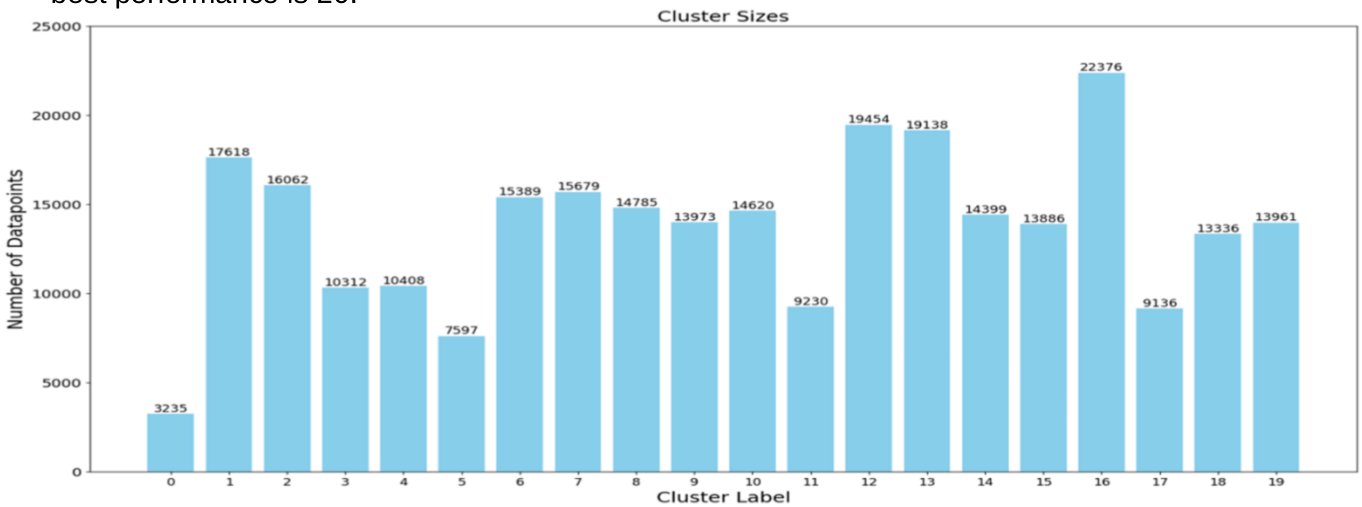
As explained above, to fit the data to calculate the Gower Distance, the numerical features in the train datasets are normalized and the categorical features are one-hot encoded.

3. Find the nearest neighbors
   Compute Calculate Gower Distance with each customer in the train dataset for each valid user and get the nearest neighbor in the train dataset. The recommendation items are the items bought by the nearest neighbor in the train dataset in the past five weeks.
4. Evaluate: MAP@12= 0.00064

Model 4 - K-means Clustering Based on RFM and Associate Rule Mining

1. Take the last 6-week transactions data: the transactions data for last week (2020-09-16 to 2020-09-22) as valid data and the rest of the 5-weeks (2020-08-12 to 2020-09-15) transactions data as train data

2. Calculate the RFM Score for each customer in the train dataset. RFM stands for Recency, Frequency, and Monetary value, and each factor corresponds to a key customer trait.
   - Recency measures how recently a customer made a purchase.
   - Frequency measures how often a customer makes a purchase within a specified time.
   - Monetary Value measures the total amount of money a customer has spent within a specified time.
3. Use K-means clustering to cluster the customers based on RFM scores. Different numbers of clusters are used to evaluate the performance. The optimal clustering number to generate the best performance is 20.




4. Find the items frequently purchased together in each cluster by using Association rule Mining. The Apriori algorithm is used for frequent item set mining and association rule learning to identify the items pairs frequently purchased together with. The minimum support is set as 0.001 and the minimum confidence value as 0.1.
5. Based on this analysis, for existing customers, we recommend the 12 most popular items within their cluster and suggest items frequently bought together within the same cluster based on their previous purchases. For new customers, we recommend the 12 most popular items purchased in the past 5 weeks.
6. Evaluate: MAP@12=0.00524

Model 5-K- Alternating Least Squares Collaborative Filtering

1. Take the last 6-week transactions data: the transactions data for last week (2020-09-16 to 2020-09-22) as valid data and the rest of the 5-weeks (2020-08-12 to 2020-09-15) transactions data as train data.
2. The dataset was converted into a matrix where each row represents a user, each column represents an item, and each cell contains an interaction score. In this case, the interaction score is the frequency (How often a user interacted with the item)
3. The ALS algorithm was applied to decompose the sparse user-item matrix into two smaller matrices: one for customers and one for products. These matrices approximate the original matrix by capturing latent patterns. The ALS algorithm was applied to decompose the sparse user-item matrix into two smaller matrices: one for customers and one for products. These matrices approximate the original matrix by capturing latent patterns.[5]
4. Parameter tuning: Factors, regularization and iterations have been tuned. The best parameter combination is Factor = 150, Regularization = 0.05, Iterations = 10

5. We also expanded the training data size to one year of transactions to test for performance improvement, but the MAP@12 dropped to 0.00460, suggesting that the 5-week transaction data is more effective for training.
6. Evaluate the model with best parameter combination: MAP@12=0.00786

Model 6 – Item Based Collaborative Filtering

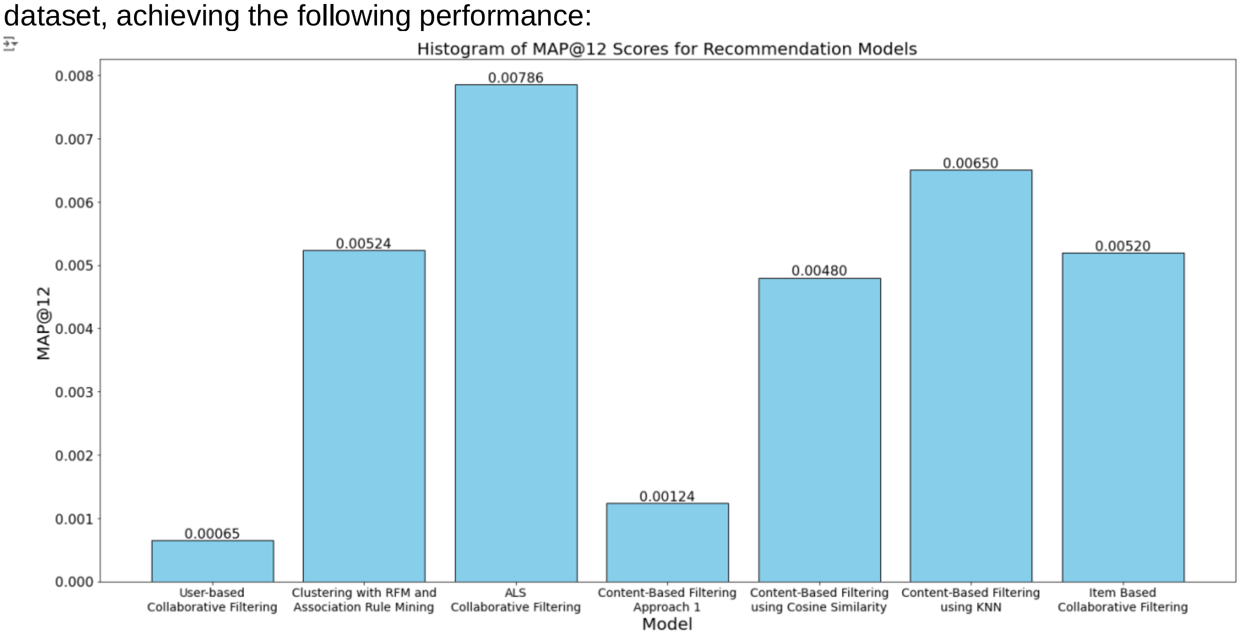
1. The foundation of our collaborative filtering system is a sparse user-item interaction matrix. We construct this matrix using scipy's sparse matrix implementation (csr_matrix) to efficiently handle the large-scale data.
2. The matrix uses binary values, where 1 represents a purchase interaction and 0 represents no interaction. To maintain data integrity and enable easy lookups, we create mapping dictionaries that preserve the original user and item identifiers.
3. The core of our recommendation engine relies on computing item-item similarities using the cosine similarity metric. We apply this computation to the transpose of our interaction matrix to obtain similarities between items based on user purchase patterns. This approach allows us to identify products that are frequently purchased together or by similar user groups.
4. The system implements two distinct recommendation strategies to handle different user scenarios:

   For Existing Users:
   - Uses purchase history to identify similar items
   - Excludes previously purchased products
   - Recommends top 12 most similar items

   For New Users:
   - Implements popularity-based recommendations
   - Suggests top 12 most frequently purchased items
5. Evaluate: MAP@12= 0.0052.

**RESULTS**

We implemented and evaluated five different recommendation approaches on the H&M fashion dataset, achieving the following performance:



Key findings:

1. ALS Collaborative Filtering performed the best, with the highest MAP@12 score of 0.00786. This method works well in fashion retail because it captures hidden user preferences and provides more personalized, accurate recommendations.
2. Content-Based Filtering Approach 1 had the low score of 0.00124, which means that using product descriptions alone is not enough to make good recommendations for customers in this case.
3. The user-based collaborative filtering can resolve the cold start issue but cannot predict the customers' purchase behavior well, with a score of 0.00065. This is probably because there are not enough common interactions between customers and products, which makes it harder to recommend relevant items based on customer behavior alone.
4. RFM Analysis with K-means Clustering scored 0.00524, which is decent but not as strong as ALS. It shows that segmenting customers based on their purchasing habits can provide some insights and understand customer purchasing behaviors, but it does not work as well as ALS.
5. Our streamlined approach using just product descriptions outperformed the complex feature combination method (MAP@12: 0.0048 vs 0.00124), suggesting that additional features introduced noise rather than enhancing the rich information already present in descriptions.
6. Increasing the size of the dataset did not result in significant performance improvements.

## CONCLUSION

ALS Collaborative Filtering performed the best, with the highest MAP@12 score of 0.00786. The superior performance of these approaches can be attributed to their ability to effectively capture user-item interactions and handle the sparsity of the fashion retail dataset. The RFM analysis with K-means clustering provided valuable customer segmentation insights, while user-based filtering helped address the cold-start problem. However, user-based collaborative filtering showed limited effectiveness, due to the sparse nature of user interactions in the fashion domain.

Our multi-approach recommendation system delivers significant business value. We can immediately display new products through content-based filtering, while our ALS models drive personalized recommendations that boost customer engagement.

## FUTURE WORK

Looking ahead, there are several exciting opportunities we could explore to enhance these results. It involves enhancing user-based recommendations to address the cold start issue, developing hybrid recommender systems by integrating deeper learning techniques, particularly CNN models for image-based recommendations and LSTM networks for capturing temporal fashion trends. To address the cold-start problem more effectively, future work should explore hybrid approaches combining collaborative filtering with content-based methods and leveraging cross-domain knowledge. The recommendation system could be enhanced by incorporating visual features through computer vision techniques to analyze product images, colors, and patterns.We can also try NLP technologies and CNN to help in predicting related products [22].

## CONTRIBUTION

Shanmukhi, Shu, and Chin-Ya worked together on abstract, introduction, and literature review. Shu was solely responsible for the User-based Collaborative Filtering and Clustering with RFM and Association Rule Mining. Shu and Shanmukhi collaborated on the Alternating Least Squares Collaborative Filtering, while Chin-Ya worked on Content-Based Filtering Approach 1 and Shanmukhi worked on content-based filtering approach 2. The Item-Based Collaborative Filtering was developed solely by Shanmukhi. The Model Comparison was conducted by Shanmukhi, Shu, and Chin-Ya, and the Result and Conclusion sections were written collaboratively by Shanmukhi, Shu, and Chin-Ya. All three authors contributed to data collection and the writing of the paper.

## REFERENCES


[1] Deepjyoti Roy & Mala Dutta (2022). *A systematic review and research perspective on recommender systems.* Journal of Big Data 9:59.

[2] Hulong Wang et al (2021). *User-based collaborative filtering algorithm design and implementation*. Journal of Physics: Conf. Ser. 1757 012168

[3] Pierfrancesco Bellini· Luciano Alessandro Ipsaro Palesi · Paolo Nesi · Gianni Pantaleo (2022). *Multi Clustering Recommendation System for Fashion Retail*. Multimedia Tools And Applications 82:9989–10016

[4] Emre Yıldız, Ceyda Güngör Sen, And Eyüp Ensar Isık(2023). *A Hyper-Personalized Product Recommendation System Focused on Customer Segmentation: An Application in the Fashion Retail Industry*. Theoretical And Applied Electronic Commerce Research. 18(1), 571-596

[5] Manal Loukili, Fayçal Messaoudi and Fayçal Messaoudi (2023). *Personalizing Product Recommendations using Collaborative Filtering in Online Retail: A Machine Learning Approach.* 2023 International Conference on Information Technology (ICIT)

[6] Y. Koren, R. Bell, and C. Volinsky, *"Matrix factorization techniques for recommender systems,"* Computer, vol. 42, no. 8, pp. 30-37, 2009.

[7] K. Kandoi et al., *"Comparative Analysis of Deep Learning Models for Fashion Recommendation in E-commerce,"* 2023 IEEE Fifth International Conference on Advances in Electronics, Computers and Communications (ICAECC), 2023.

[8] H. Wang, Z. Chao, and N. Lou, *"A Personalized Movie Recommendation System based on LSTM-CNN,"* 2020 2nd International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI), 2020.

[9] B. Suvarna and S. Balakrishna, *"An Efficient Fashion Recommendation System using a Deep CNN Model,"* 2022 International Conference on Automation, Computing and Renewable Systems (ICACRS), 2022.

[10] E. J. Chia and M. K. Najafabadi, *"Solving Cold Start Problem for Recommendation System Using Content-Based Filtering,"* 2022 International Conference on Computer Technologies (ICCTech), 2022.

[11] T. K. Quan, I. Fuyuki, and H. Shinichi, *"Improving Accuracy of Recommender System by Clustering Items Based on Stability of User Similarity,"* 2006 International Conference on Computational Intelligence for Modelling Control and Automation (CIMCA), 2006.

[12] S. Patel et al., *"Smart Fashion Recommendation System using Random Forest Algorithm,"* 2024 International Conference on Electrical Electronics and Computing Technologies (ICEECT), 2024.

[13] Javed, U.; Shaukat, K.; Hameed, I.A.; Iqbal, F.; Alam, T.M.; Luo, S. (2021). A Review of Content-Based and Context-Based Recommendation Systems. *International Journal of Emerging Technologies in Learning (iJET)*, 16(3), 274-292.

[14] Chakraborty, S.; Hoque, M.S.; Rahman Jeem, N.; Biswas, M.C.; Bardhan, D.; Lobaton, E. (2021). Fashion Recommendation Systems, Models and Methods: A Review. *Informatics*, 8(3), 49.

[15] Geetha, G.; Safa, M.; Fancy, C.; Saranya, D. (2018). A Hybrid Approach using Collaborative Filtering and Content Based Filtering for Recommender System. *Journal of Physics: Conference Series*, 1000, 012101.

[16] Wang, J.; de Vries, A.P.; Reinders, M.J.T. (2006). Unifying User-based and Item-based Collaborative Filtering Approaches by Similarity Fusion. *In Proceedings of the SIGIR Conference*, Seattle, Washington, USA.

[17] Tu, Q.; Dong, L. (2010). An Intelligent Personalized Fashion Recommendation System. *In Proceedings of the IEEE International Conference on Multimedia and Expo*, 479-484.

[18] Hwangbo, H.; Kim, Y.S.; Cha, K.J. (2018). Recommendation system development for fashion retail e-commerce. *Electronic Commerce Research and Applications*, 28, 94-101.

[19] Pazzani, M. J., & Billsus, D. (2007). Content-Based Recommendation Systems. In P. Brusilovsky, A. Kobsa, & W. Nejdl (Eds.), *The Adaptive Web* (LNCS 4321, pp. 325–341). Springer-Verlag Berlin Heidelberg.

[20] Chakraborty, S., Hoque, M.S., Rahman Jeem, N., Biswas, M.C., Bardhan, D., Lobaton, E. (2021) Fashion *Recommendation Systems, Models and Methods: A Review*. Informatics 8, 49.

[21] Pinyan Liu, Han Yuan, Nan Liu, Marco Aurélio Peres. (2024) A modified Gower distance-based clustering analysis for mixed-type data, *SSRN* *https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4779022*

[22] Akhilesh Kumar Sharma, Bhavna Bajpai, Rachit Adhvaryu, Suthar Dhruvi Pankajkumar, Prajapati Parthkumar Gordhanbhai, Atul Kumar f. (2023) An Efficient Approach of Product Recommendation System using NLP Technique, Materials Today: Proceedings, Volume 80, Part 3

[23] Carlos García Ling, ElizabethHMGroup, FridaRim, inversion, Jaime Ferrando, Maggie, neuraloverflow, and xlsrln. H&M Personalized Fashion Recommendations. https://www.kaggle.com/competitions/h-and-m-personalized-fashion-recommendations/data ,2022. Kaggle.