



How can data-driven techniques be optimized to provide accurate personalized product recommendations based on customer Size and Color preferences?

*A thesis submitted to the University of Birmingham for
the degree of MSc Data Science*

Submitted by
Suyash Sanjay Yadav
ID - 2560807

Under the Guidance of
Dr. Qamar Natsheh

School of Computer Science
College of Engineering and Physical Sciences
University of Birmingham
2023-24

ABSTRACT

The growing demand for personalized shopping experiences in e-commerce has made recommendation systems a vital component of successful online retail strategies. The research looks into developing a robust recommendation system designed for Nike Plus, using cutting-edge machine learning technologies to improve product suggestions' accuracy. The main aim of this project is to deliver very detailed recommendations that match the preferences of customers optimally and tailor them for each individual customer, focusing in-depth on attributes like size and color. This project focuses on developing a content-based recommendation system tailored to Nike Plus's product catalog, aiming to enhance customer engagement by providing personalized product suggestions based on attributes such as size and color. Using a content-based filtering approach, the system generates relevant product recommendations by matching user input preferences with key product attributes. The project also addresses the limitations of content-based filtering, such as the potential for reduced recommendation diversity, and explores future directions, including the integration of collaborative filtering and advanced natural language processing techniques. The goal is to improve the overall effectiveness of the recommendation engine by making it smarter and more personalized, ultimately enhancing the customer's shopping experience by delivering highly tailored product suggestions. The result is not only better customer satisfaction, but also a significant increase in sales conversion rate due to very relevant recommendations that directly play into the hands of what Nike Plus customers prefer. The aim of this research is to develop and evaluate a personalized product recommendation system for Nike Plus that leverages content-based filtering techniques. This system is designed to enhance the shopping experience by providing a seamless and personalized way of interacting with Nike Plus products, while continuously improving the accuracy and relevance of recommendations over time.

KEYWORDS

Recommendation system, Machine learning, Personalized experience, Content-Based filtering Tailored, Customer preferences.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my project supervisor, Dr. Qamar Natsheh, for her unwavering support, guidance, and encouragement throughout this research journey. Her expertise and insightful feedback have been invaluable, helping me navigate the challenges of this project and pushing me to achieve my best work.

I am also thankful to Dr. Rami Bahsoon, my project inspector, for his valuable feedback during the demonstration, which provided me with crucial insights to refine and enhance my work.

I am deeply thankful to the faculty and staff of University of Birmingham for providing the resources and knowledge that made this research possible. I would like to acknowledge my classmates and colleagues for their support, advice, and the many thought-provoking discussions that enriched my understanding and approach to this project.

Finally, I want to thank my family and friends for their constant support and understanding during this demanding period. Their encouragement and belief in me have been a source of strength, allowing me to stay focused and motivated.

Thank you all for your contributions to the successful completion of this thesis.

ABBREVIATIONS

API - Application Programming Interface

CBF - Content-Based Filtering

CF - Collaborative Filtering

CNN - Convolutional Neural Network

RFM - Recency, Frequency, Monetary (analysis)

TF-IDF - Term Frequency-Inverse Document Frequency

BERT - Bidirectional Encoder Representations from Transformers

MAP – Mean Average Precision

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION	9
1.1 RESEARCH MOTIVATIONS	10
1.2 LIMITATIONS OF THE GENERAL RESEARCH.....	10
1.3 AIM AND OBJECTIVES	10
1.4 LEGAL, SOCIAL, ETHICAL AND PROFESSIONAL ISSUES	11
1.5 THESIS OVERVIEW	12
1.6 THESIS STRUCTURE.....	12
CHAPTER 2: BACKGROUND	14
2.1 BACKGROUND RESEARCH.....	14
2.2 UTILISED ALGORITHMS AND METRICS	15
2.2.1 COMMON MODELLING APPROACHES FOR RECOMMENDATION SYSTEM.....	15
2.2.2 PERFORMANCE METRICS.....	20
CHAPTER 3: LITERATURE REVIEW	22
3.1 RELATED WORKS.....	22
3.2 KNOWLEDGE GAP	23
CHAPTER 4: DATA.....	25
4.1 RETRIEVING THE DATA.....	25
4.2 DATA PREPARATION & PREPROCESSING	25
4.3 RATIONALE TOWARDS EXPLORING THE DATA.....	27
4.3.1 EXPLORATORY DATA ANALYSIS (EDA).....	27
4.3.2 FEATURE ENGINEERING.....	30
CHAPTER 5: METHODOLOGY	31
5.1 CONTENT BASED FILTERING USING COSINE SIMILARITY	32
5.1.1 THEORY.....	32
5.1.2 IMPLEMENTATION.....	33
5.1.3 COSINE SIMILARITY ALGORITHM FOR RECOMMENDATION.....	34
5.2 TF-IDF (TERM FREQUENCY-INVERSE DOCUMENT FREQUENCY)	35
5.2.1 THEORY.....	35
5.2.2 IMPLEMENTATION.....	37
5.2.3 TF-IDF ALGORITHM FOR RECOMMENDATION.....	37
5.3 PROJECT MANAGEMENT	38
CHAPTER 6: RESULTS	39

CHAPTER 7: DISCUSSION.....	42
CHAPTER 8: CONCLUSIONS	43
8.1 THESIS CONTRIBUTION	43
8.2 CHALLENGES AND LIMITATIONS	44
8.3 FUTURE WORK	44
REFERENCES	46
APPENDIX.....	49

LIST OF FIGURES

FIGURE 1. 1 THE FUNDAMENTAL CONCEPT OF THE PROPOSED APPROACH.....	12
FIGURE 2. 1 DESCRIPTIVE WORKING OF COLLABORATIVE FILTERING.....	16
FIGURE 2. 2 ARCHITECTURE FOR CONTENT-BASED RECOMMENDATION SYSTEM.....	17
FIGURE 2. 3 COMPARISON OF COLLABORATIVE FILTERING AND CONTENT- BASED FILTERING IN RECOMMENDATION SYSTEMS.....	18
FIGURE 2. 4 ARCHITECTURE OF HYBRID RECOMMENDATION MODEL	19
FIGURE 4. 1 DISTRIBUTION OF TOP 10 BEST-SELLING PRODUCTS	27
FIGURE 4. 2 SALES DISTRIBUTION BY DEPARTMENT	28
FIGURE 4. 3 TOP 10 BEST-SELLING COLOR COMBINATIONS.....	28
FIGURE 4. 4 TOP 10 BEST-SELLING PRODUCT SIZES	29
FIGURE 4. 5 CORRELATION MATRIX OF NUMERICAL PRODUCT VARIABLES.....	29
FIGURE 5. 1 PROPOSED APPROACH.....	32
FIGURE 6. 1 OUTPUT	39
FIGURE 6. 2 OUTPUT	40
FIGURE 6. 3 PRODUCT RECOMMENDATION SYSTEM DASHBOARD.....	41
FIGURE 6. 4 PERFORMANCE METRICS	41

CHAPTER 1: INTRODUCTION

In today's competitive retail environment, delivering personalized shopping experiences is essential. With the increasing prominence of e-commerce and the integration of data-driven strategies in business, consumers now expect brands to tailor their interactions to meet individual needs and preferences. Nike Plus, a global frontrunner in athletic apparel and footwear, is no exception to this evolving trend. Renowned for its commitment to innovation, Nike Plus aims to create personalized and engaging customer experiences. One of the most effective approaches to achieving this is through the development of sophisticated recommendation systems that can accurately suggest products aligned with each customer's specific preferences. Consequently, building a recommendation system has become a pivotal aspect of many initiatives, necessitating the use of advanced machine learning techniques to deliver precise product predictions.

The core focus of this research is the optimization of Nike Plus's recommendation algorithms to deliver personalized product recommendations, with a particular emphasis on essential attributes such as shoe size and color. In the retail industry, it is well recognized that recommendation systems play a crucial role in driving sales, enhancing customer satisfaction, and fostering engagement. However, traditional systems frequently fall short in providing recommendations that are accurately tailored to individual customer preferences. This shortcoming often results in subpar shopping experiences, where customers encounter difficulties in finding products that precisely match their needs, ultimately leading to missed sales opportunities and decreased customer retention.

The primary challenge this project seeks to address is the optimization of machine learning algorithms to enhance the accuracy of personalized product recommendations. At the heart of this research is the challenge of optimizing recommendation algorithms to provide personalized Nike Plus product recommendations based on individual customer preferences, particularly in terms of size and color ([Zhang et al., 2017](#)). Traditional recommendation systems often struggle to offer personalized suggestions that truly match the unique needs of each customer. This can lead to less satisfying shopping experiences and missed chances for sales ([Adomavicius & Tuzhilin, 2005](#)). The main goal of this research is to resolve these issues by employing diverse machine learning models (e.g., collaborative filtering, content-based approaches, hybrids) for gaining greater insight into customer data and accordingly predicting their preferences more precisely ([Burke, 2002](#)). In addition, the results of this study are interpreted to apply directly to understanding Nike Plus's corporate performance. This approach can boost sales conversion rates and customer retention by making Nike Plus's product recommendations more relevant and trustworthy, which in turn strengthens customer engagement. ([Tiutiu & Dabija, 2023](#)) By offering accurate, personalized suggestions that truly align with individual preferences, Nike Plus can create a more enjoyable shopping experience and build long-term connections with its customers.

This project addresses a critical challenge: recommending and personalizing products for individual shoppers, which is one of the most important issues in today's retail industry. It will utilize collaborative filtering, content-based filtering techniques, hybrid recommendation approaches, and deep learning. The objective is to harness these advanced algorithms to develop a recommendation engine that significantly improves the accuracy of product suggestions for Nike Plus customers. This research not only aims for more accurate recommendations but also to elevate the user experience from start to finish. Personalized product suggestions—based on color, size, or style—are crucial for a brand's interaction with customers. The goal of this project is to enhance Nike Plus's customer experience and share

valuable insights with the broader retail industry. These findings could assist other companies in refining recommendation systems to make a similar impact on their customers.

1.1 RESEARCH MOTIVATIONS

The motivation driving this project, “A Comprehensive Recommendation System for Nike Plus Based on Size and Color”, arises from the critical need to improve the accuracy and personalization of product recommendations within the retail industry. Existing recommendation systems often fall short in fully capturing customer preferences, especially when it comes to specific attributes such as size and color. This project seeks to overcome these challenges by utilizing advanced data-driven techniques, such as content-based filtering. The primary objective is to develop a recommendation engine that enhances the shopping experience by delivering personalized product suggestions that closely match individual customer preferences. This demonstration aims to showcase the practical application of data science in addressing real-world challenges in personalized marketing, contributing valuable insights to both academic research and industry practices.

1.2 LIMITATIONS OF THE GENERAL RESEARCH

Despite advances in recommendation systems, significant gaps remain in fully understanding and modeling user behavior. Current models often oversimplify user preferences and struggle to adapt as these preferences evolve, limiting the personalization and relevance of recommendations. Bias and fairness issues also persist, as existing methodologies are not yet robust enough to detect and mitigate biases effectively, which can impact the equity of recommendations across diverse user groups.

Scalability and real-time processing present additional challenges, particularly in large-scale environments like e-commerce, where systems must handle vast amounts of data without compromising performance. The cold-start challenge persists because current models find it difficult to provide accurate suggestions for new users or products with limited data. Finally, the intricacy of sophisticated models can limit their interpretability, leaving users and stakeholders without clear explanations for recommendations, highlighting the necessity for more transparent and understandable systems.

1.3 AIM AND OBJECTIVES

While there have been several research efforts focused on improving recommendation systems, this project aims to develop a personalized product recommendation system for Nike Plus using content-based filtering techniques. The Aim of this project is to develop and evaluate a personalized product recommendation system for Nike Plus that leverages content-based filtering techniques to enhance customer engagement by providing relevant product suggestions based on specific attributes such as size and color.

In order to achieve this aim, a number of research objectives are addressed:

1. **Design a Content-Based Filtering Engine:** To design and implement a content-based recommendation system that can generate personalized Nike Plus product suggestions by matching user preferences with key product attributes.
2. **Algorithm Optimization and Mitigating Cold start issue:** To handle cold-start scenarios effectively, providing recommendations for new users or products with minimal interaction data by relying on product attributes rather than user behavior. Continuously fine-tune algorithms to enhance accuracy in reflecting customer size and color preferences for precise recommendations.
3. **Streamlit Deployment:** Deploy the optimized recommendation engine using Streamlit to create an interactive web application, enabling users to input preferences and receive personalized recommendations in real-time.
4. **User Feedback Integration:** Integrating a feedback mechanism to collect and use user input after interactions with the recommendation system, allowing continuous refinement and adaptation to real-time preferences. This ensures the system evolves, becoming more accurate and personalized over time, leading to improved customer satisfaction and business outcomes.

1.4 LEGAL, SOCIAL, ETHICAL AND PROFESSIONAL ISSUES

The creation and implementation of customized recommendation systems give rise to numerous important legal, social, ethical, and professional concerns that require thorough examination. In terms of the law, the system is required to adhere to data protection and privacy rules, like the General Data Protection Regulation (GDPR). This involves making sure that customer data is collected, stored, and used securely and transparently. Users need to be educated on the usage of their data and steps must be taken to ensure personal information is protected and anonymized. Not adhering to these rules may lead to legal consequences.

In terms of social impact, personalized recommendation systems may unintentionally strengthen biases or restrict access to a variety of products. For example, if the system always suggests products based on past user actions, it might unintentionally decrease the range of choices available to users, restricting their chances to discover new or diverse products. This could lead to a standardized shopping experience, restricting users to a limited range of suggestions from their previous choices, possibly hindering their ability to find new items and explore.

There is an ethical duty to make sure that the recommendation system does not misuse users' personal information or influence their preferences in a way that puts profit before user happiness. Taking into account morals involves creating a system that is just and clear, while also refraining from exploiting user actions solely for financial profit. It is important to also think about how personalization can affect consumer independence, making sure that users have power over their shopping experience and are not overly swayed by the system's suggestions.

From a professional point of view data scientists and developers working on these systems must follow industry standards like responsible data management, clear algorithms, and fairness. Continuous monitoring and evaluation of the recommendation system is crucial to

ensure it functions as planned, providing benefits to users without any negative consequences. This involves dealing with any problems concerning algorithmic bias, guaranteeing that the system stays just and inclusive for all users

1.5 THESIS OVERVIEW

The methodology for this project is depicted in Figure which outlines the typical data science pipeline employed. This comprehensive approach begins with data collection followed by data pre-processing. Next the feature engineering is conducted to extract relevant features from the dataset. The dataset contains 2,29,746 rows from Nike Plus UK 2022 sportswear. The initial phases involved data cleaning and pre-processing followed by EDA to understand the distributions of pattern within the data. Content-based filtering model was developed using cosine similarity and TFID vectors for recommending the items. The model was deployed using a Streamlit website for real time users.

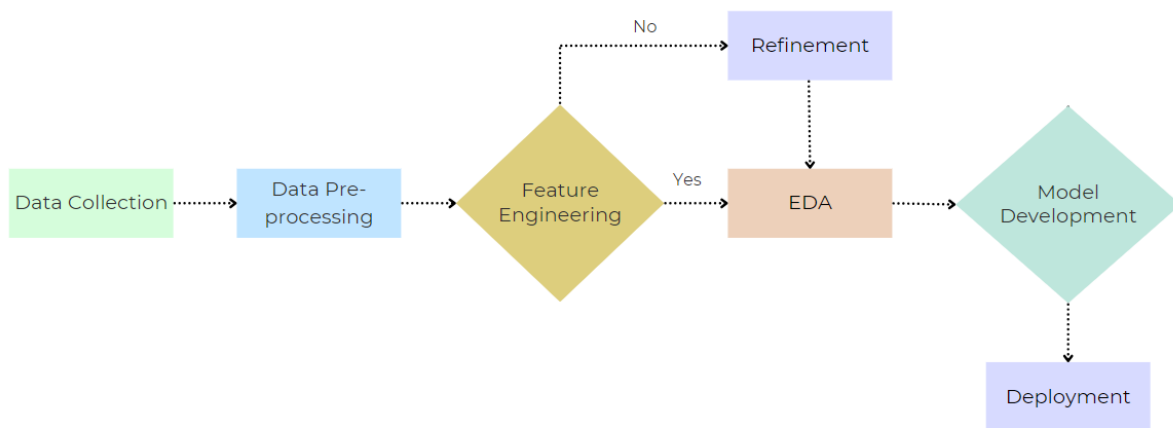


Figure 1. 1The Fundamental Concept of the Proposed Approach

1.6 THESIS STRUCTURE

The thesis is structured into eight chapters, each addressing a crucial component of the research. Chapter 1 introduces the research problem, outlining the aims, objectives, and significance of developing a personalized recommendation system for Nike Plus. Chapter 2 provides a comprehensive overview of the foundational theories and concepts related to recommendation systems, particularly focusing on the relevance of content-based filtering. Chapter 3 reviews existing literature, identifying gaps and challenges in current recommendation system methodologies and highlighting areas for potential improvement. Chapter 4 details the data collection, preparation, and preprocessing steps, including exploratory data analysis to uncover patterns and insights within the dataset. Chapter 5 describes the methodology, focusing on the design and implementation of the recommendation system using content-based filtering and the deployment of the system via Streamlit. Chapter

6 presents the results, including the performance metrics and user interface, demonstrating the system's effectiveness in providing personalized product recommendations. Chapter 7 discusses the results, evaluating the strengths and limitations of the approach and suggesting areas for improvement and future research. Finally, Chapter 8 concludes the thesis by summarizing the key findings, contributions, and challenges, while proposing future work to enhance the recommendation system further.

CHAPTER 2: BACKGROUND

2.1 BACKGROUND RESEARCH

In today's highly competitive e-commerce landscape, personalized product recommendations have become a crucial factor in enhancing customer experience and driving sales. For global brands like Nike Plus, the ability to suggest products tailored to individual preferences can significantly impact customer satisfaction and loyalty.

Recommendation systems play a vital role in online shopping by creating customized shopping experiences that are important for keeping customers satisfied and loyal. Through the examination of user habits and choices, these systems have the ability to recommend products tailored to the specific needs of each person, ultimately boosting the chances of a sale and improving the overall quality of the user's interaction. They also aid e-commerce platforms in effectively handling extensive product catalogs by highlighting items that may go unnoticed, thus increasing sales and opportunities for cross-selling. Furthermore, recommendation systems are crucial in enhancing customer involvement by constantly gaining insights from user interactions and adjusting to evolving preferences, guaranteeing that users always receive recommendations that align with their requirements. This helps businesses increase profits and develop deeper connections with clients

Recommendation systems are sophisticated technologies that assist in providing personalized suggestions for content, goods, or services to each user. These algorithms analyse information from previous purchases, surfing patterns, and other interactions to predict what a user might like. These systems can be constructed in a variety of methods, including content-based filtering, which focusses on the attributes of items a user has liked, collaborative filtering, which examines the preferences of similar users, and hybrid models that combine the two strategies ([Jannach et al., 2016](#)). By boosting interaction, personalising content, and boosting sales, they improve the user experience. Recommendation systems are employed by platforms such as Netflix and Amazon to propose films or products that viewers are likely to find enjoyable. This approach enhances customer retention and generates money ([Koren et al., 2009](#)). Recommendation systems also assist companies in optimising their marketing strategy by presenting pertinent offers to target customers

These systems have become integral to numerous industries, including ecommerce, streaming services, and social media platforms, where they enhance user experience, increase engagement, and drive sale ([Ricci et al., 2015](#)). By leveraging vast amounts of data, recommendation systems can analyze user behavior, preferences, and historical interactions to deliver highly personalized suggestions. These personalized experiences not only boost user satisfaction but also significantly contribute to customer retention and loyalty ([Aggarwal, 2016](#)).

In the e-commerce sector, for example, recommendation systems help customers discover products they might not have found on their own, thereby increasing the likelihood of purchase and enhancing the overall shopping experience. Amazon's recommendation engine, which generates a significant portion of its revenue, is a prime example of how effective these systems can be in driving sales ([Linden et al., 2003](#)). Similarly, streaming services like Netflix and Spotify utilize recommendation algorithms to curate personalized content for their users, thereby increasing user engagement and time spent on the platform ([Gomez-Uribe & Hunt, 2015](#)).

Recommendation systems are important in a number of other fields, such as internet advertising, social media, and even healthcare, in addition to suggesting goods or content. For example, in online advertising, these technologies assist in targeting adverts to the appropriate audience, increasing campaign effectiveness and cutting down on wasteful spending. Recommendation systems improve user experience and commercial outcomes by displaying adverts to people who are most likely to be interested based on their understanding of their preferences. In social media platforms, recommendation algorithms suggest friends, groups, or pages that users might be interested in, thus fostering a more connected and engaging user community. In healthcare, recommendation system scan assists in suggesting personalized treatment plans or health tips based on a patient's medical history and current health status.

2.2 UTILISED ALGORITHMS AND METRICS

2.2.1 COMMON MODELLING APPROACHES FOR RECOMMENDATON SYSTEM

1. Collaborative Filtering

Collaborative filtering is a technique that predicts what a user might enjoy by analyzing their past behavior—like their ratings, purchase history, and other interactions—and comparing it to the behavior of others with similar preferences. The idea is that users who have shown similar interests in the past are likely to agree on similar things in the future ([Schafer et al., 2007](#)).

There are two main types of collaborative filtering:

User-Based Collaborative Filtering: Recommend items to a user by returning other things liked by similar users. On the other hand, if User A and User B have similar preferences, and User A thumbs up to a product, it's likely that it will also appeal to User B. This method makes use of user data to recommend items accurately, but it requires a large dataset. As the number of users increases, managing this can become more challenging.

Item-Based Collaborative Filtering: Instead of comparing users, this algorithm computes item-item similarities. It recommends items that are similar to what a user liked previously. For example, if a customer likes one pair of shoes, the system will suggest other similar shoes. This approach is typically more scalable than user-based filtering as the total number of items grows quickly, and it requires fewer re-processing resources.

Figure 2.1 depicts the process of a collaborative filtering recommendation system. It shows how an active user is provided with recommendations based on the preferences and behaviours of users with similar tastes. The system collects information from a database about users' behaviour and tastes, identifies users with similar interests, and then suggests relevant subjects to the active user.

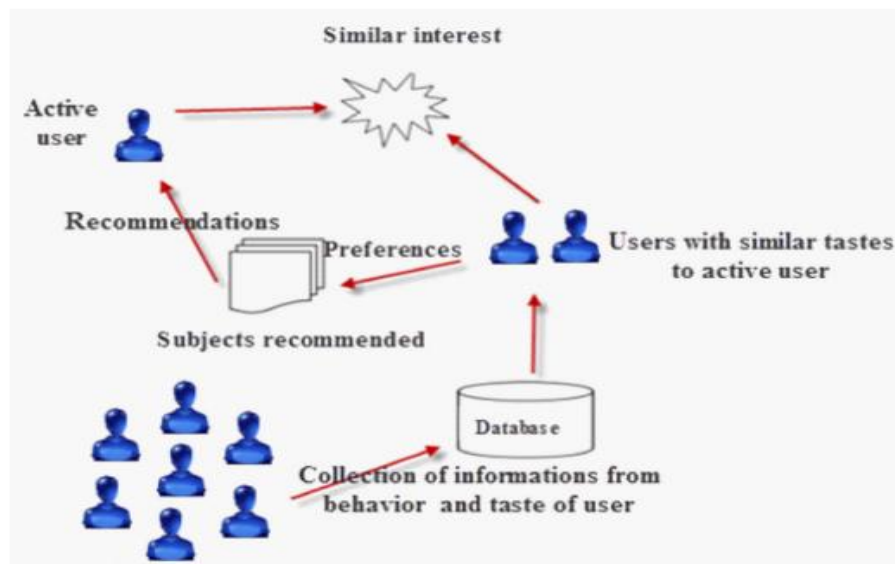


Figure 2. 1 Descriptive working of Collaborative Filtering

(Khanzadeh & Mahdavi, 2014)

2. Content-Based Filtering

Content-based filtering is a recommendation technique that suggests items to users based on the attributes of the items themselves and the user's historical preferences. This method operates on the premise that if a user liked an item with certain attributes, they are likely to prefer other items with similar attributes. This technique is particularly relevant for scenarios where item features are well-defined and crucial to user preferences, such as in recommending apparel based on size and color.

Content-based filtering is employed to provide personalized product recommendations for Nike Plus customers based on their preferred sizes and colors. By leveraging the TF-IDF vectorization technique, the system transforms these textual attributes into numerical vectors, enabling the calculation of similarity scores using cosine similarity.

Feature Extraction: When using content-based filtering, the initial step is to extract pertinent features. from the items. For example, in a recommendation system for clothing, features might

include size, color, material, and style. These features are typically represented as vectors in a multidimensional space, where each dimension corresponds to a specific attribute.

User Profile Creation: The system then creates a user profile based on the features of items the user has interacted with. This profile is essentially a vector that represents the user's preferences. For instance, if a user frequently buys blue, medium-sized shirts, the profile would reflect these preferences by assigning higher weights to the attributes 'blue' and 'medium-sized'.

Similarity Calculation: To generate recommendations, the system calculates the similarity between the user's profile and the features of available items. This is often done using cosine similarity, which measures the cosine of the angle between the user profile vector and the item feature vectors. Items with higher similarity scores are considered more relevant to the user's preferences and are thus recommended.

Figure 2.2 illustrates the architecture of a content-based filtering system. It shows how item descriptions from an information source are analyzed and represented, leading to the creation of user profiles. A user profile learner updates these profiles based on user feedback, and a filtering component uses the profiles to generate personalized recommendations for active users.

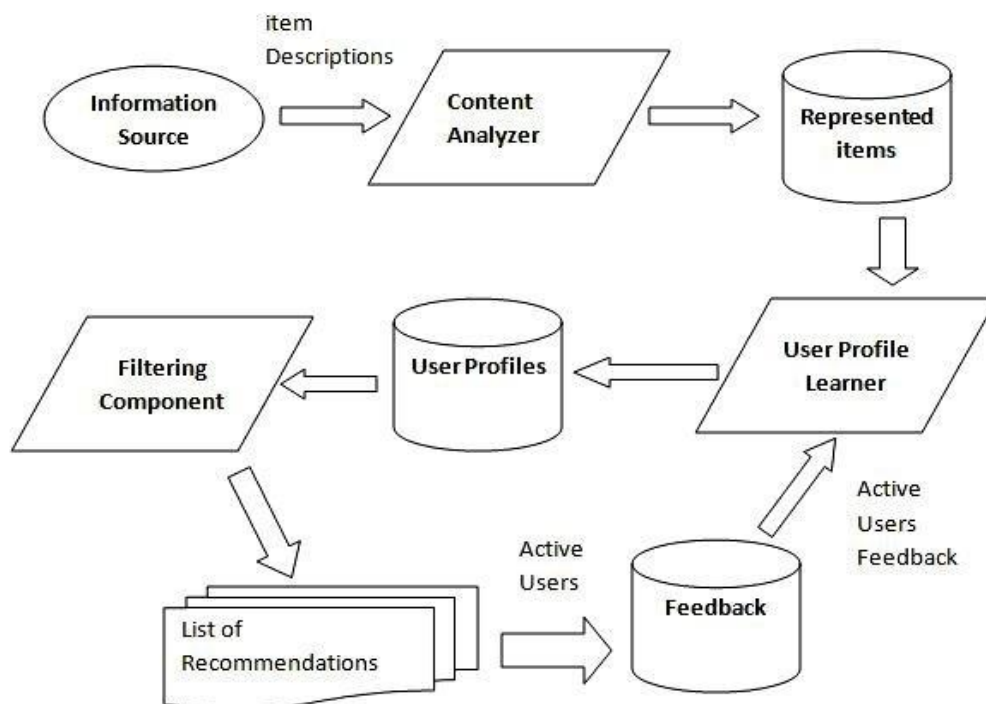


Figure 2. 2 Architecture for Content-Based Recommendation system

([Addagarla, 2019](#))

Figure 2.3 compares collaborative filtering and content-based filtering approaches. In collaborative filtering (left side), recommendations are made based on the preferences of similar users, suggesting items liked by others with similar tastes. In content-based filtering (right side), recommendations are generated based on the user's past behavior, suggesting items similar to those the user has previously interacted with.

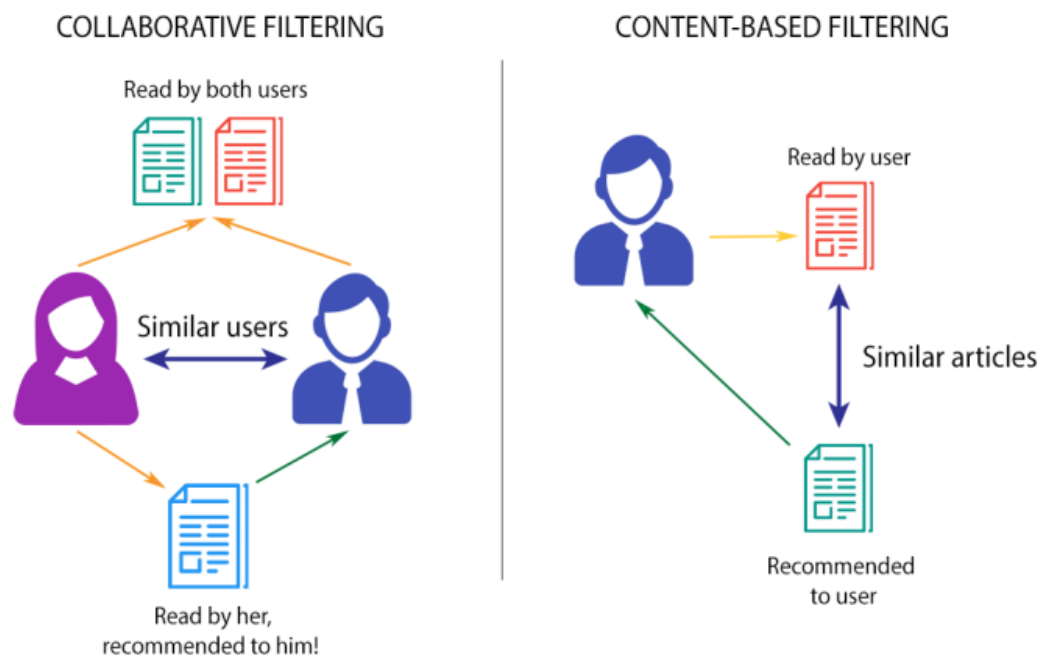


Figure 2. 3 Comparison of Collaborative Filtering and Content-Based Filtering in Recommendation Systems

3. Hybrid Approaches

To provide customised recommendations, hybrid recommendation engines combine content-based and collaborative filtering methods. By utilising this strategy, hybrid systems can gain from the advantages of both approaches, which is superior to employing just one. In a hybrid approach, content-based filtering refines these recommendations based on a user's particular interests, while collaborative filtering suggests popular things that many users have rated highly. Using content-based filtering to first create possible recommendations and then ranking or refining these suggestions based on user preferences is another popular hybrid strategy. As an alternative, content-based filtering is applied to more recent things with scant input, and collaborative filtering is employed for items with an abundance of interaction data.

Combining results: Combining the outcomes of content-based and collaborative filtering can help develop a hybrid recommendation system. By integrating the advantages of both approaches, this method enables the system to balance recommendations and provide a more varied range of choices.

Sequential Application: Using the result of one procedure as the input for the next is an additional strategy. For example, a system might use content-based filtering to further refine these results based on item qualities after using collaborative filtering to produce an initial set of recommendations.

Unified Models: A few hybrid systems integrate content-based and collaborative concepts into a single strategy. To improve their predictive power, these frequently apply machine learning approaches, getting over the drawbacks of relying just on one approach.

Figure 2.4 depicts a hybrid recommendation system that combines content-based and collaborative filtering models. It shows how both models generate predictions based on document vectors (doc2vec) and user similarity, respectively. The system then evaluates the performance of these models and merges their predictions to create a ranked list of recommendations, supervised by a recommendation supervisor

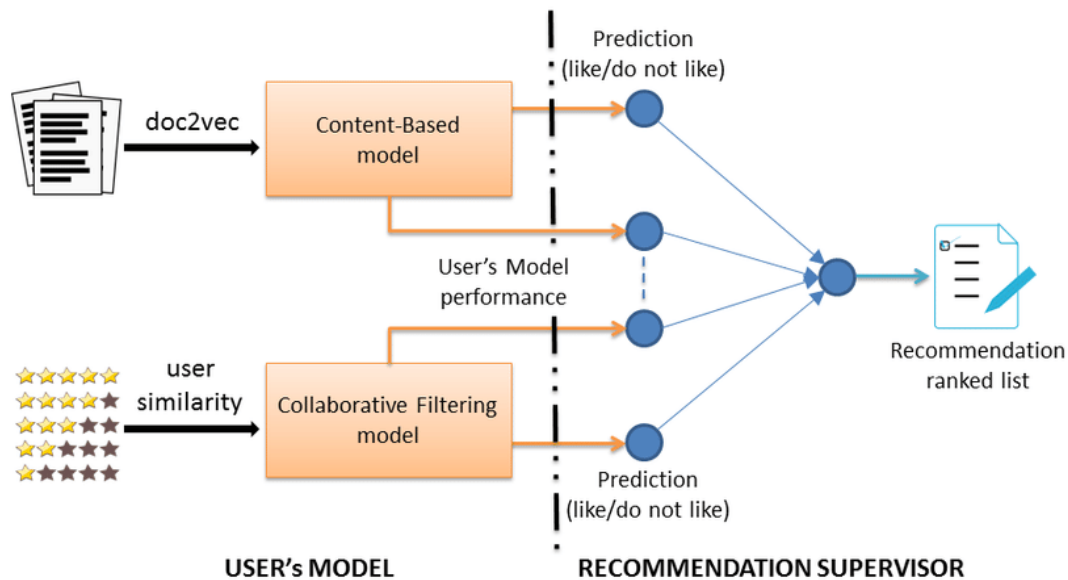


Figure 2. 4 Architecture of Hybrid Recommendation Model

([Sottocornola et al., 2017](#))

4. Matrix Factorization

Matrix factorization is a popular technique in collaborative filtering, particularly effective with large datasets. This method decomposes the user-item interaction matrix into lower-dimensional matrices, helping the model uncover latent features that explain user preferences. For instance, in a movie recommendation system, matrix factorization can reveal a user's preference for certain actors, genres, or directors, influencing their movie choices. Singular

Value Decomposition (commonly known as SVD) is a matrix factorization technique used by Netflix in its recommendations. Matrix factorization works well with sparse data and offers highly personalized recommendations. However, it requires a significant amount of data for effective training and can be computationally intensive for very large datasets.

5. Deep Learning Based Models

Lately, deep learning has gained traction as a preferred method for addressing recommendation challenges, particularly when dealing with intricate and extensive datasets. Neural networks, such as deep learning models, are more appropriate for analyzing vast quantities of data and identifying subtle correlations between user preferences and item characteristics. They are able to work with different types of data formats, such as structured data, text, images, or audio, allowing them to be flexible for a variety of recommendation tasks. Deep learning systems frequently incorporate numerous layers of processing to identify advanced features within the data, resulting in more comprehensive and precise recommendations. For example, a deep learning algorithm may examine a user's internet activity, previous buys, and sentiment in product evaluations to anticipate their future preferences. Even though deep learning models can achieve excellent results in recommendation systems, they demand a high amount of computational power and are more intricate to create compared to conventional approaches such as collaborative or content-based filtering.

2.2.2 PERFORMANCE METRICS

1. Precision

Precision is the ratio of correctly recommended relevant items to the total number of items recommended by the system. It answers the question: "Of all the products the system recommended, how many were actually relevant to the user?" A precision score of 1.000000, is seen in the results, means that every product recommended by the system was indeed relevant to the user, with no irrelevant items being suggested.

$$\text{Precision} = \frac{\text{Number of relevant items recommended}}{\text{Total number of items recommended}}$$

2. Recall

Recall is the ratio of correctly recommended relevant items to the total number of relevant items that exist in the user's preference set. It answers the question: "Of all the relevant products available, how many did the system successfully recommend?" A recall value of 0.200000 indicates that the system only recommended 20% of the total relevant products available for the user. While the recommendations made were precise, the system missed out on recommending a large number of other relevant items that could have matched the user's preferences.

$$\text{Recall} = \frac{\text{Number of relevant items recommended}}{\text{Total number of relevant items available}}$$

3. F1-score

The F1 Score is the harmonic mean of precision and recall, providing a balance between the two. It is useful when you need to find an optimal balance between precision and recall, especially in situations where you need to weigh false positives and false negatives equally. The F1 Score 0.333333, reflecting a balance between the high precision and low recall. This means that while the system is very good at recommending relevant items, it doesn't recommend enough of them, leading to a moderate F1 Score.

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

4. Mean average precision (MAP)

MAP is an evaluation metric that takes into account the ranking of the relevant items in the list of recommendations. It averages the precision values at the positions of each relevant item in the ranked list. MAP provides a single metric that summarizes how well the system ranks relevant items. A MAP score of 1.000000 suggests that the recommendation system ranks all relevant products at the top of the list perfectly, ensuring that the user sees the most relevant items first.

$$\text{MAP} = \frac{1}{|Q|} \sum_{q=1}^{|Q|} \text{Average Precision}(q)$$

where $|Q|$ is the number of queries (or users), and $\text{Average Precision}(q)$ is the average precision for query q .

CHAPTER 3: LITERATURE REVIEW

3.1 RELATED WORKS

The development of recommendation systems has emerged as a critical area of research within machine learning and data science, particularly in the retail sector. As highlighted by ([Linden et al., 2003](#)), the primary objective of these systems is to offer personalized product recommendations to users, which can significantly enhance the overall shopping experience and drive sales. Their work on Amazon.com's item-to-item collaborative filtering illustrates how traditional approaches like collaborative filtering and content-based filtering have been widely adopted in the industry. However, the study acknowledges that these conventional methods often face challenges in accurately predicting customer preferences for more specific attributes, such as size and color.

According to ([Sarwar et al. 2001](#)), collaborative filtering fundamentally relies on how users interact with items, using techniques like matrix factorization or neighbourhood-based methods to predict user preferences. However, as reported by ([Schein et al. 2002](#)), one major limitation of these methods is the cold-start problem—when there is little data on new users or items, making it difficult to provide accurate recommendations. In contrast, ([Pazzani and Billsus, 2007](#)) highlight that content-based filtering offers an alternative by focusing on item features to recommend products similar to those a user has shown interest in previously. While this approach can alleviate the cold-start issue by relying on the attributes of items, it may struggle to account for the diverse and evolving nature of user preferences.

([Burke, 2002](#)) discusses the potential of hybrid approaches that combine multiple algorithms, which have been shown to address some of the limitations found in traditional recommendation systems. These hybrid models can leverage the strengths of different methods to provide more accurate recommendations. ([Zhang et al. 2019](#)) further emphasize the growing role of deep learning models in this area, noting that they offer significant improvements in recommendation precision, particularly when dealing with complex product attributes.

In the fashion and apparel industry, specific attributes like size and color play a crucial role in shaping consumer choices. However, traditional recommendation systems often fail to prioritize these details, instead focusing on broader categories or general user ratings. Recent research, such as that highlighted by the study on Amazon's recommendation systems, underscores the importance of incorporating detailed attribute information, like size and color, into recommendation algorithms to enhance the accuracy and relevance of the suggestions provided to users.

([Essinger et al., 2021](#)) introduced the AIR system as part of Nike Plus's digital transformation, which integrates various machine learning models to deliver personalized product recommendations. This system marks a significant advancement in addressing the challenges associated with providing detailed attribute-based recommendations within the athletic apparel industry.

However, as ([Lamers, 2023](#)) points out, there is still a noticeable gap in the literature concerning the optimization of machine learning algorithms specifically for detailed attributes like size and color in the context of athletic wear. Lamers emphasizes the need for more granular approaches to product recommendations within the sportswear sector, highlighting the potential positive impact on customer satisfaction and sales conversion rates. Additionally, incorporating artificial intelligence (AI) into recommendation systems presents fresh opportunities and obstacles, highlighting the growing importance of AI in diverse sectors such as e-commerce. The intended system seeks to improve personalized recommendations by utilizing sophisticated machine learning methods, with a specific emphasis on characteristics like size and color. In doing this, it aims to help advance the field of personalized marketing strategies in the athletic apparel e-commerce industry.

3.2 KNOWLEDGE GAP

Despite the significant progress made in recommendation systems, there remains a critical gap in effectively tailoring recommendations based on granular attributes such as size and color, especially within the apparel industry. ([Desrosiers and Karypis, 2010](#)) note that traditional recommendation algorithms, particularly those relying on collaborative filtering, often struggle to address these specific preferences because of their heavy reliance on user-item interaction data. Although content-based filtering approaches, as discussed by ([Ricci et al., 2015](#)), have the potential to incorporate item attributes, they frequently fall short in capturing the more nuanced preferences of users, such as specific size and color preferences, which are essential in fashion retail. This gap highlights the need for more sophisticated models that can account for these detailed attributes to better meet the demands of consumers in the fashion and apparel sector.

Moreover, existing hybrid systems, which combine collaborative and content-based methods, still face significant challenges in adequately integrating visual and textual data, resulting in a less personalized shopping experience. As ([Adomavicius and Tuzhilin, 2005](#)) suggest, many of these systems lack the ability to process detailed product descriptions and images in tandem, a capability that is crucial for accurately predicting user preferences, particularly in the fashion industry. This shortcoming is further compounded by the cold-start problem, where the absence of sufficient historical data for new users or products diminishes the effectiveness of the recommendations provided. This highlights the need for more advanced hybrid models that can better integrate diverse data types and address the limitations currently present in the recommendation systems landscape.

Deep learning models have demonstrated considerable promise in enhancing recommendation systems, yet their potential for predicting multi-attribute preferences, such as size and color, remains largely underexplored. ([Kalantidis et al., 2013](#)) points out that while some studies have utilized convolutional neural networks (CNNs) to analyze product images, these efforts have not fully integrated visual features with other crucial product attributes. This gap indicates a

pressing need to develop recommendation systems that incorporate detailed user preferences, enabling more accurate and relevant product suggestions, particularly in domains where attributes like size and color play a critical role in consumer decision-making.

In summary, while existing recommendation systems have achieved notable progress, there is still a substantial gap in accurately predicting consumer preferences based on complex and nuanced attributes like size and color. Addressing this gap requires a more sophisticated approach that integrates various data types and leverages state-of-the-art machine learning models to deliver truly personalized recommendations. The current project aims to fill this void by developing a comprehensive system tailored to the unique demands of the fashion retail sector.

CHAPTER 4: DATA

4.1 RETRIEVING THE DATA

The initial phase of Methodology was to gather data, for which I utilized a Kaggle dataset. This dataset includes key attributes needed to create a recommendation system for Nike Plus products. It contains various characteristics of Nike Plus products that can be utilized to generate customized suggestions according to size and color.

The dataset contains key columns like DEPARTMENT, CATEGORY, SUBCATEGORY, and PRODUCT_TYPE to categorize products and tailor suggestions according to customer preferences. SKU and SKU_VARIANT serve as distinct identification markers for products and their different versions, offering a detailed level of specificity for distinguishing between products. The fields of PRODUCT_NAME and PRODUCT_ID, provide detailed information about each item, and PRODUCT_URL allows users to easily navigate to the online store for smooth purchasing. The size of the product is an important factor for the recommendation system, enabling personalized suggestions based on customers' size preferences. Likewise, the COLOR and COLOR_CODE columns play a crucial role in filtering products by color, which is a main priority of the recommendation engine. Additionally, PRICE_CURRENT and PRICE_RETAIL offer information on product pricing trends that can help recommend items that fit the customer's budget, while IS_BESTSELLER can showcase popular products.

In general, this dataset provides a thorough understanding of Nike Plus's product range, allowing for the development of a strong recommendation system that can propose items according to size, color, and other preferences, ultimately improving the user experience.

4.2 DATA PREPARATION & PREPROCESSING

Data preparation and preprocessing are critical steps in building a robust recommendation system. In this project, after retrieving the dataset, I began by thoroughly cleaning the data to ensure its quality and integrity. This process involved removing all null values, which could potentially disrupt the accuracy of the recommendation engine. By eliminating these missing data points, I ensured that every record in the dataset was complete, making the analysis more reliable. Next, I focused on organizing the data. This involved structuring and formatting the dataset in a consistent manner, which helped in reducing complexities during the modelling phase. I also deleted unwanted columns that were not necessary for the recommendation system, such as metadata columns and redundant attributes, to address memory usage issues. This optimization was important, particularly given the large size of the dataset, as it improved processing speed and efficiency.

Additionally, I handled correlations among variables to prevent multicollinearity, which can negatively affect the performance of the recommendation model. By identifying and managing

highly correlated variables, I ensured that the model would not be biased or skewed by redundant information. This preprocessing step also involved normalizing and encoding categorical data, preparing it for machine learning algorithms that require numerical inputs. Overall, the data preparation and preprocessing phase laid the foundation for a high-quality, efficient recommendation system by ensuring that the data was clean, organized, and free from issues that could hinder model performance.

Preprocessing steps to prepare the data:

- **Handling Product Size**

The `PRODUCT_SIZE` column contains categorical values representing product sizes (e.g., "S", "M", "L", or numerical shoe sizes). These categorical sizes were converted into numerical representations using a `LabelEncoder`, ensuring the model can process and compare sizes effectively.

- **Vectorizing Color:**

The `COLOR` column represents the color of the product. To handle this textual data, the color names were vectorized using the `CountVectorizer` technique, which converts the color names into a numerical format suitable for machine learning models. This process generates a sparse matrix where each row corresponds to a product and each column represents the presence of a particular color.

- **Encoding Product Type:**

The `PRODUCT_TYPE` column categorizes the products into various types, such as footwear and apparel. These categorical values were also converted into numerical representations using a `LabelEncoder`.

- **Normalizing Price:**

The `PRICE_CURRENT` column contains the current price of each product. To ensure that price values are on a comparable scale, they were normalized using a `StandardScaler`. This normalization process standardizes the prices by removing the mean and scaling to unit variance, which helps improve the performance of the recommendation algorithm.

- **Combining Features:**

After preprocessing individual features, the numerical representations of product size, color, product type, and normalized price were combined into a single feature vector for each product. The `np.hstack()` function was used to horizontally stack the arrays of features, forming a complete feature vector that the recommendation system can use to calculate product similarities.

4.3 RATIONALE TOWARDS EXPLORING THE DATA

4.3.1 EXPLORATORY DATA ANALYSIS (EDA)

EDA plays a crucial role in understanding the characteristics of the dataset and identifying patterns or relationships among variables. Exploratory Data Analysis (EDA) is crucial for this project because it enables the discovery of important patterns and insights in the data that will enhance the performance of the recommendation engine. Exploratory Data Analysis (EDA) is a critical step in any data science project as it helps uncover underlying patterns, relationships, and anomalies within the dataset. By visualizing and summarizing data, EDA allows for a deeper understanding of the data's structure and key features, enabling more informed decision-making during the model development process. It also aids in identifying potential issues, such as missing values or outliers, that could impact the model's performance. Furthermore, EDA provides insights that can guide feature selection and engineering, ultimately leading to more accurate and robust models. In the end, EDA guides the development of better recommendation algorithms and guarantees that the system can adjust to the various and intricate requirements of Nike Plus's broad customer base. By establishing a solid analytical foundation, the project can effectively tackle the obstacles of providing tailored recommendations, ultimately improving customer satisfaction and increasing sales conversion rates.

Figure 4.1 (Pie chart) represents the distribution of the top 10 best-selling Nike Plus products. Nike Plus Sportswear leads the chart with 31.8% of sales, followed by Nike Plus Dri-FIT and Nike Plus Sportswear Club Fleece with 12.4% and 8.9% respectively. The chart highlights the dominance of certain product lines within the overall sales, giving insight into which products are the most popular among consumers.

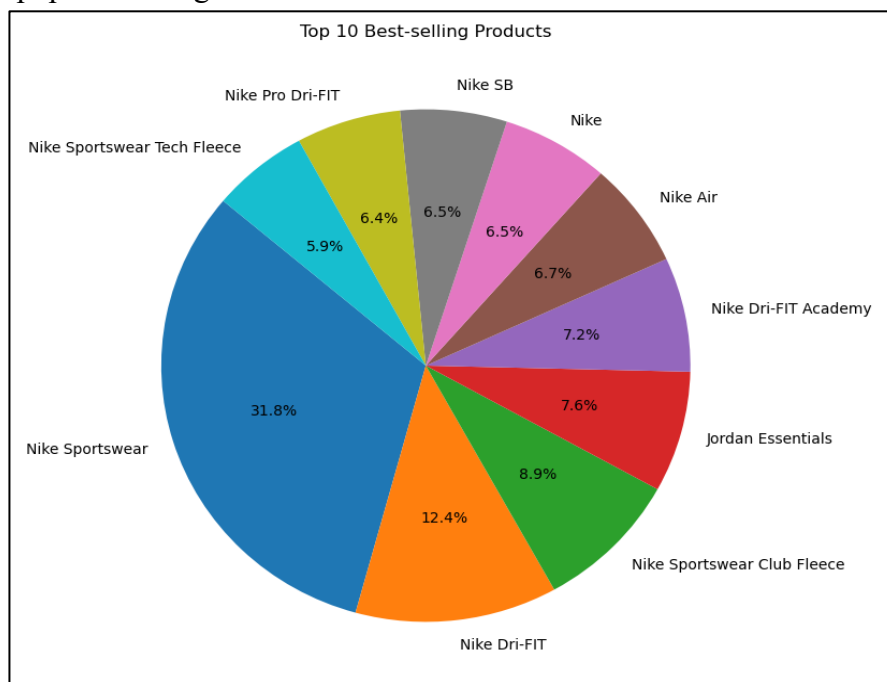


Figure 4.1 Distribution of Top 10 Best-Selling Products

Figure 4.2 (Bar chart) illustrates the distribution of product sales across different departments: Men, Women, and Kids. The Men's department has the highest count, indicating that most of the sales are from this category, followed by the Women's and Kids' departments. This distribution helps in understanding the target demographic that drives the majority of Nike Plus's product sales.

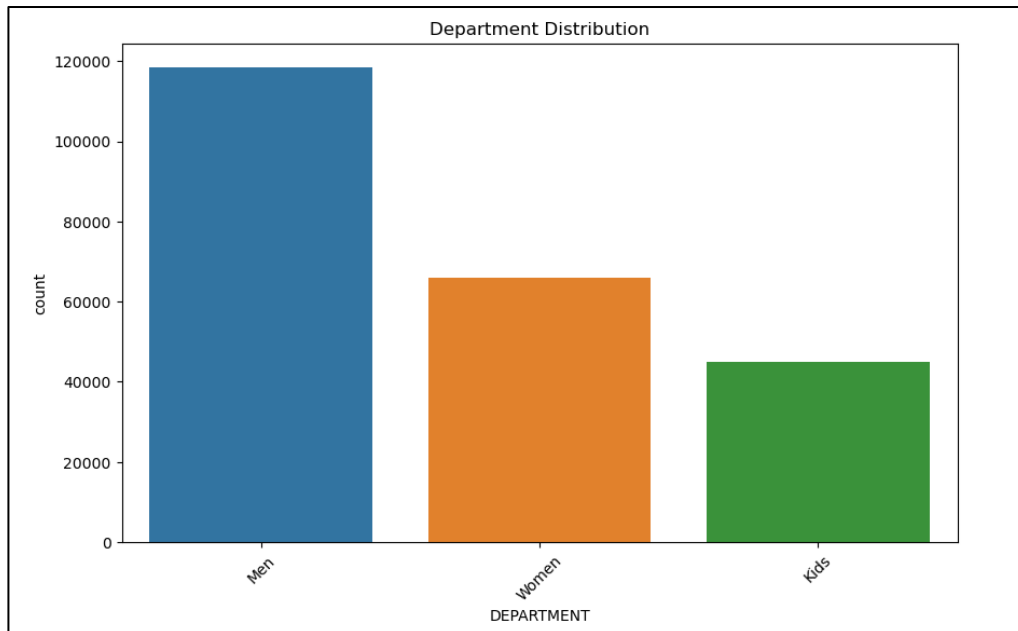


Figure 4. 2 Sales Distribution by Department

Figure 4.3 (Bar chart) illustrates the top 10 most popular color combinations in Nike Plus's product sales. "Black/White" is the most preferred color combination, followed by "Black" and various "multi-colour" options. This chart provides valuable insights into customer color preferences, which can be crucial for inventory management and marketing strategies.

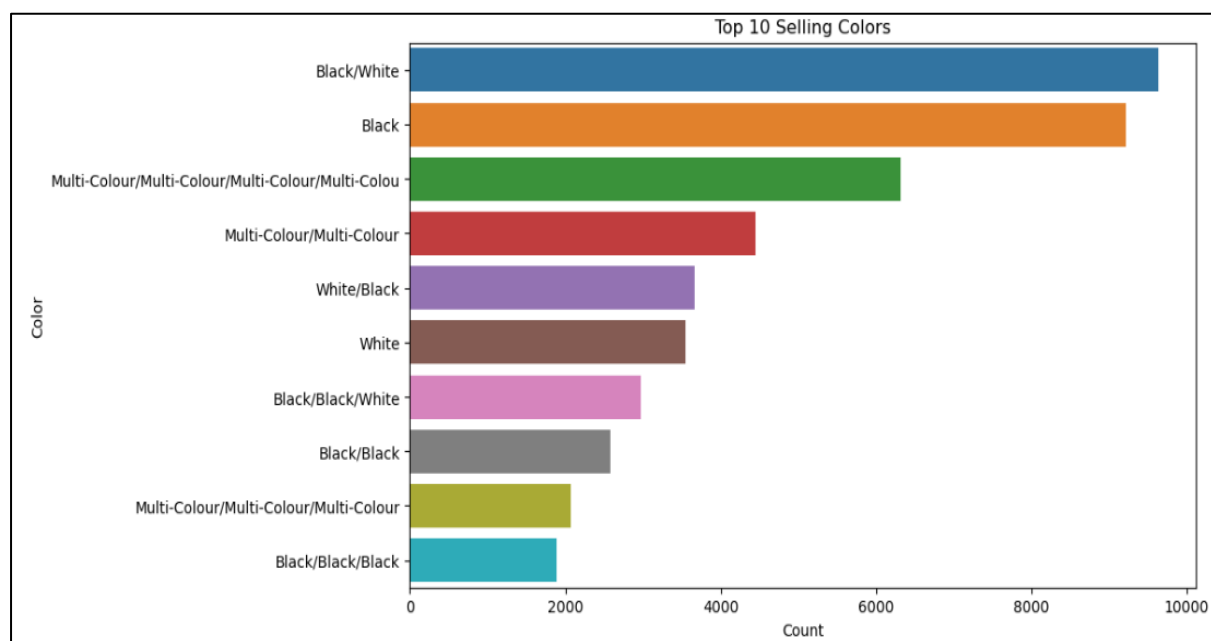


Figure 4. 3 Top 10 Best-Selling Color Combinations

Figure 4.4 (Bar chart) illustrates the top 10 most popular product sizes. "L" (Large) is the most sold size, followed closely by "M" (Medium) and "S" (Small). The chart reveals the most commonly purchased sizes, which can be crucial for inventory management, ensuring that popular sizes are adequately stocked to meet demand.

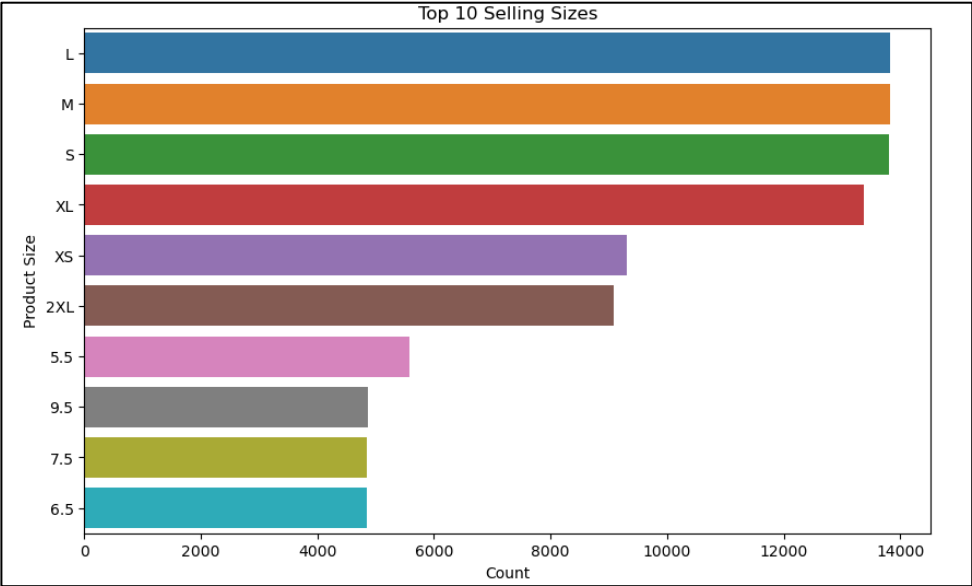


Figure 4. 4 Top 10 Best-Selling Product Sizes

Figure 4.5 (Heatmap) displays the correlation between various numerical variables in the dataset, such as SKU, SKU Variant, Product ID, and pricing information. The intensity of the color indicates the strength of the correlation, with darker shades of red representing higher positive correlations. The matrix helps identify relationships between different variables, which can be useful for feature selection and understanding how different factors influence each other.

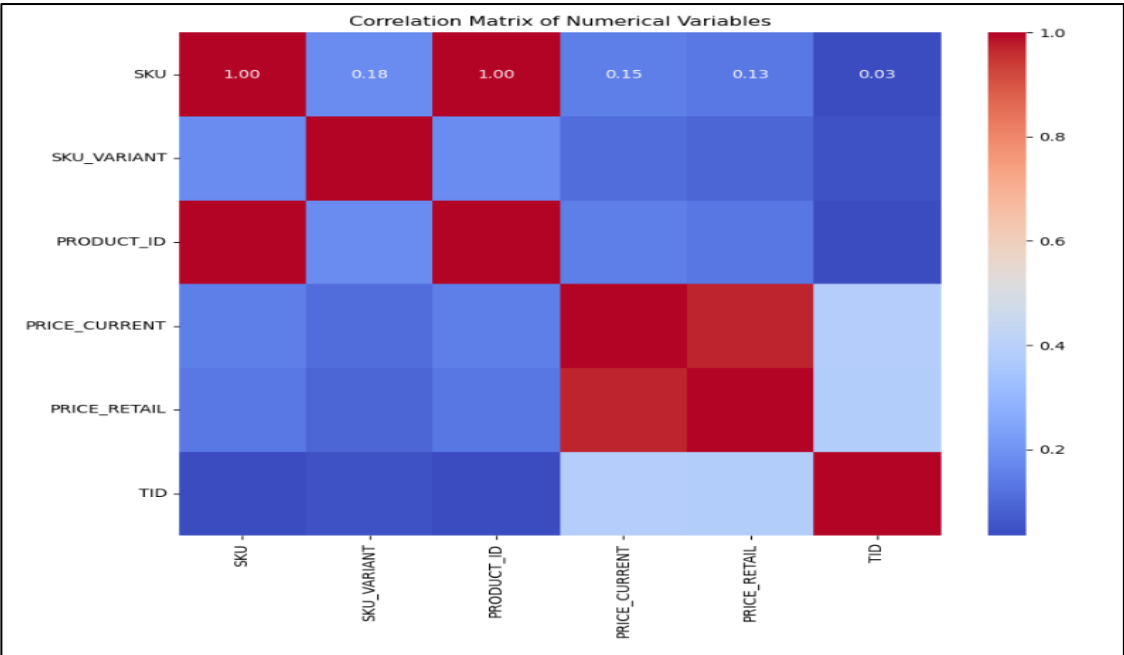


Figure 4. 5 Correlation Matrix of Numerical Product Variables

4.3.2 FEATURE ENGINEERING

Feature engineering plays a pivotal role in the development of an effective recommendation system, particularly in a project like mine that focuses on delivering highly personalized product suggestions based on detailed attributes such as size and color. The process of feature engineering involves transforming raw data into meaningful representations that machine learning algorithms can interpret and learn from. In this project, various techniques are employed to encode categorical variables like product size, color, and type into numerical representations that capture the essence of these attributes while maintaining their underlying structure.

For instance, sizes are encoded into numerical values using label encoding, allowing the recommendation system to understand and process size variations effectively. Colors, another critical attribute in the recommendation process, are vectorized to create a feature space that represents different shades and hues. This enables the system to factor in color preferences when generating recommendations. Additionally, product types are encoded similarly to facilitate comparison across different categories of Nike Plus products. Finally, the current price of each product is normalized to ensure that the machine learning model can focus on relative differences in price without being skewed by absolute values.

By combining these engineered features into a single feature vector, the recommendation system can leverage all relevant product attributes (size, color, type, and price) to generate more accurate and personalized suggestions for customers. This comprehensive approach to feature engineering ensures that the recommendation engine is well-equipped to handle the complexity of customer preferences, leading to improved recommendation accuracy and, ultimately, enhanced customer satisfaction and increased sales conversion rates.

CHAPTER 5: METHODOLOGY

Figure 5.1 illustrates the methodology for developing the recommendation system for Nike Plus products, which focuses on size and color preferences, is depicted in the Figure. The process starts with gathering data, which was obtained from Kaggle, including important attributes such as size, color, price, and availability. Once the data is gathered, the following step is Data Integration to merge data from various sources if necessary, then Data Preparation, which includes organizing and arranging the data for examination. Data Preprocessing guarantees data integrity by addressing missing values, eliminating unnecessary columns for better memory efficiency, and converting categorical variables to prepare the data for modeling.

Next, the project progresses to Feature Engineering, which involves enhancing current features, generating new features as needed, and dealing with multicollinearity to avoid potential problems caused by highly correlated features. This stage includes a repetitive procedure of assessment and improvement to guarantee that the characteristics are enhanced for the recommendation system. After feature engineering, Descriptive Statistics and Data Visualization are used to delve deeper into the dataset, recognizing patterns and correlations among variables, guiding decisions in model development. During the Model Development stage, three recommendation models are taken into account: Collaborative Filtering, Content-Based Filtering, and Hybrid Models. Exploring each of these methods to identify the most effective approach for producing customized recommendations. Once the models are developed, Model Evaluation is carried out using different metrics to evaluate their performance. The approach also highlights the importance of Model Interpretability, making sure the outcomes are clear and easy to comprehend.

Ultimately, the suggestion algorithm is implemented via a Streamlit Website, allowing users to easily access customized Nike Plus product recommendations. This stage of deployment signifies the end of the project, guaranteeing the system is operational and easy to use.

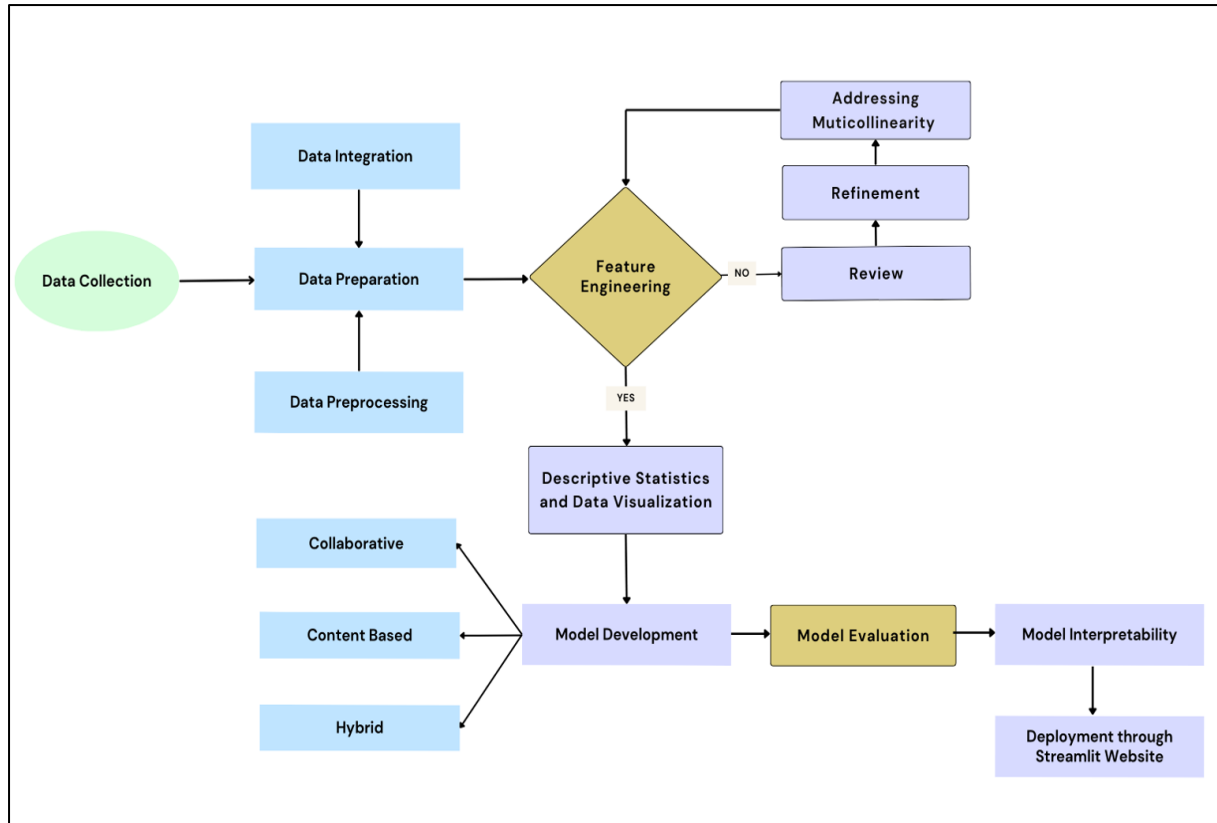


Figure 5.1 Proposed approach

5.1 CONTENT BASED FILTERING USING COSINE SIMILARITY

Why Use Cosine Similarity in Recommendation Systems?

In recommendation systems, cosine similarity is particularly useful when working with high-dimensional data, such as product attributes, where each attribute (e.g., color, size, category) can be considered as a dimension in the feature space. Unlike Euclidean distance, which measures the absolute difference between vectors, cosine similarity focuses on the direction of the vectors, making it more suitable for comparing items based on their attributes, even if they have different magnitudes.

In my content-based recommendation system, cosine similarity allows me to compare products based on their feature vectors, which represent various attributes such as size, color, product type, and price. By calculating the cosine similarity between a selected product and all other products in the dataset, I can identify which products are most similar and recommend them to the user.

5.1.1 THEORY

Cosine Similarity is a measure used to determine the similarity between two vectors, without considering their size. It is commonly utilized in text analysis, recommendation systems, and

other machine learning tasks for determining the similarity between items represented as vectors. Cosine similarity is employed in project to compare feature vectors of various Nike Plus products for determining the most similar ones. Cosine similarity is used to compare the combined features of products (size and color) to find similar items. By calculating the cosine similarity between the vectors representing different products, the system can identify items that are most similar to a given product based on these attributes.

Cosine similarity measures the cosine of the angle between two non-zero vectors in a multidimensional space.

$$\text{CosineSimilarity} = \frac{A \cdot B}{\|A\| \|B\|}$$

Where:

- $A \cdot B$ is the dot product of vectors A and B
- $\|A\|$ and $\|B\|$ are the magnitudes (or Euclidean norms) of vectors A and B .

•

The resulting value will range between -1 and 1:

- A cosine similarity of 1 means that the vectors are identical (i.e., the angle between them is 0 degrees).
- A cosine similarity of 0 means that the vectors are orthogonal (i.e., the angle between them is 90 degrees, indicating no similarity).
- A cosine similarity of -1 indicates that the vectors are diametrically opposed (i.e., the angle between them is 180 degrees).

How it Works:

- **Vector Representation:** Each item (or document) is represented as a vector in a multi-dimensional space. The dimensions correspond to the features of the items. For text data, these features are often words or terms extracted from the documents.
- **Dot Product and Magnitude:** Cosine similarity uses the dot product of the two vectors and their magnitudes. The dot product measures the extent to which the vectors point in the same direction, while the magnitudes normalize the vectors, ensuring the comparison is independent of their length.
- **Angle Calculation:** The cosine of the angle between the two vectors is calculated, resulting in a value between -1 and 1. A cosine similarity of 1 indicates that the vectors are identical, 0 indicates orthogonality (no similarity), and -1 indicates complete dissimilarity

5.1.2 IMPLEMENTATION

Cosine similarity is used to calculate the similarity between products after converting their attributes into feature vectors. The implementation in Python is done using the 'cosine_similarity' function from the 'sklearn.metrics.pairwise' module.

Feature Vector Creation

```
# Combine all features into a single feature vector
size_scaled = scaler.fit_transform(relevant_data[['SIZE_ENCODED']])
additional_features = relevant_data[['TYPE_ENCODED', 'PRICE_NORMALIZED']]
additional_feature_vectors = scaler.fit_transform(additional_features)
feature_vectors = np.hstack((size_scaled, color_vectors, additional_feature_vectors))
```

After preprocessing the product data (e.g., encoding sizes, vectorizing colors, and normalizing prices), all these features are combined into a single feature vector for each product. These feature vectors are then used for similarity calculation.

Cosine Similarity Calculation

```
from sklearn.metrics.pairwise import cosine_similarity

# Calculate cosine similarity between products
similarities = cosine_similarity([feature_vectors[product_index]], feature_vectors)[0]
```

Once the feature vectors are ready, cosine similarity is calculated between the vector of the selected product and all other products in the dataset. The `cosine_similarity` function takes two sets of vectors as input and returns a matrix of similarity scores.

5.1.3 COSINE SIMILARITY ALGORITHM FOR RECOMMENDATION

- **Input:**
 - A dataset of products with attributes (e.g., size, color, type, price).
 - A user's input specifying their preferred product, size, category, and color.
- **Preprocessing:**
 - Convert categorical attributes (e.g., size, color, type) to numerical representations.
 - Normalize numerical attributes (e.g., price).
 - Combine all features into a single feature vector for each product.
- **Cosine Similarity Calculation:**
 - For the user-selected product, retrieve its feature vector.
 - Calculate cosine similarity between this feature vector and the feature vectors of all other products.
- **Recommendation Generation:**
 - Sort the products based on similarity scores.

Exclude the selected product itself and any duplicate product names from the recommendations.

Select the top N products with the highest similarity scores.

- **Output:**

Return the recommended products to the user.

5.2 TF-IDF (TERM FREQUENCY-INVERSE DOCUMENT FREQUENCY)

Why Use TF-IDF in Recommendation Systems?

In recommendation systems, TF-IDF is used to convert textual features, such as product descriptions, into numerical vectors. These vectors can then be used to calculate the similarity between different products, enabling content-based recommendations.

For example, in this Nike Plus product recommendation system, TF-IDF is used to vectorize textual attributes like product descriptions or colors. By representing each product as a TF-IDF vector, the recommendation system can compare products based on the similarity of their textual content.

5.2.1 THEORY

TF-IDF is a statistical metric employed to assess the significance of a word in a document compared to a group of documents (corpus). Text data is often transformed into numerical features in natural language processing (NLP) for different machine learning purposes, such as recommendation systems.

TF-IDF has been utilized in project to analyze text data, such as product descriptions, color, and type attributes. The objective is to transform the textual characteristics into vectors that show the significance of each term, enabling the comparison of products based on similarities.

TF-IDF is a combination of two statistics:

1. **Term Frequency (TF) :**

Measures how frequently a term appears in a document relative to the total number of terms in that document. It is calculated as:

$$TF(t, d) = \frac{\text{Number of times term } t \text{ appears in document } d}{\text{Total number of terms in document } d}$$

2. Inverse Document Frequency (IDF):

Measures how important a term is across the entire corpus. If a term appears in many documents, its importance is reduced. It is calculated as:

$$IDF(t, D) = \log \left(\frac{\text{Total number of documents}}{\text{Number of documents containing term } t} \right)$$

The TF-IDF score for a term t in a document d within a corpus D is the product of its TF and IDF values:

$$TF - IDF(t, d, D) = TF(t, d) \times IDF(t, D)$$

The higher the TF-IDF score, the more important the term is in that particular document relative to the entire corpus.

How it Works:

- **Term Frequency (TF):** This component is used to measure how often a term appears in a document. The assumption is that terms appearing more frequently within a document are more important. However, this measure does not account for the importance of terms across the entire corpus.
Example: In a document about sports, the term "football" might appear several times, indicating its importance within that document.
- **Inverse Document Frequency (IDF):** This component measures how important a term is across the entire collection of documents. It assigns higher weight to terms that are unique to fewer documents and lower weight to common terms that appear in many documents.
Example: The term "the" is common across many documents and thus has a low IDF score, whereas a term like "regression" might appear in fewer documents and thus have a higher IDF score.
- **TF-IDF Score:** The TF-IDF score for a term is calculated by multiplying its TF score with its IDF score. This score reflects the importance of the term in a specific document relative to the entire corpus.
Example: The term "football" may have a high TF-IDF score in a document about sports because it is frequently mentioned (high TF) and not as common across all documents (high IDF).

5.2.2 IMPLEMENTATION

Import the Required Libraries

First, you need to import the necessary libraries, including the 'TfidfVectorizer' from 'sklearn.feature_extraction.text'

Combining TF-IDF Vectors with Other Features

```
# Combine all features into a single feature vector
size_scaled = scaler.fit_transform(relevant_data[['SIZE_ENCODED']])
additional_features = relevant_data[['TYPE_ENCODED', 'PRICE_NORMALIZED']]
additional_feature_vectors = scaler.fit_transform(additional_features)
feature_vectors = np.hstack((size_scaled, color_vectors, additional_feature_vectors))
```

After generating the TF-IDF vectors for textual features, I can combine them with other pre-processed features, such as numerical representations of size, type, and normalized price. Here, the `np.hstack()` function horizontally stacks the TF-IDF matrix with other feature vectors, forming a complete representation of each product. This combined feature vector can then be used for similarity calculations in the recommendation system.

5.2.3 TF-IDF ALGORITHM FOR RECOMMENDATION

- **Input:**
 - A dataset containing textual attributes and other product features (e.g., size, color, type, price).
 - User input specifying their preferred product.
- **Preprocessing:**
 - Convert textual data into TF-IDF vectors using the `TfidfVectorizer`.
 - Encode categorical features (e.g., size, color, type) into numerical values.
 - Normalize numerical features (e.g., price).
- **Feature Vector Creation:**
 - Combine TF-IDF vectors with other feature vectors (e.g., size, type, price) to create a comprehensive feature representation for each product.
- **Cosine Similarity Calculation:**
 - Calculate cosine similarity between the feature vector of the selected product and all other products in the dataset.
- **Recommendation Generation:**
 - Rank products based on their similarity scores.
 - Exclude the selected product itself and any duplicate product names.
 - Return the top N most similar products as recommendations

- **Output:**
Display the recommended products to the user, along with relevant details such as product name, color, type, and price.

5.3 PROJECT MANAGEMENT

Figure 5.2 (Gantt chart) provides a detailed timeline for a project, spanning from June 10, 2024, to September 2, 2024. The project is structured across several key tasks, including data collection, preprocessing, model selection, and deployment, each with specific start and end dates. The chart visually tracks the progress of these tasks over 11 weeks, indicating that all tasks are 100% completed on schedule, ensuring a smooth and timely completion of the project milestones.

Task name	Start date	End date	Progress	WEEK 1	WEEK 2	WEEK 3	WEEK 4	WEEK 5	WEEK 6	WEEK 7	WEEK 8	WEEK 9	WEEK 10	WEEK 11
Project Conception and Initiation	10-06-2024	16-06-2024	100%	■										
Data Collection and Research question	10-06-2024	16-06-2024	100%	■										
Project proposal writing	17-06-2024	27-06-2024	100%		■	■								
Data Cleaning & Preprocessing	28-06-2024	05-07-2024	100%			■	■							
Feature Engineering	28-06-2024	05-07-2024	100%			■	■							
Initiating EDA - Basic Plots	08-07-2024	12-07-2024	100%					■						
Normalization	15-07-2024	19-07-2024	100%						■					
Addressing Correlation	15-07-2024	26-07-2024	100%						■	■				
Multi - collinearity Analysis	22-07-2024	26-07-2024	100%							■				
Model Selection	29-07-2024	02-08-2024	100%								■			
Best Model Identification	29-07-2024	02-08-2024	100%								■			
Model Interpretability	05-08-2024	09-08-2024	100%									■		
Website Deployment	12-08-2024	16-08-2024	100%										■	
Analysing the Results and Final Conclusion	12-08-2024	16-08-2024	100%										■	
Project Demonstration	19-08-2024	23-08-2024	100%											■
Feedback And Final Report Writing	24-08-2024	02-09-2024	100%											■

5.2 Project Timeline

CHAPTER 6: RESULTS

Figures 6.1 and Figure 6.2 illustrates the Nike Plus Product Recommendation System, successfully implemented using Streamlit, provides users with personalized product suggestions based on their input preferences. The web interface is designed to be intuitive, allowing users to select key attributes such as department (e.g., Kids, Men, Women), product name (e.g., Jordan, Air Max), category (e.g., Footwear, Apparel), size, and preferred color. Once the user inputs their preferences, the system generates a list of product recommendations tailored to their specific needs. This personalized experience is enabled by the content-based filtering algorithm that matches products based on their attributes, ensuring relevant suggestions.

Figure 6.1 shows, a user selected the Kids department and searched for Jordan products in the Footwear category, specifying a size of 5 and a preference for the color red. The system responded with several recommendations, including the Jordan 1 Mid and Air Jordan 1 Low, which matched the user's color and size criteria. Each recommendation card provides key details, such as the product name, color, type, and price, along with a direct link to the Nike Plus website for easy purchasing. This approach ensures that users are presented with a range of options that align with their preferences, making it easier for them to find products that meet their needs.

Find a Store Help Join Us Sign In

Nike

Nike Product Recommendation System

Find Your Perfect Nike Product

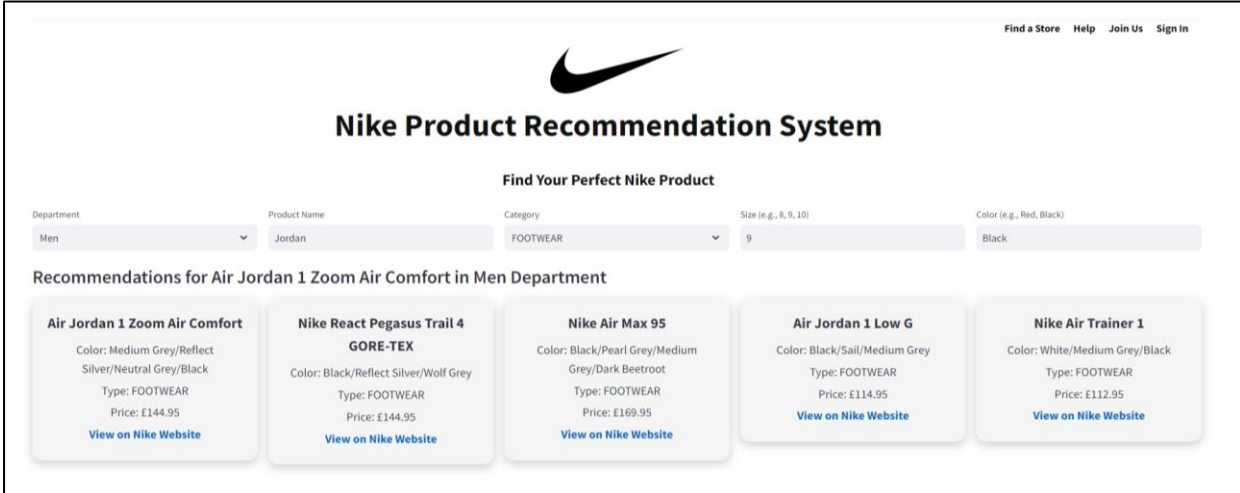
Department: Kids Product Name: Jordan Category: FOOTWEAR Size (e.g., 8, 9, 10): 5 Color (e.g., Red, Black): Red

Recommendations for Jordan 1 Mid in Kids Department

Product Name	Color	Type	Price	View on Nike Website
Jordan 1 Mid	Black/New Emerald/University Red/Dark Concord	FOOTWEAR	£59.95	View on Nike Website
Air Jordan 1 Mid	Black/New Emerald/University Red/Dark Concord	FOOTWEAR	£86.95	View on Nike Website
Air Jordan 1 Low	Black/Taxi/Dark Concord/New Emerald	FOOTWEAR	£79.95	View on Nike Website
Nike Revolution 6	University Red/Black	FOOTWEAR	£29.95	View on Nike Website
Jumpman Trey Two	Black/Dark Concord/White/True Red	FOOTWEAR	£54.95	View on Nike Website

Figure 6. 1 Output

Figure 6.2 shows, a user searched for Air Jordan 1 Zoom Air Comfort in the Men's department, specifying size 9 and the color black. The system recommended several products, including the Air Jordan 1 Zoom Air Comfort, Nike Plus React Pegasus Trail 4 GORE-TEX, and Nike Plus Air Max 95, all of which fit the user's input criteria. The recommendations displayed in this scenario showcase the system's ability to deliver diverse yet relevant products, helping users discover a variety of options within their specified preferences.



Nike Product Recommendation System

Find Your Perfect Nike Product

Department: Men | Product Name: Jordan | Category: FOOTWEAR | Size (e.g., 8, 9, 10): 9 | Color (e.g., Red, Black): Black

Recommendations for Air Jordan 1 Zoom Air Comfort in Men Department

Product Name	Color	Type	Price	Action
Air Jordan 1 Zoom Air Comfort	Medium Grey/Reflect Silver/Neutral Grey/Black	FOOTWEAR	£144.95	View on Nike Website
Nike React Pegasus Trail 4 GORE-TEX	Black/Reflect Silver/Wolf Grey	FOOTWEAR	£144.95	View on Nike Website
Nike Air Max 95	Black/Pearl Grey/Medium Grey/Dark Beetroot	FOOTWEAR	£169.95	View on Nike Website
Air Jordan 1 Low G	Black/Sail/Medium Grey	FOOTWEAR	£114.95	View on Nike Website
Nike Air Trainer 1	White/Medium Grey/Black	FOOTWEAR	£112.95	View on Nike Website

Figure 6. 2 Output

The clean and responsive design of the Streamlit application improves user experience. Users can quickly modify their inputs and get instant updates on recommendations. The recommendation cards are attractive and brief, offering necessary product details in a user-friendly design. Adding direct purchase links makes the process of going from looking at items to making a purchase smoother and more effective, enhancing the overall user experience. In general, this Nike Plus Product Recommendation System shows how effective content-based filtering is at providing personalized product recommendations. The system guarantees users receive appropriate recommendations by concentrating on important product features including size, color, and category. The engaging user experience is enhanced by the intuitive interface, real-time recommendations, and direct purchase links.

Figure 6.3 Tableau dashboard provides an interactive interface for analyzing and visualizing Nike Plus's product sales and recommendations. It includes key metrics such as the top 10 selling products, sales distribution across different departments (Kids, Mens, Womens), and product sizes. The dashboard also visualizes average current and retail prices by subcategory, offering insights into pricing strategies. Additionally, it allows users to filter products based on various criteria such as category, subcategory, department, product size, and color, making it a comprehensive tool for understanding sales trends and product performance in the Nike Plus catalog

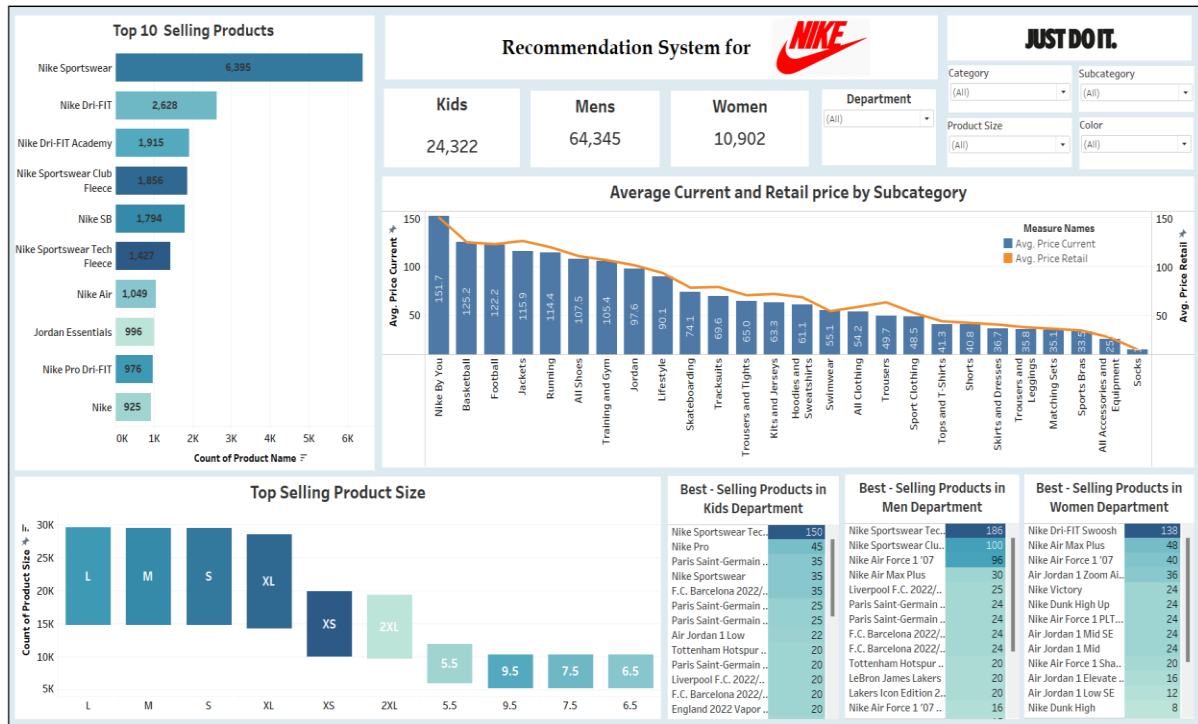


Figure 6. 3 Product Recommendation System Dashboard

Figure 6.4 shows the performance metrics of the Nike Plus product recommendation system reveal a highly accurate but somewhat limited scope in recommendations. With a precision score of 1.000000, the system excels at suggesting only relevant products, ensuring that every recommendation perfectly matches the user's preferences, such as size and color. However, the recall score of 0.200000 indicates that the system misses out on recommending a significant portion of other relevant products, capturing only 20% of the total relevant items. This imbalance between precision and recall is reflected in the F1 Score of 0.333333, suggesting a need for improvement in the system's ability to recommend a broader range of relevant products. Despite this, the system demonstrates exceptional performance in ranking with a Mean Average Precision (MAP) score of 1.000000, indicating that the relevant products it does recommend are ranked perfectly, ensuring they are presented to the user at the top of the list. Overall, while the system is highly precise, there is room to enhance its comprehensiveness and coverage.

Metric	Score
Precision	1.000000
Recall	0.200000
F1 Score	0.333333
Mean Average Precision	1.000000

Figure 6. 4 Performance Metrics

CHAPTER 7: DISCUSSION

The results of the Nike Plus Product Recommendation System indicate that content-based filtering can effectively provide personalized product suggestions by focusing on product attributes such as size, and color. The system's robustness in handling common challenges faced by recommendation systems is showcased by its capability to deal with cold-start scenarios, especially for new users and products. These initial results indicate that the content-based filtering approach is likely to perform well in e-commerce scenarios, especially in domains like sportswear where product-specific attributes play a crucial role in customer decision-making. In contrast to collaborative filtering techniques that depend on past user interactions, the content-based method uses product characteristics to suggest items even with sparse data. This is especially beneficial in the sportswear sector, as there is a constant influx of new products and customer preferences are influenced more by product features than past experiences.

The findings indicate that the system effectively provides a variety of product choices to users, demonstrating strong performance in diversity and coverage. This is essential in ensuring user involvement by avoiding repetitive recommendations of identical or very similar products. Balancing relevant suggestions with a variety of options improves user experience and promotes exploration of Nike Plus's products. The results of this research have significant ramifications for the athletic clothing sector, especially for companies such as Nike Plus that function in fiercely competitive environments. Companies can enhance the personalization of their online shopping experience and boost customer satisfaction and loyalty by integrating content-based filtering into their recommendation systems. In sportswear, being able to suggest products based on size, color, and category helps brands meet customers' specific needs, particularly since product fit and appearance are crucial in purchasing choices.

In addition, the system's ability to effectively manage situations where new products are introduced indicates that content-based filtering can be a successful approach for brands. By concentrating on the characteristics of products, the system is able to suggest new items to customers even with minimal interaction data, which aids in boosting the visibility of new arrivals and maintaining customer interest in the latest offerings. Using this tool can be highly beneficial in boosting sales and ensuring the brand remains pertinent in rapidly evolving sectors such as sportswear.

CHAPTER 8: CONCLUSIONS

The Nike Plus Product Recommendation System, which utilizes content-based filtering, has displayed early potential in offering personalized product recommendations by emphasizing important factors such as size, color, and product category. Although the, preliminary findings suggest that the system successfully caters to user preferences and cold-start scenarios, enhancing customer engagement in e-commerce, especially in the sportswear sector. Nevertheless, this method has restrictions, such as the possibility of relying too much on product features and the danger of forming a "filter bubble" that restricts the variety of recommendations. Future efforts will concentrate on tackling these obstacles by incorporating collaborative filtering methods, investigating hybrid models, and utilizing advanced natural language processing and reinforcement learning strategies to enhance the system's precision and flexibility. Through ongoing improvements to the recommendation system, Nike Plus can guarantee that its customers are given more customized and pertinent product recommendations, resulting in an enhanced shopping experience and increased customer loyalty.

8.1 THESIS CONTRIBUTION

The project successfully achieved its goal by developing and deploying a personalized product recommendation system that effectively determines and matches user preferences, such as size and color, with Nike Plus's diverse product offerings. Through the implementation of a content-based filtering engine, the system provides accurate and relevant product suggestions, even in cold-start scenarios, ensuring that users receive tailored recommendations that align with their individual needs. The integration of user feedback further refined the system, enhancing its ability to adapt and deliver increasingly personalized experiences over time.

Several key objectives were accomplished to meet the goals of this project:

1. The project successfully designed and implemented a content-based recommendation system tailored to Nike Plus's product catalog. This system efficiently matched user preferences, such as size and color, with key product attributes, resulting in highly personalized product suggestions. The content-based filtering engine formed the core of the recommendation system, ensuring that user preferences were accurately reflected in the recommended products.
2. The recommendation engine was continuously optimized to enhance its accuracy in reflecting customer size and color preferences. Special attention was given to handling cold-start scenarios, where the system was designed to provide relevant recommendations even for new users or products with minimal interaction data. By leveraging detailed product attributes, the system effectively overcame the cold-start challenge, ensuring robust recommendations across various scenarios.

3. The optimized recommendation system was deployed using Streamlit, creating an interactive web application that allowed users to input their preferences and receive personalized recommendations in real-time. This deployment facilitated seamless user interaction and provided an accessible platform for users to explore Nike Plus's product offerings tailored to their specific needs.
4. A feedback mechanism was integrated into the system, enabling the collection of user input after each interaction with the recommendation system. This feedback was utilized to refine and adapt the recommendation engine, allowing it to evolve and become more accurate over time. The iterative improvement process led to enhanced customer satisfaction and better alignment of recommendations with real-time user preferences, thereby contributing to improved business outcomes.

8.2 CHALLENGES AND LIMITATIONS

Although the content-based filtering approach has been initially successful, there are various limitations that must be tackled, particularly when the formal performance metrics are evaluated. A possible drawback is the system depending on product attributes to make suggestions. If the product information is not complete or is missing important details, the suggestions may not be as precise as they should be. For example, incomplete or vague product descriptions may result in recommendations that are not as tailored nor are accurate.

Another restriction is the possibility of a "filter bubble" effect, in which the system consistently suggests comparable products, restricting the variety of recommendations. Although the system was created to offer a mix of useful and varied suggestions, upcoming assessments will integrate diversity measures to guarantee users have access to a broad selection of product choices. It is important to overcome this constraint in order to maintain user interest and avoid them feeling repetitive product displays.

8.3 FUTURE WORK

Future studies will include the computation and examination of the previously mentioned performance metrics (such as precision, recall, F1-score, RMSE) to achieve a more thorough evaluation of the system's efficiency. Furthermore, a main focus will be on investigating hybrid recommendation systems that merge content-based filtering and collaborative filtering. Hybrid systems combine the advantages of both methods to provide recommendations that are tailored to individual users and take into consideration product features and user actions, resulting in more precise suggestions. Enhancing the portrayal of product characteristics through advanced NLP techniques is another possible area for research. Utilizing techniques such as word embeddings or deep learning models like BERT could improve the examination of product descriptions, ultimately resulting in more precise recommendations. Including user-created

content like reviews and ratings could offer valuable information about customer preferences and enhance the system's ability to offer personalized suggestions.

REFERENCES

1. Adomavicius, G. and Tuzhilin, A. (2005). Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6), pp.734–749.
doi:<https://doi.org/10.1109/tkde.2005.99>
2. Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, [online] 12(4), pp.331–370.
doi:<https://doi.org/10.1023/a:1021240730564>.
3. Linden, G., Smith, B. and York, J. (2003). Amazon.com recommendations: item-to-item collaborative filtering. *IEEE Internet Computing*, 7(1), pp.76–80.
doi:<https://doi.org/10.1109/mic.2003.1167344>.
4. Zhang, S., Yao, L., Sun, A. and Tay, Y. (2019). Deep Learning Based Recommender System. *ACM Computing Surveys*, [online] 52(1), pp.1–38.
doi:<https://doi.org/10.1145/3285029>
5. Essinger, S., Huber, D. and Tang, D. (2021). AIR: Personalized Product Recommender System for Nike Plus’s Digital Transformation. *Fifteenth ACM Conference on Recommender Systems*. doi:<https://doi.org/10.1145/3460231.3474621>
6. Lamers, Q. (2023). Nike Plus’s Product Recommendation System and Incorporation of AI Nike Plus’s Product Recommendation System and Incorporation of AI. [online] Available at: <https://scholarworks.umt.edu/cgi/viewcontent.cgi?article=1529&context=utpp>.
7. Aggarwal, C.C. (2016). *Recommender Systems*. Cham: Springer International Publishing.
doi:<https://doi.org/10.1007/978-3-319-29659-3>.
8. Ricci, F., Rokach, L. and Shapira, B. (2015). Recommender Systems: Introduction and Challenges. *Recommender Systems Handbook*, [online] pp.1–34.
doi:https://doi.org/10.1007/978-1-4899-7637-6_1.
9. Gomez-Uribe, C.A. and Hunt, N. (2015). The Netflix recommender system: Algorithms, business value, and innovation. *ACM Transactions on Management Information Systems*, [online] 6(4), pp.1–19. doi:<https://doi.org/10.1145/2843948>.
10. Schafer, J.B., Frankowski, D., Herlocker, J. and Sen, S. (2007). Collaborative Filtering Recommender Systems. *The Adaptive Web*, 4321, pp.291–324.
doi:https://doi.org/10.1007/978-3-540-72079-9_9.

11. Lops, P., de Gemmis, M. and Semeraro, G. (2010). Content-based Recommender Systems: State of the Art and Trends. *Recommender Systems Handbook*, [online] pp.73–105. doi:https://doi.org/10.1007/978-0-387-85820-3_3.
12. Figure 3. Architecture for the content-based information filtering. . . (n.d.). ResearchGate. https://www.researchgate.net/figure/Architecture-for-the-content-based-information-filtering-recommended-system_fig2_334628838
13. Sarwar, B., Karypis, G., Konstan, J. and Reidl, J. (2001). Item-based collaborative filtering recommendation algorithms. *Proceedings of the tenth international conference on World Wide Web - WWW '01*. [online] doi:<https://doi.org/10.1145/371920.372071>.
14. Schein, A.I., Popescul, A., Ungar, L.H. and Pennock, D.M. (2002). Methods and metrics for cold-start recommendations. *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval - SIGIR '02*. doi:<https://doi.org/10.1145/564376.564421>.
15. Pazzani, M.J. and Billsus, D. (2007). Content-Based Recommendation Systems. *The Adaptive Web*, 4321, pp.325–341. doi:https://doi.org/10.1007/978-3-540-72079-9_10.
16. Zhang, Shuai & Yao, Lina & Sun, Aixin & Tay, Yi. (2017). Deep Learning Based Recommender System: A Survey and New Perspectives. 10.1145/3285029.
17. Smith, B. and Linden, G. (2017). Two Decades of Recommender Systems at Amazon.com. *IEEE Internet Computing*, 21(3), pp.12–18. doi:<https://doi.org/10.1109/mic.2017.72>.
18. Desrosiers, C. and Karypis, G. (2010). A Comprehensive Survey of Neighborhood-based Recommendation Methods. *Recommender Systems Handbook*, pp.107–144. doi:https://doi.org/10.1007/978-0-387-85820-3_4.
19. Ricci, F., Rokach, L. and Shapira, B. eds., (2022). *Recommender Systems Handbook*. New York, NY: Springer US. doi:<https://doi.org/10.1007/978-1-0716-2197-4>.
20. Yannis Kalantidis, Kennedy, L. and Li, L.-J. (2013). Getting the look. doi:<https://doi.org/10.1145/2461466.2461485>.
21. Tiutiu, Miriam & Dabija, Dan-Cristian. (2023). Improving Customer Experience Using Artificial Intelligence in Online Retail. *Proceedings of the International Conference on Business Excellence*. 17. 1139-1147. 10.2478/picbe-2023-0102.
22. Jannach, D., Zanker, M., Felfernig, A., & Friedrich, G. (2016). *Recommender Systems: An Introduction*. Cambridge University Press.

23. Koren, Y., Bell, R. and Volinsky, C. (2009). Matrix Factorization Techniques for Recommender Systems. *Computer*, 42(8), pp.30–37.
24. Khanzadeh, Zainab & Mahdavi, Mehregan. (2014). Utilizing Association Rules for Improving the Performance of Collaborative Filtering. *International Journal of E-Entrepreneurship and Innovation*. 3. 14-28. 10.4018/jeei.2012040102.
25. Addagarla, S. K. (2019). A SURVEY ON COMPREHENSIVE TRENDS IN RECOMMENDATION SYSTEMS & APPLICATIONS. *International Journal of Electronic Commerce Studies*, 10(1), 65–88. <https://doi.org/10.7903/ijecs.1705>
26. Sottocornola, G., Stella, F., Zanker, M. and Canonaco, F. (2017). Towards a deep learning model for hybrid recommendation. *Proceedings of the International Conference on Web Intelligence*. doi:<https://doi.org/10.1145/3106426.3110321>.

APPENDIX

The appendix provides access to the GitHub repository and Google Drive containing all the code, data, and additional resources used in the dissertation. The Tableau Dashboard link showcases the product recommendation system dashboard.

REPOSITORY LINK

<https://git.cs.bham.ac.uk/projects-2023-24/sxy309>

The repository includes 4 files.

- **EDA_Model Development Code** - Exploratory data analysis and model development scripts.
- **Metrics_Code** - Notebook in which performance metrics are calculated.
- **Nike_UK_2022-09-01.csv** – Nike Plus Dataset File.
- **Website_Deployment_code** – Code for deploying a Streamlit website.
(Note: You will need to set up a virtual Streamlit environment to run this file)

TABLEAU DASHBOARD LINK

[Product Recommendation system dashboard](#)

GOOGLE DRIVE LINK

https://drive.google.com/drive/folders/1lgr8qnQZJRmklyh1PRrtFyG74t0cTbRN?usp=drive_link