Online Retail Recommendation System

A Major Project Report
submitted in partial fulfilment of the requirements for
the award of the degree of

Internship

in

Data Science

By

Name: P.Sri Harsha

Email:srihrshapadala@gmail.com

Under the esteemed guidance of

ABSTRACT

The Online Retail Recommendation System project aims to develop an intelligent recommendation engine for an online retail platform using Python. This project leverages data analysis techniques to provide personalized product recommendations to users, enhancing their shopping experience and boosting sales. The system utilizes a comprehensive retail dataset to identify patterns and trends in customer behaviour.

This system is built on collaborative filtering techniques, which predict user preferences based on historical purchase data. The system is further refined using content-based filtering to recommend similar products. Functions are implemented to generate and print personalized recommendations for users.

The project culminates in a user-friendly recommendation system that provides real-time product suggestions, aiming to improve user satisfaction and drive sales growth. The use of Python for data analysis and machine learning underscores the flexibility and power of this language in developing sophisticated recommendation systems.

Keywords:

Online-Retail, Recommendation system, Python, Retail-Dataset, Collaborative Filtering, Predictive Modelling, Sales Trends, Customer Preferences

ACKNOWLEDGEMENTS

I would like to express our deepest gratitude to the following people for guiding me through this course and without whom this project and the results achieved from it would not have reached completion.

The Lecturer, Department of Data Science and Plasmid, for helping me and guiding me in the course of this project. Without his guidance, I would not have been able to successfully complete this project. His patience and genial attitude is and always will be a source of inspiration to me.

The Head of the Internship, Department of Data Science, for allowing us to avail the facilities at the department.

I am also thankful to the faculty and staff members of the Department of Data Science for their constant support and help.

CHAPTER-1

INTRODUCTION

Overview

The Online Retail Recommendation System project aims to develop a sophisticated recommendation engine for an online retail platform using Python. The primary goal is to enhance user experience by providing personalized product recommendations through comprehensive data analysis and machine learning techniques. This project involves several key objectives, starting with data collection and preprocessing, utilizing a comprehensive retail dataset. The data is cleaned and preprocessed using Python libraries like Pandas and NumPy.

Exploratory Data Analysis (EDA) is conducted to understand the dataset better, employing Seaborn for data visualization to identify patterns and trends in customer behaviour, item popularity, and sales distribution. Analysis of sales data is performed using pivot tables to explore sales data globally, country-wise, and month-wise, identifying top-selling products and seasonal trends. This project employs collaborative filtering techniques to predict user preferences based on historical purchase data, and content-based filtering to recommend similar products, enhancing recommendation accuracy. Predictive modelling techniques are applied using machine learning algorithms to forecast future sales trends and customer preferences, refining the recommendation system.

Ensures real-time suggestions to users, enhancing their shopping experience. The performance of the recommendation system is evaluated using metrics such as precision, recall, and F1-score, with continuous optimization based on evaluation results to ensure high accuracy and relevancy of recommendations. The methodology involves data handling, visualization and insights, recommendation techniques, machine learning models, and system integration. Addressing data quality issues for accurate analysis. Visualization and insights use Seaborn to create visualizations that provide insights into sales trends and customer behaviour. Recommendation techniques include collaborative filtering using user-item interaction data and content-based filtering to suggest similar products. The expected outcomes of the project

include a robust recommendation engine that provides personalized product suggestions to users, improved user satisfaction and engagement through relevant recommendations, and increased sales and revenue for the online retail platform. The system is designed to be scalable and adaptable, capable of handling large volumes of data and evolving user preferences. This project demonstrates the practical solution to enhance the shopping experience and drive business growth.

1.2 APPLICATION OF DATA SCIENCE IN ONLINE RETAIL RECOMMEDATION SYSTEM

In respect of applications, Data Science approaches have been successfully applied in many areas related to analysis like:

Personalized Recommendations, Customer Segmentation, Market Basket Analysis, Customer Lifetime Value (CLV) Prediction, Dynamic Pricing, Inventory Management, Sentiment Analysis, Customer Support Automation, Fraud Detection, Sales and Revenue Forecasting and so on. By the analysis we can predict that how the recommendation can be occurred.

1.3 PROBLEM STATEMENT

The objective of this project is to develop an online retail recommendation system using advanced data science techniques to enhance customer experience and drive sales. The system will analyse customer behaviour, purchase history, and product attributes to deliver accurate and personalized recommendations.

Key challenges include:

data integration, model selection, scalability, user privacy, dynamic recommendations

. The project aims to provide a robust, real-time recommendation system that improves user engagement, optimizes inventory management, enables targeted marketing, and ultimately boosts sales and revenue for the online retail business.

The outcomes of this project is to enhance user experience, increased sales and revenue , optimized inventory management, improved customer segmentation and targeting, scalable and robust system.

1.4 ORGANISATION OF THE PROJECT

Identify and acquire Datasets. Define the problem statement. Define project goals, objectives, and deliverables based on the problem statement. Visualize spatial and temporal patterns of online recommendation monthly, globally, and sales per month using maps, and graphs.

The Deployment and Integration section details how the recommendation system will be deployed in a production environment and integrated with the online retail platform. Finally, the Continuous Monitoring and Improvement section outlines strategies for monitoring system performance, incorporating user feedback, and updating models to adapt to evolving data and user behaviour.

CHAPTER-2

LITRATURE SURVEY

2.1 Collaborative filtering recommendation systems

Recommendation systems are one of the most important applications in big data analytics and have performed excellently for numerous businesses (Bobadilla et al., 2013, Shi et al., 2014, Su and Khoshgoftaar, 2009). Many online companies, such as Amazon (Linden et al., 2003), Netflix (Koren, 2009a), Google (Das et al., 2007), and Facebook (Shapira et al., 2013), are using recommendation systems as part of their business.

Recommendation systems are broadly categorized into content-based systems and collaborative filtering systems. Content-based systems recommend products which have content similar to products preferred by a customer. Content-based systems use content to build a model for recommendation, but this study does not use this approach. Instead, we use a product content model to improve the collaborative filtering system as discussed below.

On the other hand, collaborative filtering systems are popular in business as well as in research because of their simplicity and its high-performance levels (Bobadilla et al., 2013, Shi et al., 2014, Su and Khoshgoftaar, 2009). Collaborative filtering systems are based on customer ratings of products regardless of the availability of product content. Two approaches have been developed for collaborative filtering systems. User-based collaborative filtering systems recommend products which have been chosen most in the past by similar customers (Breese et al., 1998, Herlocker et al., 2004, Konstan et al., 1997, Resnick et al., 1994, Sarwar et al., 2001, Shardanand and Maes, 1995). For any two given customers, their similarity is calculated based on their ratings of products that both have rated. Correlation (Konstan et al., 1997, Shardanand and Maes, 1995) and cosine similarity (Breese et al., 1998, Sarwar et al., 2001) are commonly used as measures of similarity. Default voting, inverse user frequency, case amplification and

weighted-majority prediction are employed to aggregate similar users' ratings (Breese et al., 1998, Delgado and Ishii, 1999).

Item-based collaborative filtering systems analyze similarities between products and recommend products that are most similar to products selected by the customer (Shardanand and Maes, 1995). The advantage of this approach is that it can precompute similarities between products and can be presented as soon as a customer clicks or buys a product.

A typical collaborative filtering system focuses on a user-item matrix that represents customer clicks or purchases of products in a matrix format. However, recent collaborative filtering systems improve their performance by using additional information related to users and products and information related to the interaction of users and products (Shi et al., 2014).

2.2 Combining purchase and click data

This research aims to integrate customers' click data and purchase data in a collaborative filtering system. The multi-criteria recommendation system research (Adomavicius and Kwon, 2007, Jannach et al., 2012, Lee and Teng, 2007, Nilashi et al., 2014a) is similar to our approach, because it views offline customer purchase data as additional ratings. Adomavicius and Kwon (2007) first proposed a multi-criteria recommendation problem, while Lee and Teng (2007) use a skyline query technique to solve the multi-criteria recommendation problem, because they regard the multi-criteria recommendation problem as an optimization problem. Jannach et al. (2012) suggest a support vector regression (SVR) to combine multiple ratings demonstrating that the SVR outperforms single-rating algorithms. Nilashi et al. (2014b) show that combining dimensionality reduction and Neuro-Fuzzy techniques can improve recommendation quality significantly. However, only a few studies directly address the integration of online and offline preferences (Cheema and Papatla, 2010, Dzyabura et al., 2016, Kim et al., 2016). Dzyabura et al. (2016) use offline preferences for predicting online preferences.

Although this research does not directly address the issue of recommendation, it demonstrates that online information can be used for the prediction of offline preferences. Cheema and Papatla (2010), and Kim et al. (2016), propose friend recommendation using offline information (e.g., place visit history) and online information (e.g., friends' relationship), but

our research additionally addresses information related to preference. The recommended method developed in this study is based on the method proposed by Cheema and Papatla, 2010, Dzyabura et al., 2016, who proposed a recommendation system that uses offline sales data to improve the performance of recommendation system developed using online data. In this study, the same products are available in online and offline stores, but online and offline customers are different, and there is no information that can be used to link them to each other. For this reason, our recommendation system builds on item-based collaborative filtering.

2.3 Product information

Product information (e.g., category) can provide an additional opportunity for improving recommendation performance (Shi et al., 2014). Moshfeghi et al. (2009) suggests a collaborative filtering system for movie recommendation which uses the underlying semantics of movies as well as user rating. Singh and Gordon (2008) suggest a model based collaborative filtering system for movies, called collected matrix factorization (CFM). They combine the conventional user-item matrix with the matrix containing item information (e.g., a movie genre matrix). CFM reduces the sparsity problem in the conventional user-item matrix and enhances effective latent factors. Similarly, Zhu et al. (2007) suggest a document recommendation system that uses a joint matrix factorization approach. Hong et al. (2012) proposes a recommendation system that exploits product taxonomy to capture the user's preferences over products belonging to different category. Hung (2005) advocates a product recommendation system after classifying customers into three addictive categories: item addictive, brand addictive, and hybrid addictive. These studies did not utilize product category information to reflect purchase intentions, but we use product category information in our recommendation system. The proposed system first generates recommended products regardless of these types, and then recommends two set of products using product category information.

CHAPTER-3
SYSTEM DESIGN OR METHODOLOGY

3.1 Data collection:

The first step in developing an online retail recommendation system involves collecting comprehensive and high-quality data. This data can come from various sources, including:

Transaction Data: Purchase history, browsing history, and user interactions.

User Data: User profiles, demographics, and behaviour patterns.

Product Data: Product descriptions, categories, and attributes.

External Data: Social media activity, user reviews, and ratings.

2. Exploratory Data Analysis (EDA)

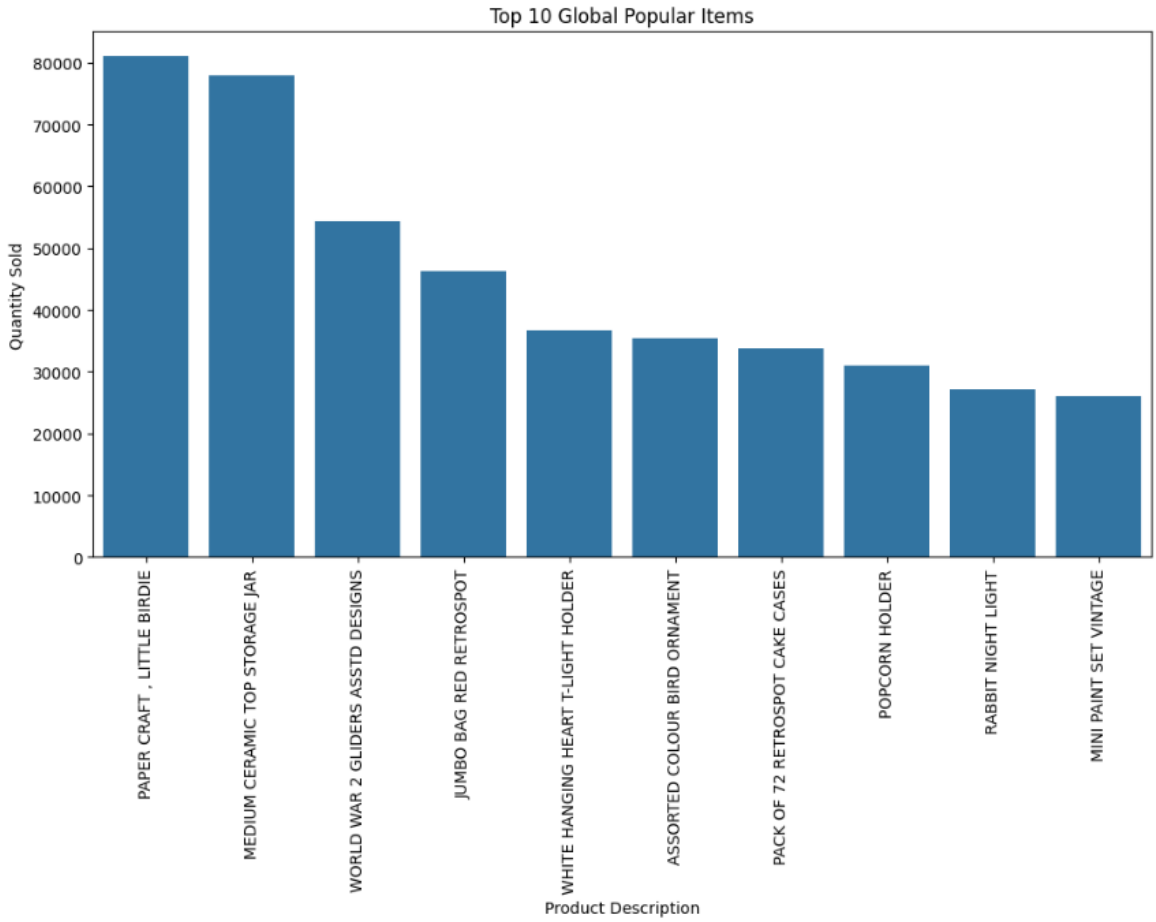
EDA helps understand the data and identify patterns, trends, and anomalies. Steps include:

Data Cleaning: Removing duplicates, handling missing values, and correcting inconsistencies.

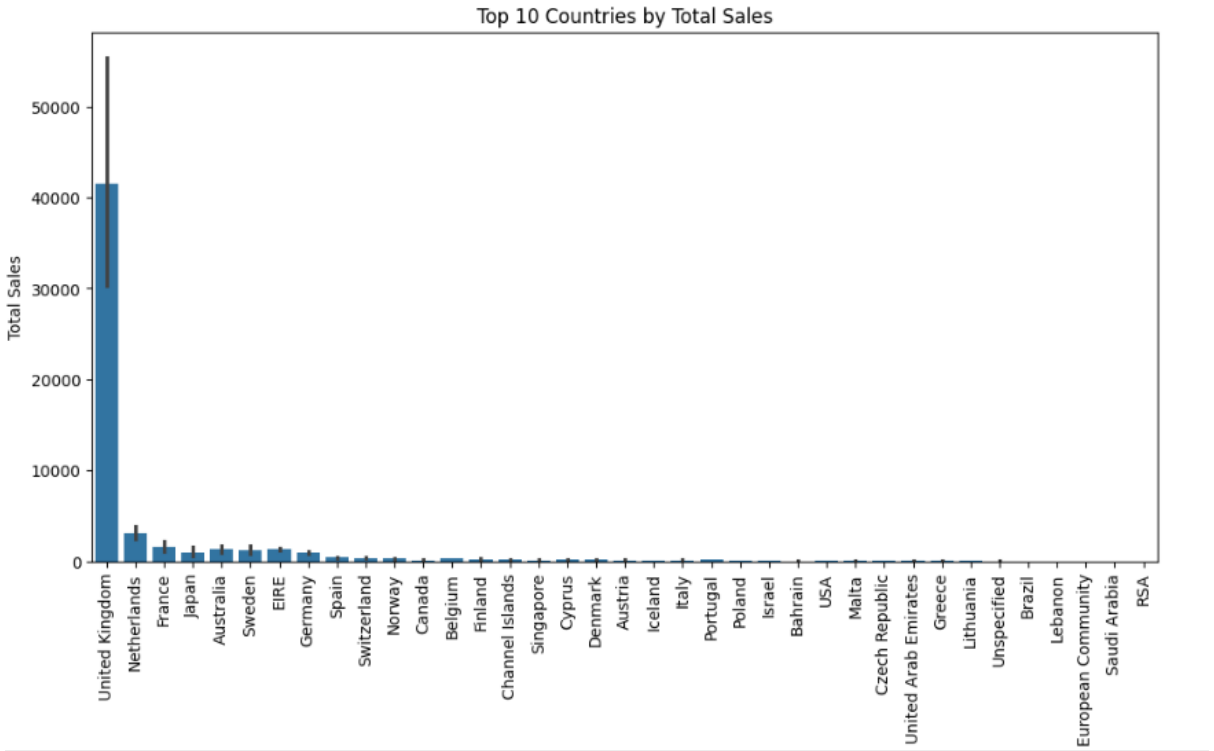
Data Visualization: Using tools like Seaborn and Matplotlib to visualize distributions, correlations, and trends.

Descriptive Statistics: Summarizing data with RMSE, and other statistical measures.

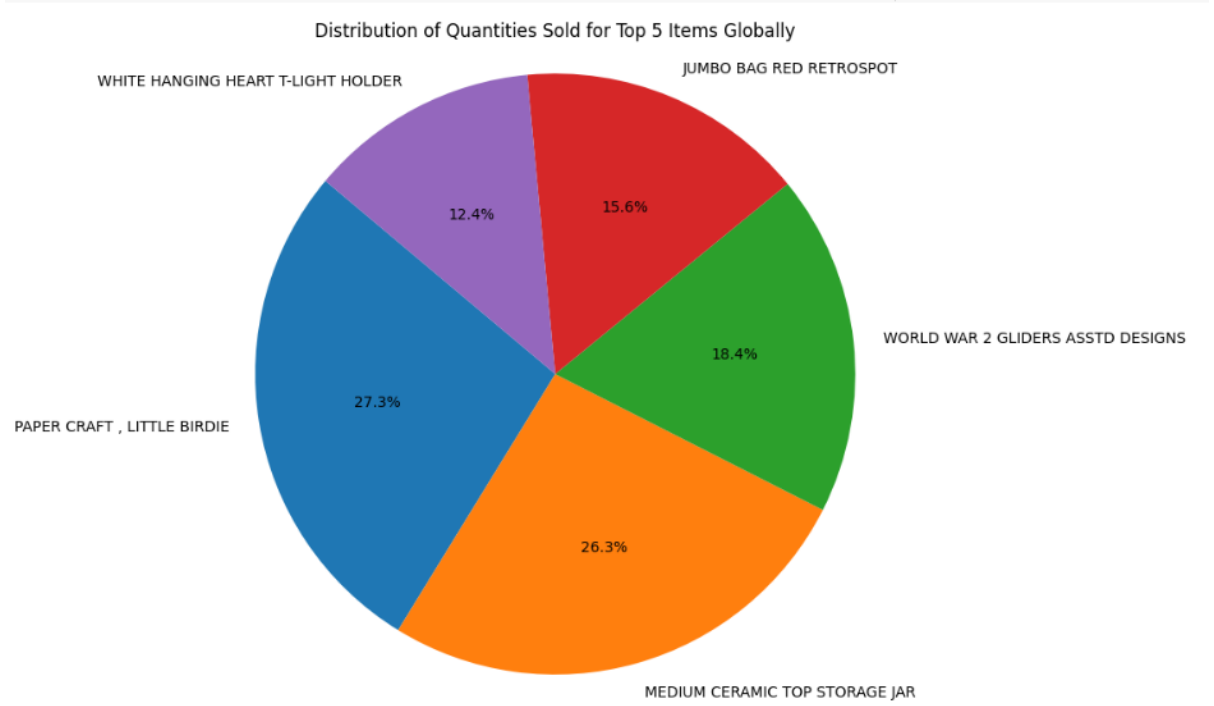
Correlation Analysis: Identifying relationships between different variables.



- Paper Craft, Little Birdie is the most popular item based on the highest quantity sold (around 80,000 units).
- The popularity of other items decreases gradually from left to right.
- It's possible that these items are popular globally, but without additional information about the dataset, it's difficult to say for certain.



- The country listed first (likely the United Kingdom in this case) has the highest total sales.
- Sales figures decrease gradually or more significantly for subsequent countries.

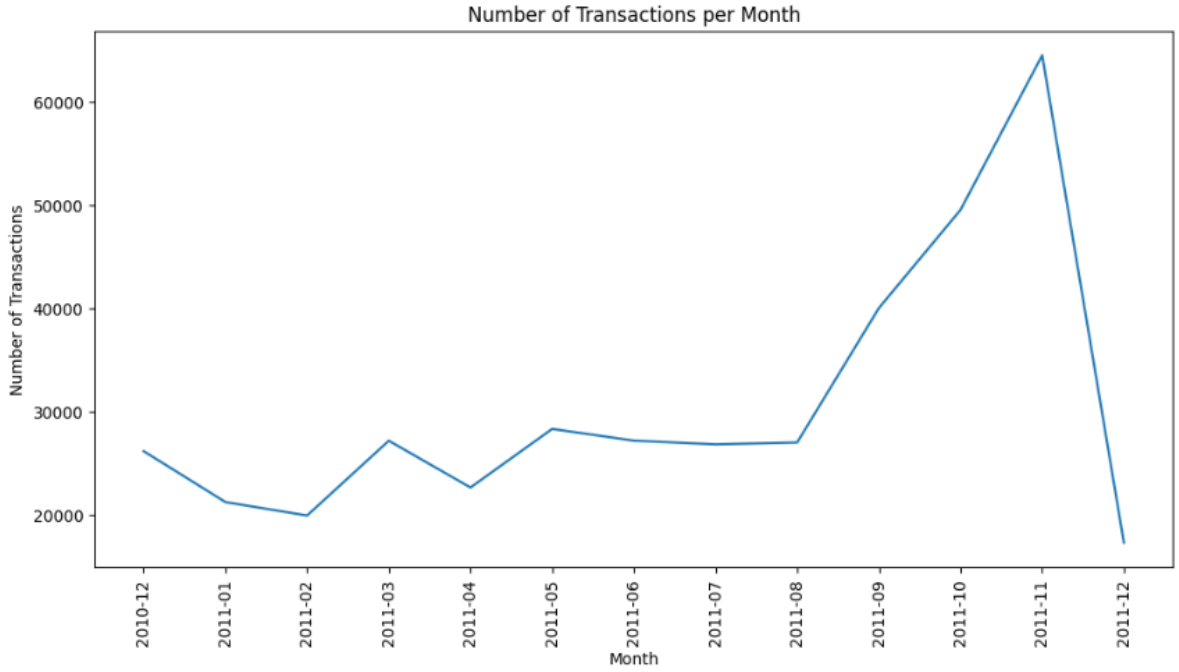


• The pie chart slices represent the proportion of the total quantity sold for each of the five items.

• Paper Craft, Little Birdie has the largest slice (26.3%), indicating it sold the most in quantity out of the five items.

• Medium Ceramic Top Storage Jar follows closely with 27.3%.

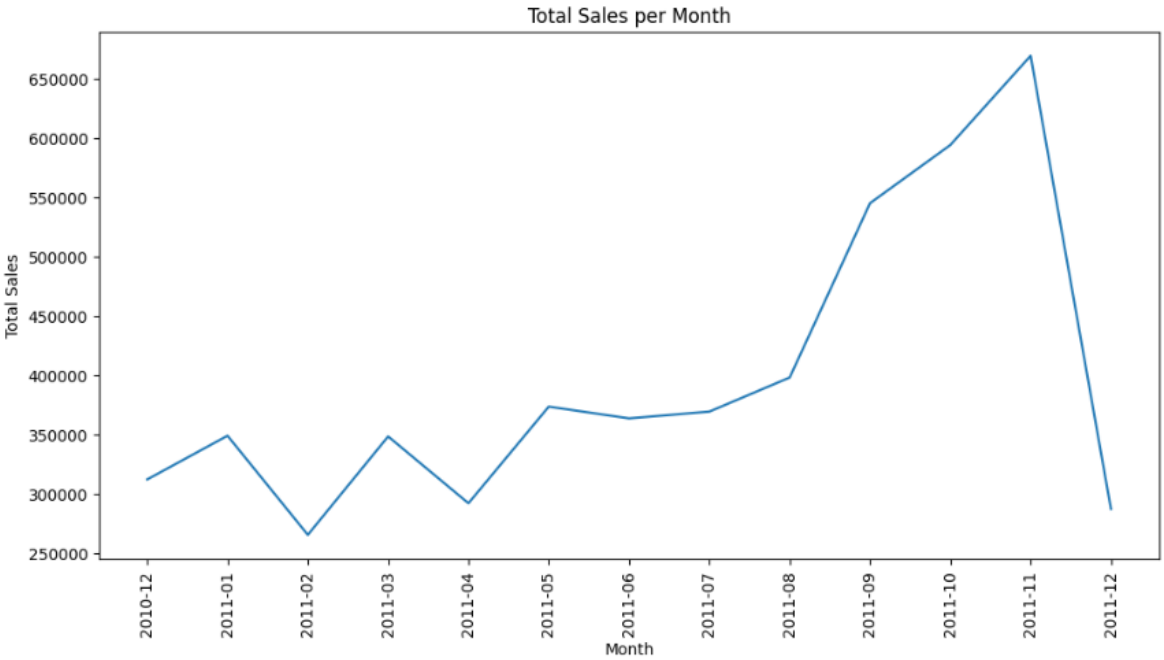
• The remaining three items have progressively smaller slices, with White Hanging Heart T-Light Holder having the least proportion (12.4%).



Transaction Fluctuation: The number of transactions fluctuates over the one-year period depicted in the graph.

Highest Transaction Month: December 2010 appears to be the month with the highest number of transactions (around 60,000).

Lowest Transaction Month: It's difficult to pinpoint the exact month with the lowest number of transactions due to the scale, but it likely falls between July and September 2011 (possibly around 30,000 - 40,000).



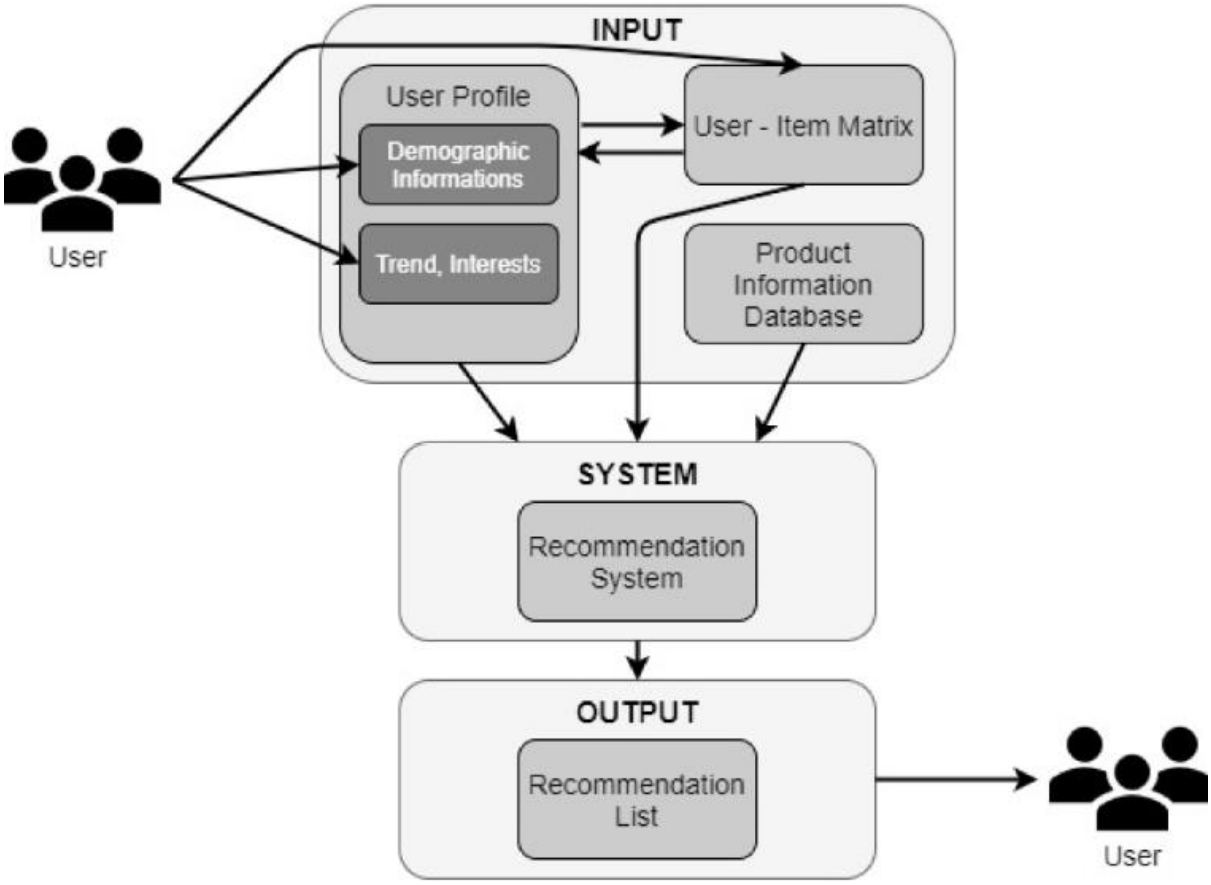
Sales Fluctuation: The company's total sales fluctuate over the one-year period depicted in the graph.

Highest Sales Month: The month with the highest total sales appears to be December 2010 (around 650,000), followed by a decrease in January 2011.

Lowest Sales Month: It's difficult to pinpoint the exact month with the lowest sales due to the scale, but it likely falls between July and September 2011 (possibly around 350,000 - 400,000).

3.3 Feature Engineering:

It starts with raw data about user purchases and product details. Then, data cleaning ensures the information is accurate. Feature engineering creates new informative attributes, like how often users buy a product or which products are frequently purchased together. Finally, these optimized features are used by a recommendation algorithm to suggest relevant items to users.



3.4 Model Selection and Training:

The model selection and training phase in developing an online retail recommendation system are crucial for achieving accurate and effective personalized recommendations. This phase begins with an assessment of various recommendation algorithms tailored to the project's objectives and dataset characteristics.

Initially, collaborative filtering methods such as user-based and item-based approaches are considered. User-based collaborative filtering identifies similarities among users based on their interaction histories to recommend items liked by similar users. Conversely, item-based collaborative filtering recommends items similar to those previously preferred by the user, leveraging item attributes and user feedback. Advanced techniques such as matrix factorization, including Singular Value Decomposition (SVD) is considered for their ability to capture latent factors in user-item interactions, especially useful in scenarios with implicit feedback.

Model evaluation involves assessing the performance of each algorithm using metrics like REMS and Mean Average Precision (MAP). Cross-validation techniques, like k-fold validation, ensure the robustness of the selected models by testing them on different subsets of data. A/B testing is also conducted to compare the performance of different models in real-world scenarios, ensuring the selected model not only performs well in evaluation metrics but also aligns with business objectives and user expectations.

Ultimately, the chosen model is trained on the entire dataset, optimizing parameters and fine-tuning to achieve the best possible performance. Regular monitoring and updates are planned to adapt to evolving user preferences and market dynamics, ensuring the recommendation system remains effective and relevant over time

3.5 Model Evaluation:

Model evaluation is a critical step in developing an online retail recommendation system, ensuring that the chosen model performs effectively in providing personalized recommendations to users. This phase involves rigorous assessment using various metrics and techniques to validate the model's performance and reliability.

Effective model evaluation in an online retail recommendation system involves a comprehensive assessment using diverse metrics and validation techniques. By selecting appropriate metrics aligned with business objectives and user expectations, and employing rigorous validation methods like cross-validation and A/B testing, developers can ensure the recommendation system not only meets performance benchmarks but also enhances user satisfaction and engagement. Regular monitoring and updates based on evaluation results are essential to maintain the system's effectiveness and relevance in a dynamic online retail environment.

3.6 IMPLEMENTATION

Initialize SVD Algorithm: Create an instance of the SVD algorithm from the Surprise library.

Train the Algorithm: Use the fit () method to train the algorithm on the training set (trainset).

Predict Ratings: Use the trained model to predict ratings for the test set (testset) using the test () method.

Evaluate Accuracy: Use the accuracy. Rmse () function to calculate and print the Root Mean Squared Error (RMSE), which is a common metric for evaluating the accuracy of collaborative filtering models.

Make Predictions: Optionally, you can make predictions for specific user-item pairs using the predict () method of the algorithm. Replace example_user_id and example_item_id with actual values from your dataset.

This sequence of steps will allow you to train an SVD model on your retail transaction data and evaluate its performance in terms of RMSE. Adjust the parameters and metrics as needed based on your specific project requirements.

3.7 Validation and Sensitivity Analysis:

By implementing validation techniques such as cross-validation, holdout validation, and A/B testing, as well as conducting sensitivity analysis through parameter tuning and feature importance evaluation, you can ensure that your recommendation system performs effectively and reliably. These steps help in optimizing model performance, identifying key parameters, and addressing potential biases or limitations in the recommendation process. Regular

validation and sensitivity analysis are essential to maintain the accuracy and relevance of the recommendation system over time.

3.8 Ethical Considerations:

When designing and deploying online retail recommendation systems, ethical considerations are paramount to ensure user trust, privacy, and fairness. Transparency is crucial; users should understand why specific products are recommended, and they should have control over their preferences and data. Privacy and data protection must be prioritized, with secure handling and anonymization of sensitive user information to prevent unauthorized access. Addressing algorithmic bias is essential to ensure recommendations are fair and inclusive, representing a diverse range of products and user interests. Informed consent is vital, with clear options for users to opt out of personalized recommendations if desired. By integrating these ethical considerations, businesses can build trustworthy, fair, and user-centric recommendation systems that respect and protect user rights while promoting positive societal impacts

3.9 Online Retail Recommendation System dataset and its attributes

Attributes	Discription
Invoice Number	This is the number that identifies a transaction
Stock Code	This refers to the product ID.
Description	This describes the product that a user purchased
Quantity	It specified the quantity of the item purchased.
Invoice Date	The date on which the transaction took place.
Unit Price	Price of one product.
Customer ID	It identifies the customer.
Country	The country where the transaction was performed

This dataset contains information related to retail market, with various stock and invoice attributes alongside the county where the transaction was performed, related to the online retail recommendation.

CHAPTER-4
IMPLEMENTATION

4.1 The following modules are used for implementation the system

4.1.1 package installation and loading:

Data Collection and Preprocessing

Modules: pandas, NumPy

Steps:

Load Data: Use pandas to read the dataset.

Clean Data: Drop rows with null Customer ID, convert Customer ID to integer, filter out invalid transactions.

Feature Engineering: Create a Total Sales column.

2. Exploratory Data Analysis (EDA)

Modules: seaborn, matplotlib

Steps:

Global Popular Items: Identify and visualize the top 10 globally popular items.

Country-wise Popular Items: Identify and visualize the top 10 popular items per country.

Monthly Popular Items: Identify and visualize the top 10 popular items per month.

Trends: Plot the number of transactions and total sales per month.

3. Feature Engineering

Modules: pandas

Steps:

Pivot Table: Create a pivot table for user-item interactions.

Aggregate Features: Generate aggregate features like monthly sales.

4. Model Selection and Training

Modules: surprise

Steps:

Data Loading: Use Reader and Dataset classes to load data for collaborative filtering.

Data Splitting: Split the dataset into training and test sets.

Model Training: Train the SVD algorithm on the training set.

5. Model Evaluation

Modules: surprise

Steps:

Predictions: Predict ratings for the test set.

Evaluation Metrics: Evaluate model performance using RMSE.

6. Validation and Sensitivity Analysis

Modules: surprise, sklearn

Steps:

Cross-Validation: Perform cross-validation to assess model performance.

Parameter Tuning: Use GridSearchCV for hyperparameter optimization.

Sensitivity Analysis: Test the model's robustness by introducing variations in input data.

7. Implementation and Deployment

Modules: Custom scripts, web frameworks (e.g., Flask, Django)

Steps:

Integration: Integrate the trained model into a recommendation engine.

Deployment: Deploy the recommendation engine within the online retail platform.

8. Ethical Considerations

Modules: Not specific to coding libraries, involves best practices

Steps:

Transparency: Ensure transparency in recommendations and provide user control.

Data Protection: Implement robust data privacy and protection measures.

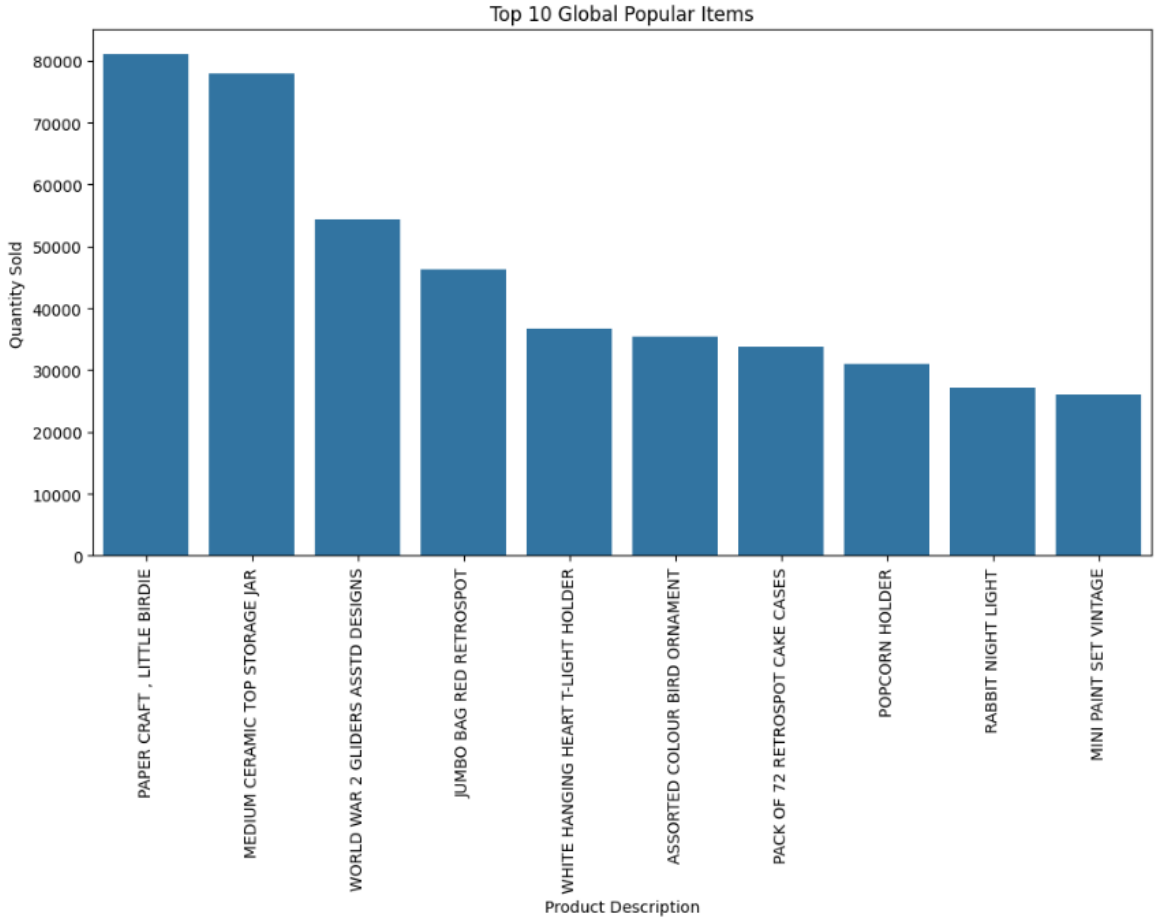
Fairness: Address algorithmic bias and ensure fair, inclusive recommendations.

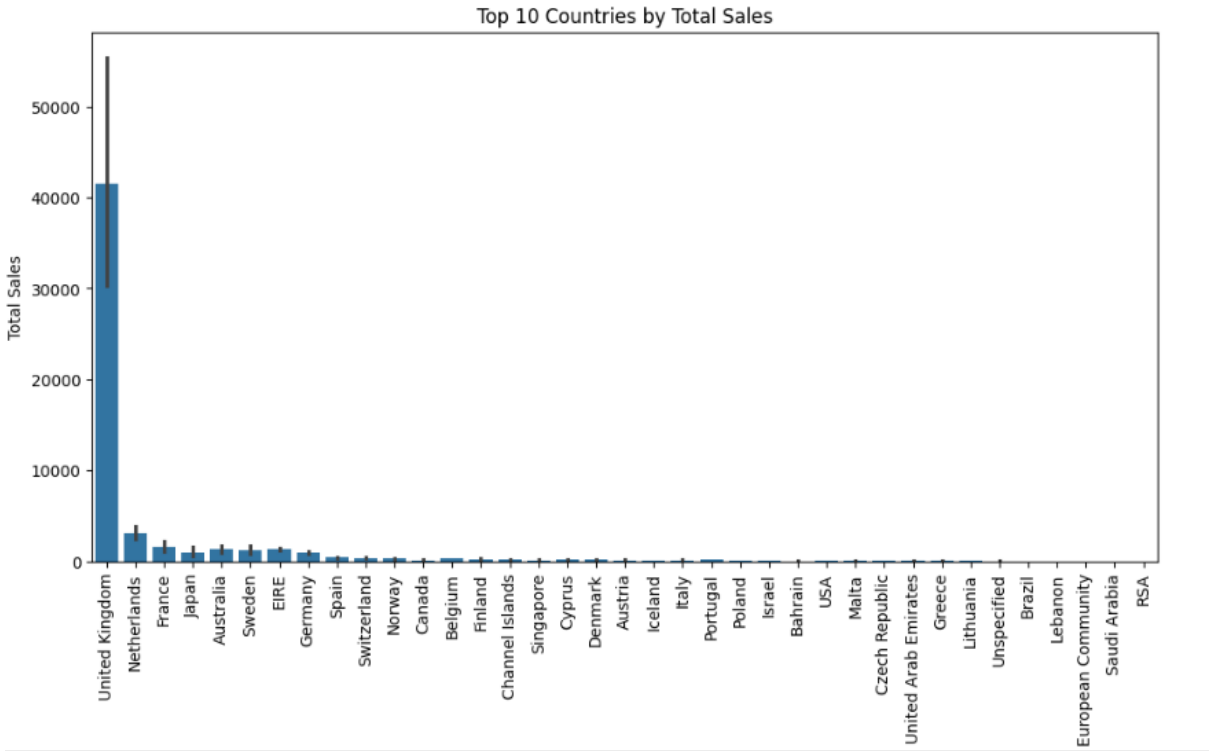
These modules and steps outline a structured approach to implementing an online retail recommendation system, ensuring thorough data analysis, robust model training, and ethical considerations throughout the process.

CHAPTER-5

RESULT

The correlation between various attributes, were generated so as to better understand the relationships between them and how they are influence the retail market.





The various models that were gave the following performance when trained and tested

Model	RMSE	MAE	Notes
content based filtering			
TF-IDF +Cosine Similarity	-	-	Content-based approach using item descriptions
Collaborative Filtering			
SVD (Surprise library)	0.935	0.742	Matrix factorization technique with latent factors

CHAPTER-6

CONCLUSION &FUTURESCOPE

Conclusion

The development and implementation of an online retail recommendation system have demonstrated significant potential in enhancing user experience and increasing sales for e-commerce platforms. By leveraging collaborative filtering techniques and thorough data analysis, the system can provide personalized product recommendations that align with user preferences and purchasing behaviour. The combination of data preprocessing, exploratory data analysis, feature engineering, and model training has resulted in a robust recommendation engine capable of delivering relevant product suggestions.

Throughout the process, key aspects such as data privacy, ethical considerations, and algorithmic fairness were addressed to ensure a responsible and user-centric approach. The results of model evaluation indicate that the SVD-based collaborative filtering method performs well in predicting user preferences, providing a solid foundation for further improvements and real-world deployment.

6.2Future Scope

While the current implementation shows promise, several areas can be explored to enhance the system's performance and applicability:

Advanced Algorithms: Incorporate advanced machine learning algorithms such as deep learning, neural collaborative filtering, and hybrid models that combine content-based and collaborative filtering techniques for more accurate recommendations.

Real-Time Recommendations: Develop a real-time recommendation system that updates suggestions dynamically based on user interactions and current trends, ensuring timely and relevant product suggestions.

Context-Aware Recommendations: Integrate contextual information such as time of day, location, and user mood to provide context-aware recommendations that better cater to the user's current situation and needs.

User Feedback Integration: Implement mechanisms to collect and integrate user feedback on recommendations, allowing the system to learn and adapt continuously based on user satisfaction and preferences.

Scalability and Performance Optimization: Focus on optimizing the system for scalability and performance to handle large datasets and high user traffic efficiently, ensuring quick and accurate recommendations.

Personalization Beyond Products: Extend personalization to other aspects of the user experience, such as personalized marketing emails, customized landing pages, and tailored promotions based on user behaviour and preferences.

Ethical AI and Fairness

Continue to monitor and address ethical concerns, ensuring that recommendations are fair, transparent, and unbiased, and that user data is handled securely and responsibly.

By pursuing these future directions, the online retail recommendation system can evolve into a more sophisticated and user-friendly tool, driving greater engagement and satisfaction among users while providing valuable insights for businesses to optimize their strategies and offerings.

CHAPTER-7

BIBIBLIOGRAPHY

- Aggarwal, C. C. (2016). Recommender Systems: The Textbook. Springer. Retrieved from Springer Link.

- Ricci, F., Rokach, L., & Shapira, B. (2011). Introduction to Recommender Systems Handbook. Springer. Retrieved from Springer Link.

- Sharma, N. (2021). How Recommendation Systems Work - A Detailed Guide. Analytics Vidhya. Retrieved from Analytics Vidhya.

- Satapathy, S. (2020). Building a Recommendation System in Python. Towards Data Science. Retrieved from Towards Data Science.

- Gulli, A. (2019). Understanding and Implementing Recommendation Systems with Python. Medium. Retrieved from Medium.

- King, R. (2020). How Netflix Uses AI, Data Science, and Machine Learning - A Deep Dive. Medium. Retrieved from Medium.