2024

Product Recommendation System

Prepared by

Karma Gurung

THE PROBLEM STATEMENT

Many e-commerce companies encounter difficulties in delivering personalized product recommendations to customers, leading to decreased engagement and missed sales opportunities. A successful recommendation system, powered by machine learning algorithms and customer data, is crucial for enhancing user engagement, improving user satisfaction, and driving sales growth. The goal of this project is to develop and deploy a tailored recommendation system within six months, with the aim of achieving a 5%-10% sales increase within a year through continuous optimization.

DATA SOURCES

The dataset provided includes purchase data, which comprises product details such as name, category, sub-category, and value, along with customer information such as their purchased items and ratings.
https://www.kaggle.com/datasets/shivanir23/online-retail-history?select=products.csv

METHOD

Three Main Types of Recommendation Systems:

Content-Based Filtering: This method recommends items similar to what a user has liked in the past. It analyzes product features (e.g., genre for movies, size for clothes) and recommends items with similar attributes. This approach works best when product profiles are well-defined and detailed.

Collaborative Filtering: This method identifies users with similar tastes and recommends products those users have rated highly. It relies on a large pool of explicit user ratings to establish user-to-user similarities. However, this can be a challenge for new customer bases where data is limited.

Hybrid Recommendation Systems: As the name suggests, these systems combine content-based and collaborative filtering techniques. This is a common approach. Initially, when a new user enters the platform, content-based filtering might be used due to the lack of user interaction data. As the user interacts with the platform and rates items, the system can transition to collaborative filtering to provide more personalized recommendations.

DATA CLEANING AND DATA WRANGLING

Here are the key steps:

- Data Loading & Inspection: Loaded customer and product data, inspected initial rows for understanding.
- Data Joining: Merged datasets based on product ID to create a comprehensive view of purchases.
- Data Exploration & Analysis: Analyzed data types, presence of null values, and summarized key statistics.
- Data Cleaning: Removed redundancies, renamed columns for clarity, and cleaned the price data (currency conversion, format).
- Data Verification: Confirmed data integrity through sampling and identified no missing values.

This wrangling process improved data quality by addressing inconsistencies, redundancies, and data type mismatches, resulting in a dataset ready for modeling or analysis.

In [7]:
```
# data distribution
df.describe().T
```

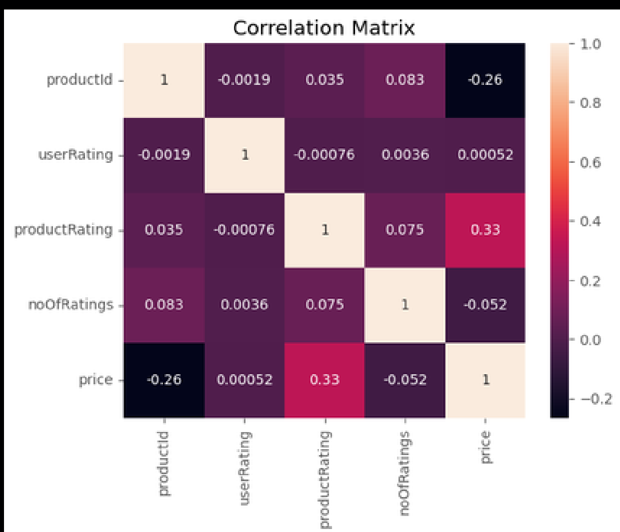
Out[7]:

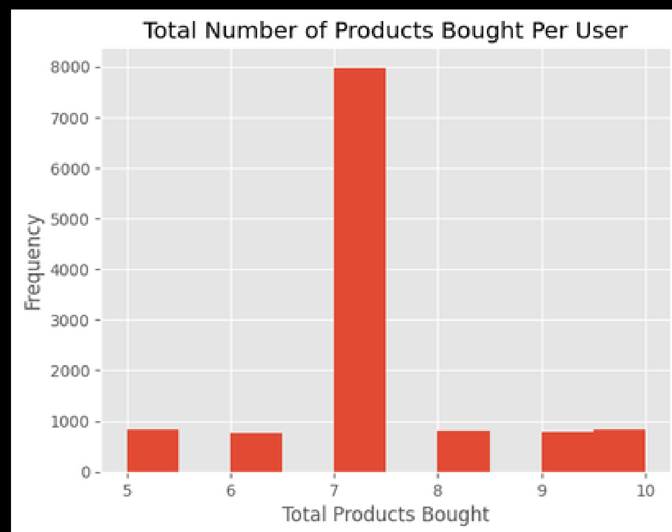
	count	mean	std	min	25%	50%	75%	max
user_id	86465.0	5996.300584	3463.975651	1.0	2995.0	5993.0	8996.0	12000.0
product_id	86465.0	25.561117	14.416922	1.0	13.0	26.0	38.0	50.0
user_rating	86465.0	3.004235	1.153766	1.0	2.0	3.0	4.0	5.0
Product_ID	86465.0	25.561117	14.416922	1.0	13.0	26.0	38.0	50.0
ratings	86465.0	4.034567	0.397229	2.5	4.0	4.1	4.3	4.6
no_of_ratings	86465.0	526.301290	283.258477	8.0	287.0	539.0	765.0	965.0

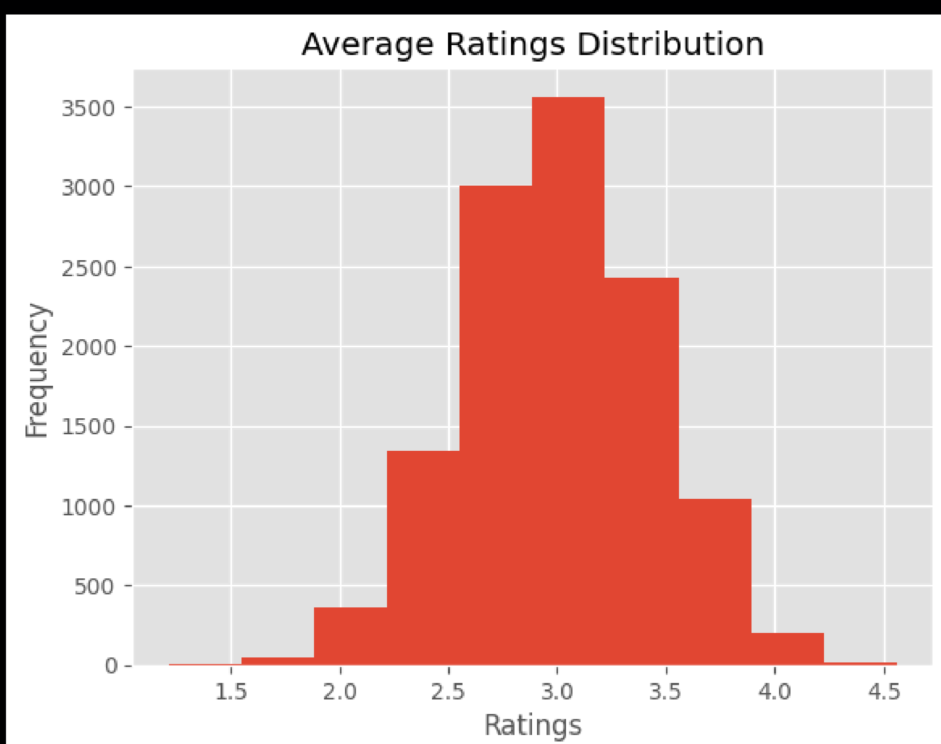
EDA

The EDA revealed:

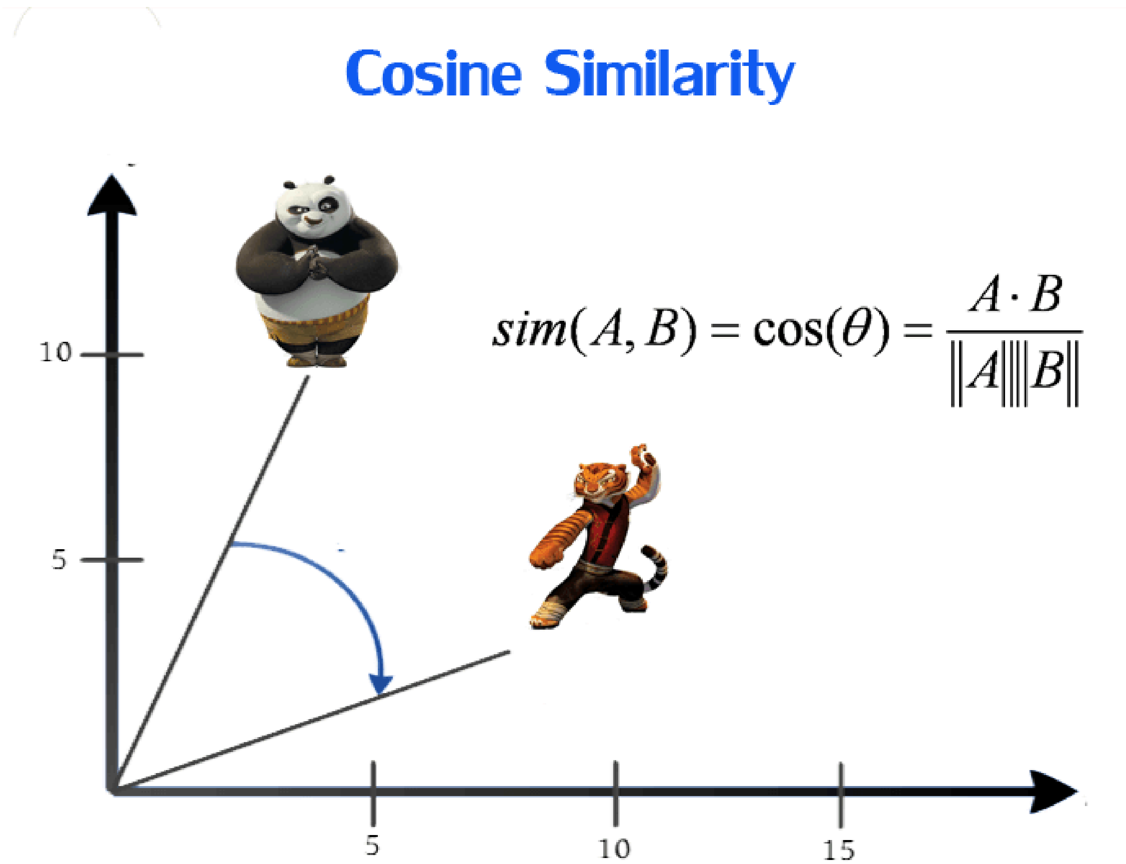
- A weak positive correlation between product rating and price.
- Users purchased an average of 7 products.
- User ratings averaged 3.0.
- Top purchased and top-rated products were identified.
- These insights inform customer behavior and product performance, aiding in recommendation systems and marketing strategies.







ALGORITHM AND RECOMMENDATION



This project leveraged collaborative filtering with cosine similarity to predict product recommendations.

It works by:

- Identifying similar users: Based on past interactions (purchases, ratings), the system finds users with similar tastes.
- Measuring item similarity: Cosine similarity, a mathematical technique, calculates how similar items are based on user ratings.
- Recommending similar items: Products with high cosine similarity to a user's liked items are recommended.

This approach personalizes recommendations based on user behavior, but may struggle with new users or limited data. Overall, it's a strong foundation for building a recommendation system.

PREDICTION EXAMPLE

PRODUCT RECOMMENDATION FOR USER_ID = 123

```
# 1 Example: Recommend products to user with userId = 123
user_id = 123
recommendations = recommend_products(user_id)
print(recommendations)
```

✓ 0.0s

```
productName
OnePlus Bullets Z2 Bluetooth Wireless in Ear Earphones with Mic, Bombastic Bass - 12.4 Mm Drivers, 10 Mins Charge - 20 Hrs
boAt Airdopes 141 Bluetooth Truly Wireless in Ear Earbuds with 42H Playtime, Beast Mode(Low Latency Upto 80ms) for Gaming,
boAt Rockerz 255 Pro+ in-Ear Bluetooth Neckband with Upto 40 Hours Playback, ASAP□ Charge, IPX7, Dual Pairing, BT v5.0, wi
Samsung Galaxy M33 5G (Mystique Green, 6GB, 128GB Storage) | 6000mAh Battery | Upto 12GB RAM with RAM Plus | Travel Adapte
Hammer Ace 3.0 Bluetooth Calling Smart Watch with Largest 1.85" IPS Display, Dual Mode, Spo2, Heart Rate, Strong Metallic
dtype: float64
```

FUTURE IMPROVEMENTS

Data Quality:
Prioritize datasets with richer features, particularly user ratings with detailed rating scales (e.g., star ratings, thumbs up/down) to provide more nuanced insights into user preferences.

Hybrid Recommendation System:
Implement a hybrid system that combines content-based filtering (utilizing categories, subcategories) with collaborative filtering (leveraging user ratings, gender, location) to provide more comprehensive and personalized recommendations.