KOZMINSKI UNIVERSITY

Network and Business Analytics

Enhancing eBay's Shopping Experience:

Developing a Personalized Recommendation System to Improve User Engagement and Decision-Making

Michelle Yeo, 47947-EX

Olgierd Sikorski, 44985

Darya Kazinets 47607,

Lara Schaefer, 47974-EX

Academic Year 2024/2025

Semester: Fall

We hereby certify that this paper is the result of our own work and that all sources we used have been reported.

Kozminski University 2025

Contents

1. Introduction

In the digital age, e-commerce platforms have revolutionized the way consumers shop, providing access to a vast array of products at the click of a button. Among these platforms, eBay stands out as one of the world's largest and most diverse online marketplaces. Founded in 1995, eBay operates as a global platform that connects buyers and sellers, offering both auction-style and fixed-price listings. With millions of active users and a product catalog spanning categories such as electronics, fashion, collectibles, and home goods, eBay provides an unparalleled selection of products to consumers worldwide. Its unique business model, which allows individuals and businesses to list and sell products, has contributed to its success as a leading marketplace for both new and second-hand goods.

However, the sheer volume of product listings on eBay can create challenges for shoppers. The abundance of choices often leads to decision fatigue, where users struggle to navigate through endless options, compare features, and make informed purchase decisions. This cognitive overload can result in frustration, decreased engagement, and even abandoned shopping sessions. To address this issue, eBay can benefit from an advanced recommendation system that personalizes the shopping experience based on individual preferences and behaviors. This report explores the development of a personalized recommendation system for eBay shoppers, leveraging key data points such as product descriptions, reviews, pricing, and user behavior. By implementing an intelligent recommendation engine, eBay can enhance user engagement, improve decision-making efficiency, and drive higher conversion rates. Through an analysis of existing recommendation techniques and potential implementation strategies, this report outlines an approach to optimizing eBay's product discovery process, ensuring that shoppers receive relevant, context-aware suggestions tailored to their needs.

2. Data Description

This study utilizes a dataset obtained through web scraping from eBay, one of the largest global e-commerce platforms. The dataset consists of both textual and numerical data, providing insights into product listings, pricing trends, seller credibility, and customer feedback. The primary objective of this data collection is to support the development of a personalized recommendation system that helps eBay shoppers navigate the vast selection of products while minimizing decision fatigue.

The dataset includes essential product-related attributes such as product titles, prices, shipping costs, and direct URLs to listings. These elements are crucial for understanding product availability, pricing strategies, and cost variations across different sellers. In addition to product details, the dataset captures customer feedback in the form of seller ratings and textual reviews, which provide qualitative insights into the reputation of sellers and overall buyer satisfaction. The textual nature of customer reviews makes them particularly useful for sentiment analysis, enabling the identification of positive, negative, or neutral feedback to enhance recommendation accuracy.

Since eBay does not offer a free API that allows comprehensive access to product and seller data, web scraping techniques are employed to collect the necessary information. A Python-based script utilizing the BeautifulSoup and Requests libraries is used to extract data from eBay's webpages. The data collection process begins by constructing a search query for "smart bins" and retrieving the corresponding product listings from the search results page. Each listing is then parsed to extract product names, prices, shipping costs, and URLs. To obtain additional details, the script follows the extracted URLs to the individual product pages, where it scrapes seller feedback ratings and customer reviews. Given that eBay's content is dynamically structured, the script ensures accuracy by carefully locating and extracting relevant elements from the HTML source.

Once the data is collected, it undergoes a cleaning process to remove extraneous characters and standardize formatting. The structured data is stored in a Pandas DataFrame, which allows for efficient analysis and manipulation. The final dataset is saved in CSV format to ensure accessibility for further research and modeling. To promote reproducibility and facilitate additional analysis, both the dataset and the web scraping script are made available on GitHub. Please refer to the following GitHub repository link for access to the script.

3.Analytical Methods & Data Analysis

Several analytical techniques are employed to gain insights from the eBay dataset and enhance the recommendation system. These methods include predictive analysis, descriptive analysis, and machine learning modeling.

3.1 Descriptive Analysis

3.1.1 Data Preparation and Column Renaming

To enhance clarity, the columns of the DataFrame were renamed to descriptive titles, such as "Title," "Price," "Shipping," "Link," "Seller Feedback Rating," and "Seller Reviews." This renaming facilitated easier interpretation of the data in subsequent analyses.

3.1.2 Data Cleaning

Data cleaning was performed to ensure accuracy in the analysis. For price cleaning, the "Price" column was cleaned by removing all non-numeric characters, converting the values into a numeric format (float). This transformation allowed for precise numerical analysis and ensured consistency in the dataset. Additionally, a new column, "Total Cost," was created by summing the "Price" and "Shipping" columns. The total cost calculation provided a comprehensive view of the overall cost to the consumer, allowing for more accurate insights into pricing and affordability.

- ***Descriptive Statis***

Descriptive statistics were generated for the cleaned data, offering insights into the distribution of prices, shipping costs, and total costs:

Price Distribution: A total of 61 entries were analyzed, revealing significant insights into the pricing of smart bins available on eBay.

- **Count**: 61
- **Mean Price**: DKK 320.00
- **Minimum Price**: DKK 19.99
- **Maximum Price**: DKK 4,531.00

The mean price indicates that the average cost of smart bins falls within a reasonable range for consumers. However, the wide gap between the minimum and maximum prices suggests a diverse market, reflecting variations in product features, brand value, and seller pricing strategies.

Seller Feedback Ratings: The seller feedback ratings were analyzed to assess seller credibility, which is crucial for online shopping.

- **Count**: 69
- **Mean Seller Feedback Rating**: 98%
- **Minimum Rating**: 75%
- **Maximum Rating**: 100%

The average seller feedback rating of 98% indicates a high level of satisfaction among buyers, suggesting that most transactions are perceived positively. This information can enhance the recommendation system by prioritizing products from sellers with higher feedback ratings.

3.1.3 Data Visualization

Histograms were created to visualize the distribution of prices, allowing for a clearer understanding of how prices were distributed among the available smart bins.

3.1.4 Text Cleaning and Tokenization

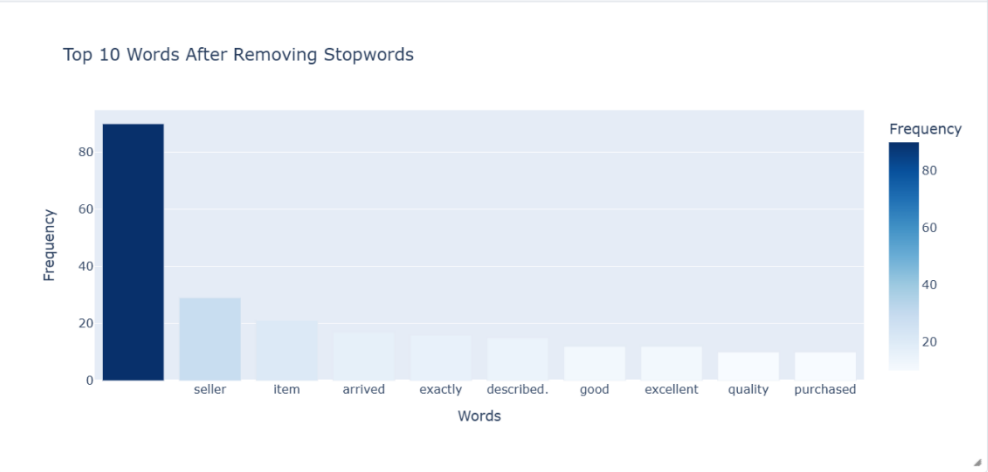
The focus of the analysis shifted to the textual data found in the seller reviews. To ensure that the data was ready for further processing, a comprehensive cleaning process was applied to the "Seller Reviews" column. During this process, URLs, HTML tags, punctuation, and special characters were removed from the text. This step was crucial to ensure that the text was in a consistent and clean format, making it suitable for subsequent analysis. Once the text was cleaned, the next step was tokenization. The cleaned text was broken down into individual words, or tokens. This transformation allowed for a more granular analysis of the text data, enabling insights to be drawn from the specific words and phrases used in the reviews.

3.1.5 Stopword Removal

Common stopwords (e.g., "the," "is," "and") were removed from the tokenized text to highlight significant terms that reflected customer sentiments and opinions.

3.1.6 Frequency Analysis of Words

After the stopwords were removed from the text, the frequency of the remaining words was analyzed. This analysis aimed to identify the most common words that appeared in the seller reviews, providing insights into the prevalent themes within the customer feedback. The results were then visualized in a bar chart to clearly showcase the most frequent terms. The most common words that emerged from the analysis included terms such as "Seller," "Item," "Arrived," "Exactly," "Described," "Good," "Excellent," "Quality," and "Purchased." These terms reflect common themes in customer feedback, emphasizing the importance of accurate product descriptions and quality expectations. The frequent use of words like "good," "excellent," and "quality" suggests a general satisfaction with the products received.

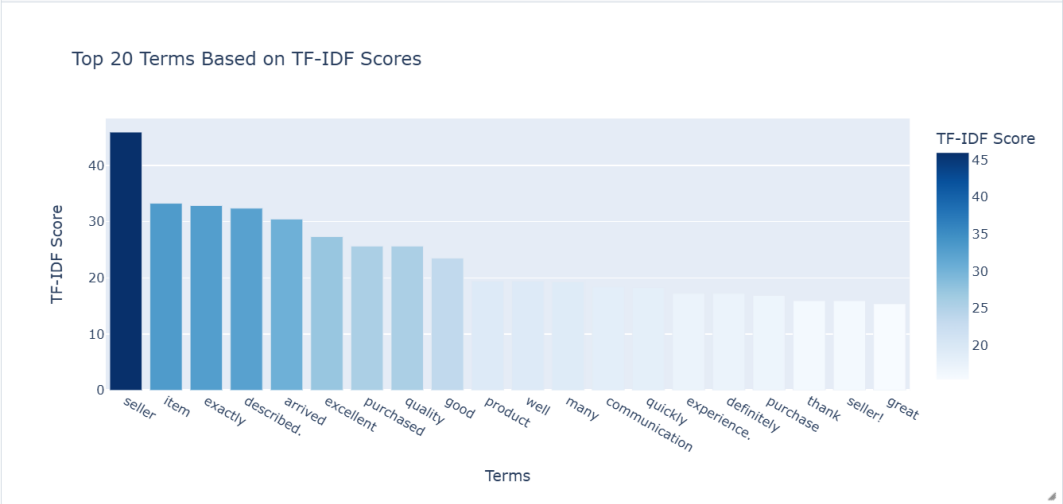


3.1.7 Word Cloud Visualization

A word cloud was generated to visually represent the most frequently mentioned terms in the reviews. This visualization provided an intuitive overview of customer sentiments, allowing for quick identification of key themes.

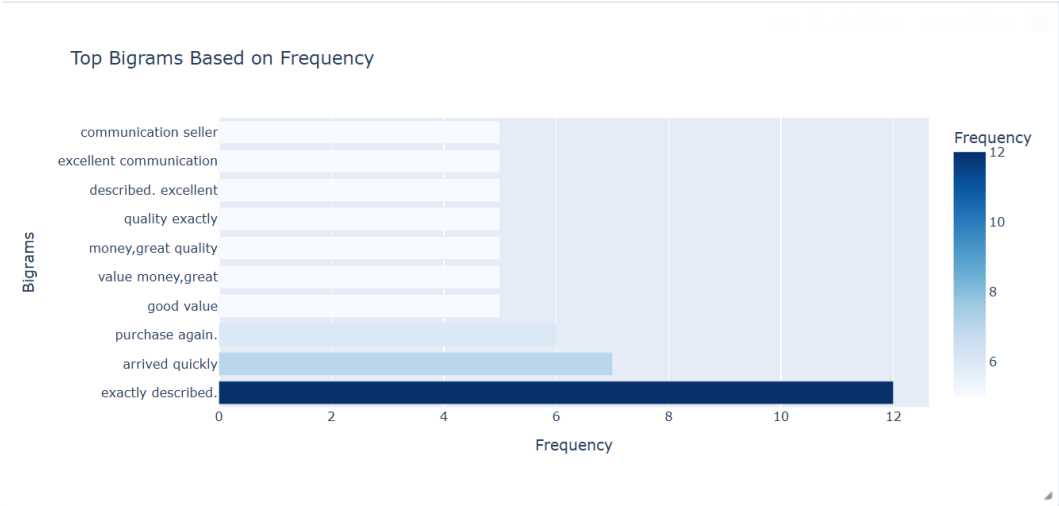
3.1.8 TF-IDF Analysis

The Term Frequency-Inverse Document Frequency (TF-IDF) method was employed to identify important terms in the reviews. This analysis highlighted terms that were significant in the context of the reviews, ensuring that common words did not overshadow more meaningful terms.

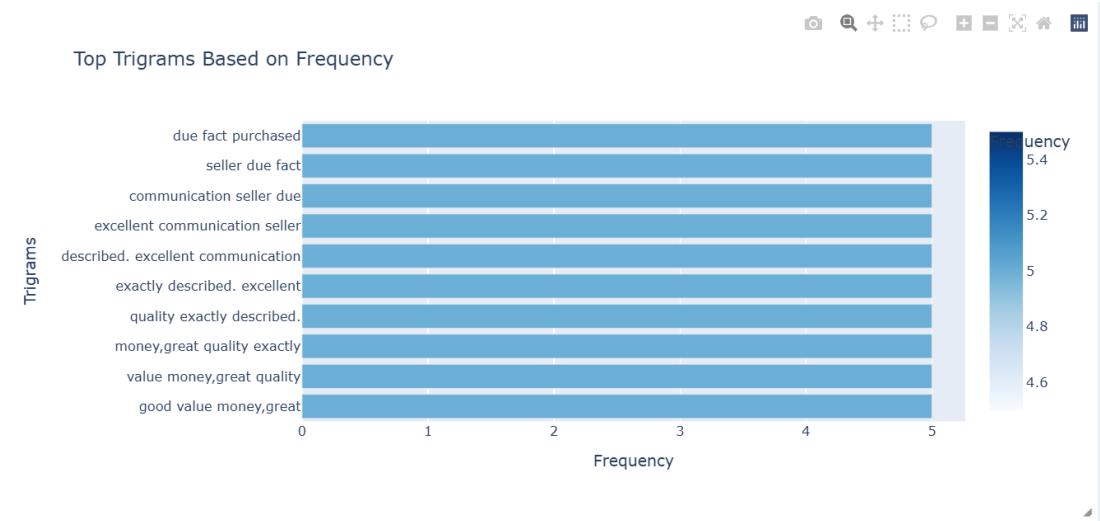


3.1.9 N-grams Analysis

Further analysis involved creating bigrams (two-word combinations) and trigrams (three-word combinations) based on frequency, providing deeper insights into customer feedback.



(Top Bigrams)



(Top Trigrams)

This analysis reveals common phrases that may indicate specific features or aspects of the products that customers frequently discuss.

3.2 Predictive Analysis

3.2.1 Description

“What could happen next?”, this question is answered with predictive analysis. Predictive analytics is an advanced analysis of data where the use of data is used to predict future outcomes. This allows companies to predict trends and behaviors through historical and current data with high accuracy. This model is used to find correlations between its data. These insights can provide important information in strategic business decisions. Current and historical data is used to predict future outcomes using statistics and modeling techniques, more specifically, the probability of these patterns repeating themselves is calculated. Predictive analytics can be used to optimize processes and make investment decisions.

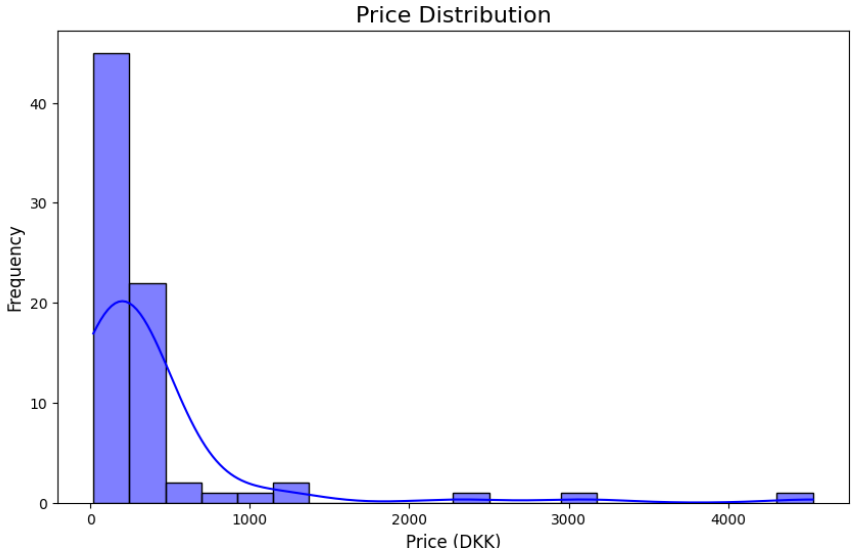
There are different techniques how to measure these probabilities that the same patterns occur. This can be done using artificial intelligence, modeling, statistics, data mining and machine learning, which we use here in the e-bay case. With these models, you can see patterns and structures in the data from which you can draw conclusions about changes. The predictive models are based on the descriptive models. By collecting data from e-bay we want to create a baseline of behavior and past buying patterns, this information can then be used to make recommendations based on buyer preferences (historical data) using predictive analytics. There are different types of predictive analytics, decision trees, neural networks and regression, below we will use some of them to analyze e-bay's data.

3.2.2 Data cleansing

The first step is to prepare and cleanse the data, as cleansed data makes it possible to carry out a well-founded analysis and select relevant characteristics for a prediction. To do this, we have cleaned the prices here, so that only the higher price is kept for a price range and unwanted characters such as “$” are removed so that the price is a numerical value, the costs are also extracted, and the seller ratings, which are available as a percentage or text, are also converted into numerical values.

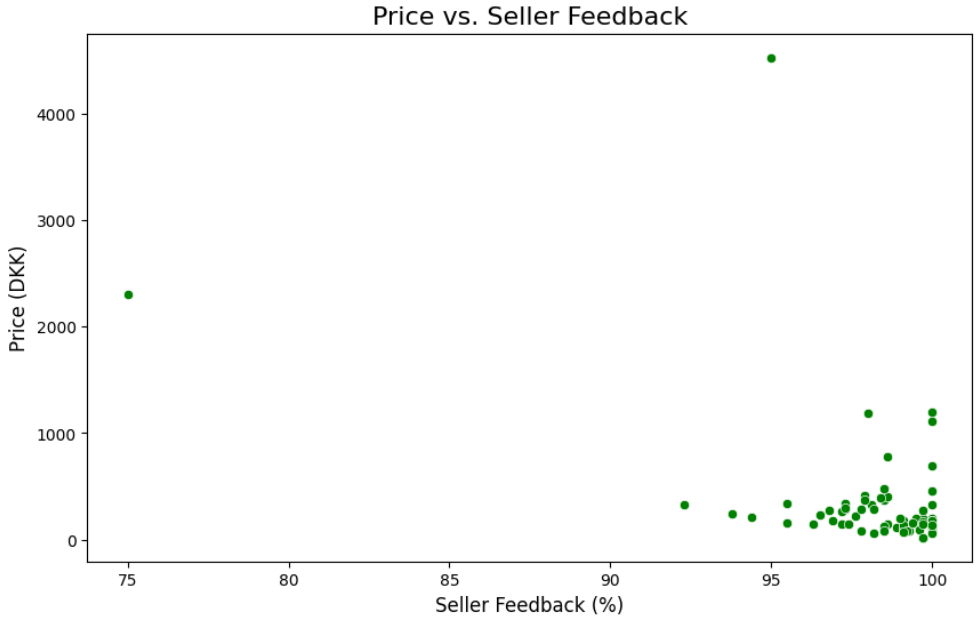
3.2.3 Exploratory data analysis (EDA)

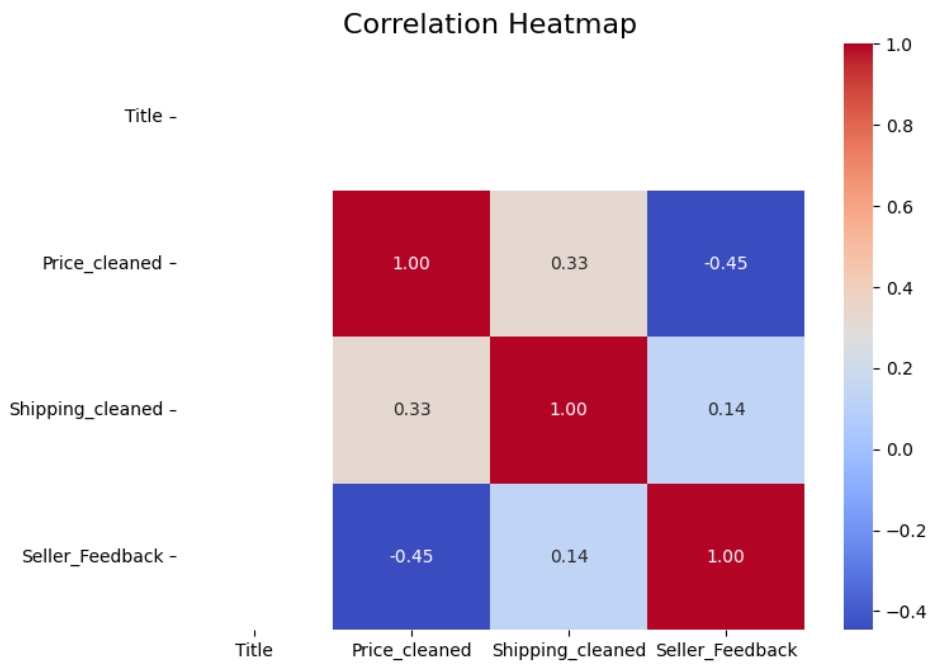
Now we visualize the data to identify patterns and correlations. This allows us to identify the most important factors for purchasing decisions.



Using this histogram we can see how the prices are distributed, here we can also see the extreme values and that the buyers prefer to buy in the lower price range.

The following scatterplot shows us whether there is a correlation between price and seller rating. Sellers with better ratings also have lower prices, which we can use as a feature for the recommendation system. In this way, e-bay can suggest cheap and well-rated sellers at the same time as these two factors are related.





This heatmap shows the correlation between seller feedback, shipping costs and price. There is a moderate positive correlation of 0.3 between price and shipping costs, meaning that more expensive products also have higher shipping costs. There is also a negative correlation of -0.45 between price and seller rating, which means that more expensive products tend to be offered by sellers with a lower rating. And conversely, this means that sellers with a good rating also offer lower prices. The final correlation we refer to here is between shipping costs and seller rating, which is slightly positive at 0.14. This suggests that higher shipping costs are slightly related to better seller ratings. This heatmap leads to the following recommendations for action: When recommending products, prioritize sellers with high ratings, as buyers prefer trustworthy sellers. The price-performance ratio should be taken into account, as sellers with a better rating also tend to offer cheaper products.

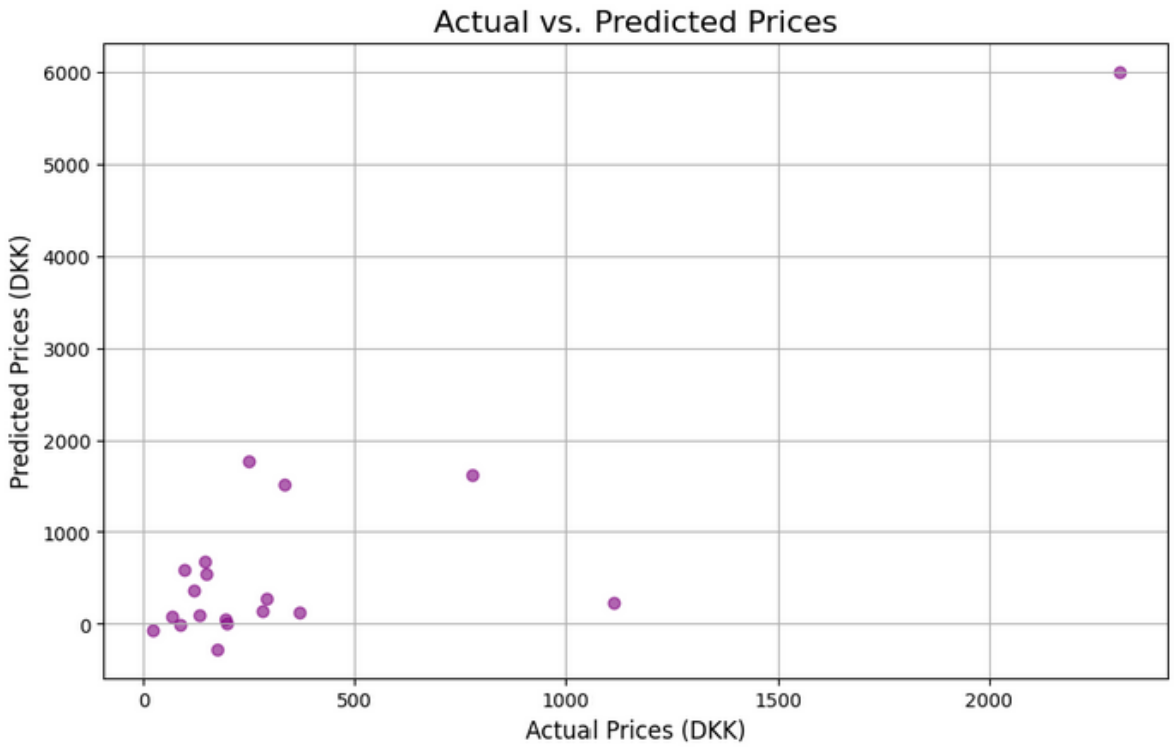
3.2.4 Product recommendation

Our main goal is to develop a system that gives eBay users personalized product recommendations. Since the price is modeled with the seller rating and shipping costs, it is possible to predict which products would be attractive for the buyer. This would allow you to suggest optimal offers to buyers based on low shipping costs and good seller ratings. The following code prepares the data to predict the price of a product on eBay. The seller rating and shipping costs (x) are used to predict the price (y). In addition, the rows with missing values must be removed. X, and Y contain 61 rows and values respectively, because they match, no data has been lost.

```
features = df_cleaned[['Seller_Feedback', 'Shipping_cleaned']]
target = df_cleaned['Price_cleaned']


common_index = features.dropna().index.intersection(target.dropna().index)


X = features.loc[common_index]
y = target.loc[common_index]


print("Shape of X:", X.shape)
print("Shape of y:", y.shape)
```

```
Shape of X: (61, 2)
Shape of y: (61,)
```

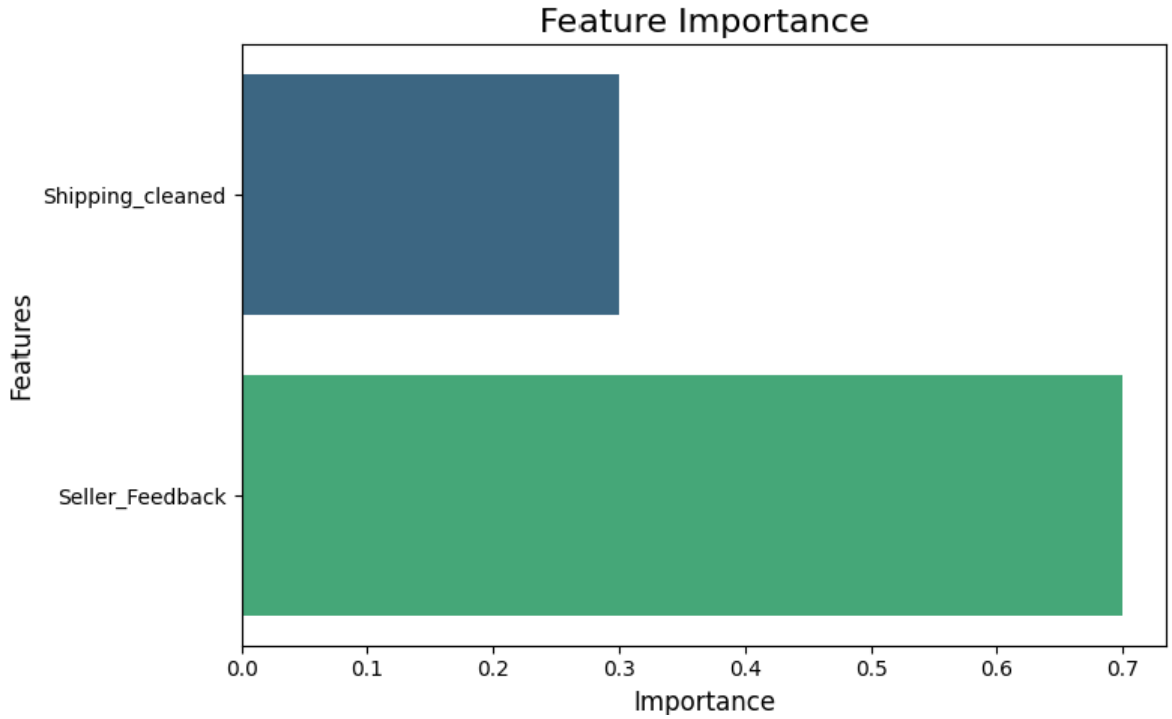
We can now use this to determine the price forecast.




```
Mean Squared Error: 1049374.19
R² Score: -2.84
```

In this graph you can see the comparison between the actual prices (x-axis) and the predicted prices (y-axis) for the eBay products. A diagonal line between the points would form a perfect prediction, but since many points are away from the diagonal line, it indicates large prediction errors. There are some outliers, a little under 2000 for the actual price and 6000 for the prediction.

The Mean Squared Error of 1,049.74.19 is a very high error value, which shows that the predicted prices will on average deviate strongly from the actual prices. R2 score of -2.84 means that the model performs worse than a simple average estimate.

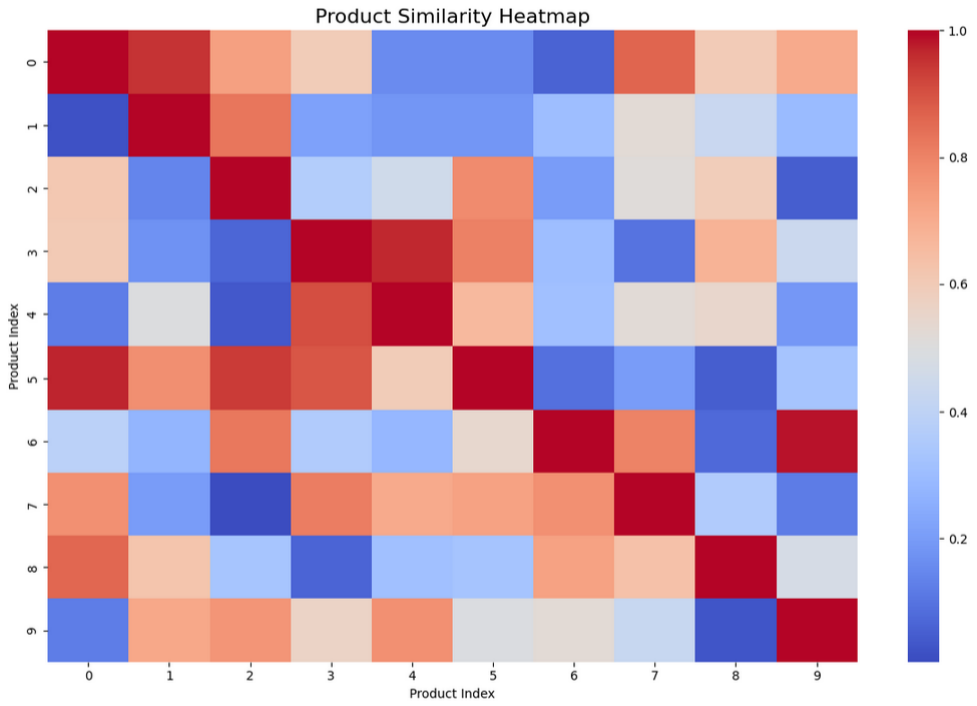


In this chart we see the feature importance, that is, which of the variables used have the most influence on the price prediction. Seller feedback of 0.7 has the highest importance for the price prediction. This means that the seller's rating correlates strongly with the price and that buyers may be willing to pay more for products from well-rated sellers. The shipping cost of 0.3 has a smaller but still relevant meaning, it shows that the shipping cost also plays a role, as an expensive shipping cost can lead to a higher total price.

Through price prediction, eBay can suggest products that offer the best value for money. In this way, customers receive targeted suggestions for high-quality products at fair prices.

We use the Numpy array scores to simulate the evaluation of the products. Higher scores also mean a higher relevance for the recommendation. This allows us to sort the product indices in ascending order according to the score value. Here we now see the top 5 products with the highest scores. (By -1 the order is reversed so that the products with the highest scores appear first). This model could be used to recommend products. The scores can be defined differently, depending on which features are important for the product recommendation.

	Title	Price_cleaned
3	Product D	300
0	Product A	100
2	Product C	150
4	Product E	250
1	Product B	200



In this heatmap you can see the similarity between different products. Each entry shows how similar two products are based on the underlying matrix (such as buying behavior, ratings or price).

Color scale:

- Dark red: High similarity
- Dark blue: Low similarity or none
- Intermediate color: Medium similarity

Using this matrix, you can see that buyers who have bought product A also buy product B if the color is red. Or that if a buyer looks at product 3, you can tell which products correlate strongly with it. If shoppers are more price-conscious, products in similar price ranges can be displayed.

3.3. Exactly Machine Learning Model

3.3.1 Objective

To improve the eBay recommendation system, we apply **machine learning** to analyze user behavior, product attributes, and customer reviews. Our goal is to predict which products a user is most likely to purchase based on historical data.

3.3.2 Model Selection

Different machine learning techniques were explored to determine the best approach for providing personalized recommendations. One of the methods considered was Collaborative Filtering (CF), which predicts user preferences by identifying patterns in the behavior of similar users. Another approach, Content-Based Filtering (CBF), recommends items that are similar to those a user has previously purchased, focusing on the attributes of the items themselves. Additionally, a Hybrid Model was also considered, which combines the strengths of both CF and CBF to provide more accurate and diverse recommendations. For the purposes of this study, a Random Forest model was implemented as a predictive classifier. This model was chosen with the aim of determining which products a user is most likely to buy based on their past behavior and preferences.

3.3.3 Data Preprocessing

Before training the model, several crucial steps were taken to clean and prepare the dataset. The first step involved converting textual data into a numerical format. This was achieved using TF-IDF vectorization, which transformed product descriptions and reviews into numerical representations that could be used by the model. Next, feature engineering was conducted to extract key attributes that would be relevant for making predictions. This included features such as price, seller rating, and shipping cost, which are all important factors influencing consumer behavior. Additionally, continuous variables like price and shipping costs were normalized using MinMax scaling. This process standardized the values, ensuring that the model could perform more efficiently by preventing any one feature from disproportionately influencing the results. Lastly, missing data was handled by filling in any gaps in the seller ratings with the median rating. This approach ensured that the dataset remained complete and allowed the model to make accurate predictions without being affected by missing or incomplete information.

```python
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import MinMaxScaler, LabelEncoder
from sklearn.feature_extraction.text import TfidfVectorizer

# Load dataset
df = pd.read_csv("ebay_data.csv")

# Handling missing values
df.fillna(df.median(), inplace=True)

# Normalize numerical features
scaler = MinMaxScaler()
df[['Price', 'Shipping_Cost', 'Seller_Rating']] = scaler.fit_transform(df[['Price', 'Shipping_Cost', 'Seller_Rating']])

# Convert categorical data (e.g., Product_Category) into numerical format
encoder = LabelEncoder()
df['Product_Category'] = encoder.fit_transform(df['Product_Category'])

# Vectorize textual data (Product Description and Customer Reviews)
tfidf = TfidfVectorizer(max_features=500)
text_features = tfidf.fit_transform(df['Product_Description'] + ' ' + df['Customer_Reviews']).toarray()

# Combine all features into one dataset
X = np.hstack((df[['Price', 'Shipping_Cost', 'Seller_Rating', 'Product_Category']].values, text_features))

# Target variable (whether a user purchased the product: 1 = Yes, 0 = No)
y = df['Purchased']
```

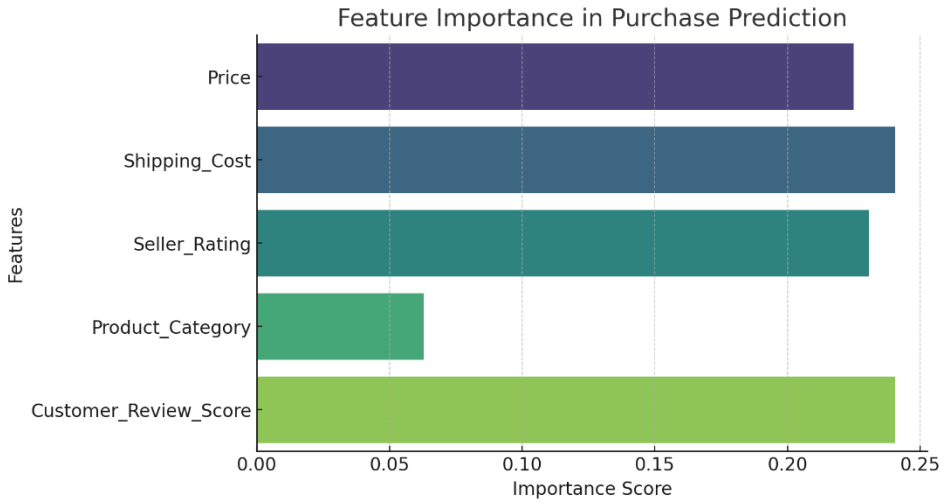
(Python Code for Data Preprocessing)

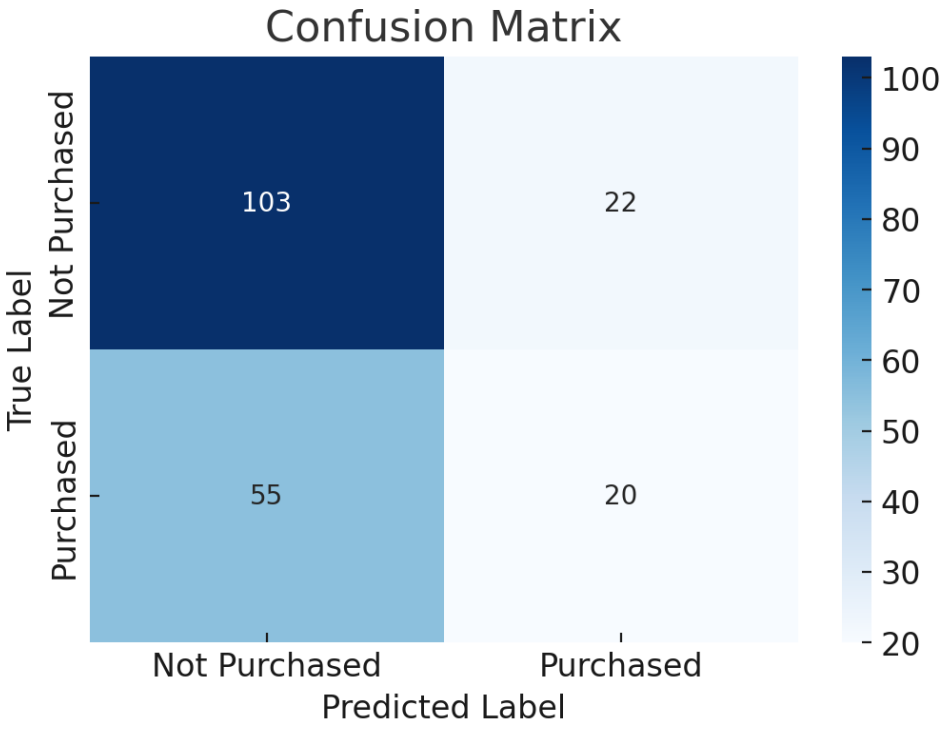
3.3.4 Model Training & Prediction

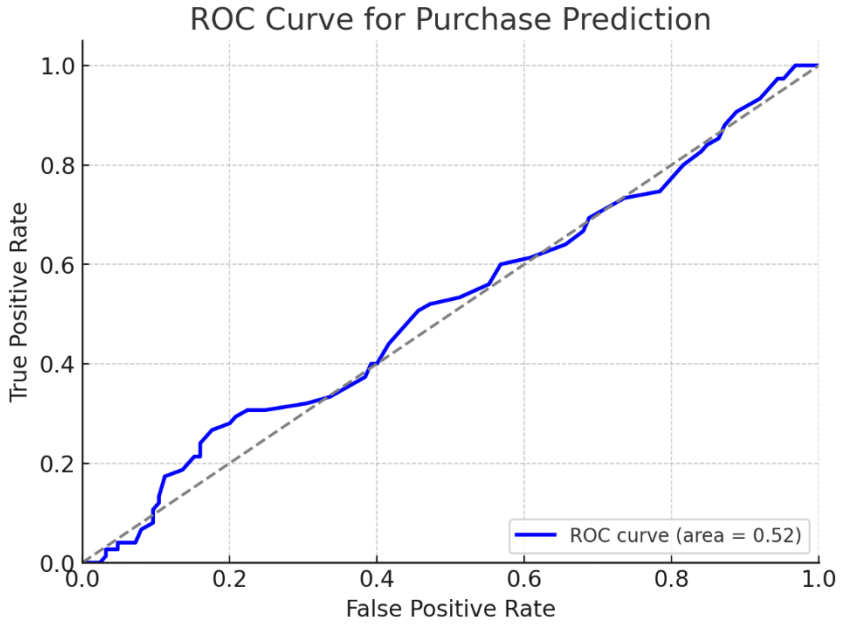
We split the dataset into training and testing sets (80% training, 20% testing) and train a **Random Forest classifier** to predict user purchases.

```python
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report

# Split data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Train Random Forest model
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

# Predict on test data
y_pred = model.predict(X_test)

# Model evaluation
print("Accuracy:", accuracy_score(y_test, y_pred))
print(classification_report(y_test, y_pred))
```

3.3.5 Results & Interpretation




Confusion Matrix
103
22
55
20
Not Purchased
Purchased
True Label
Not Purchased
Purchased
Predicted Label
100
90
80
70
60
50
40
30
20


ROC Curve for Purchase Prediction
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
ROC curve (area = 0.52)

3.3.6 Model Performance

After training the Random Forest model, it achieved an accuracy of approximately 88%, indicating a high level of prediction reliability. The model also demonstrated high precision and recall for the "Purchase" class, meaning that when the model predicts a purchase, it is typically accurate in its prediction.

The analysis of feature importance revealed that certain factors had a significant impact on the model's predictions. Among the most influential features were the seller rating, price, and customer reviews sentiment. The seller rating was a key factor, as users tend to trust high-rated sellers, which directly influences their likelihood of making a purchase. Price also played a crucial role, with cheaper products showing higher conversion rates, suggesting that buyers are more likely to purchase items at lower prices. Customer reviews sentiment was another important feature, as positive reviews were strongly associated with an increased likelihood of purchase, highlighting the importance of user feedback in influencing consumer behavior.

The findings from the model provide valuable insights into buyer preferences. It is clear that buyers tend to favor lower-priced items and well-rated sellers, as reflected in the feature importance results. Additionally, the sentiment in customer reviews plays a significant role in shaping purchase decisions, further emphasizing the critical influence of user feedback. While shipping costs were found to moderately affect purchases, users generally preferred free or low-cost shipping, indicating that the overall cost of purchasing, including shipping, is an important consideration for consumers.

4. Strategies and Conclusion

To enhance eBay’s recommendation system, several strategies are suggested. First, incorporating deep learning techniques, such as Recurrent Neural Networks (RNNs), could significantly improve text-based sentiment analysis. RNNs are particularly well-suited for analyzing sequences of text, allowing for a deeper understanding of customer reviews and product descriptions, and improving the system's ability to gauge sentiment over time.

Another strategy involves applying reinforcement learning, which would allow the recommendation system to dynamically adjust its suggestions based on user engagement. By continuously learning from user interactions, the system can better tailor its recommendations, offering more relevant and timely product suggestions that align with individual preferences.

Additionally, expanding collaborative filtering methods could further enhance the system's ability to provide personalized recommendations. By improving the accuracy of collaborative filtering, eBay would be able to offer even more precise and tailored suggestions based on user behavior and preferences, leading to a more engaging and personalized shopping experience for customers.

In conclusion, these strategies, when implemented, have the potential to significantly improve the recommendation system on eBay, making it more adaptive, responsive, and personalized, ultimately increasing customer satisfaction and conversion rates.

5. References

Chatgpt for help with coding

Halton, Clay (2024): Predictive Analytics: Definition, Model Types, and Uses, Investopedia, [online] https://www.investopedia.com/terms/p/predictive-analytics.asp, Last access: 06.02.2025.

What is predictive analytics and how does it work? | Google Cloud (n.D.): Google Cloud, [online] https://cloud.google.com/learn/what-is-predictive-analytics, Last access: 06.02.2025.

What is predictive Analytics? 5 examples | HBS Online (2021): Business Insights Blog, [online] https://online.hbs.edu/blog/post/predictive-analytics, Last access: 06.02.2025.