Assignment 2

Group Members:

Diwakar Prajapati 220382
Shivam Kumar 221013

Indian Institute of Technology Kanpur

CGS616: Human Centered Computing

Instructor: Prof. **Pragathi Balasubramani**

Github Link for Code: Click here

1 Overview

This report details the relationship between reviews and ratings to identify anomalies and also relates reviews to price of items. We extracted the *AllBeauty* **.json** file for the data from the given link provided in problem statement, then developed a Python script that clean the data, analyze it with our own ideology and gives four different visualisations.

2 Aim

To identify anamolies based on reviews & ratings and also the relation between anamolies and price of items.

3 Python Libraries Used

```
import os
import re
import html
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from textblob import TextBlob
```

- **pandas:** We used pandas for manipulating data and further analysis.
- **matplotlib.pyplot:** This library helps us to create different plots.
- **seaborn:** For more convinient statistical graphs.
- **textblob:** To compute sentiment polarity for each review, it ranges from -1 to 1.

4 Logic Used

```
df['rating_z'] = (df['rating'] - df['rating'].mean()) / df['rating'].std()
df['sentiment_z'] = (df['sentiment'] - df['sentiment'].mean()) / df['sentiment'].std()
df['composite_score'] = df['rating_z'] + df['sentiment_z']

median_composite = df['composite_score'].median()
q1 = df['composite_score'].quantile(0.25)
q3 = df['composite_score'].quantile(0.75)
iqr = q3 - q1

k = 2.5
lower_bound = median_composite - k * iqr
upper_bound = median_composite + k * iqr

df['anomaly_optimized'] = df['composite_score'].apply(
    lambda x: 1 if x < lower_bound or x > upper_bound else 0
)
```

- **Idea:** The idea behind this method is that instead relying on sentiment score and ratings, we can do composite score for better analysis. We first normalizes sentiment scores and ratings using z-score normalization and then compute the composite score. Then we use IQR which helps us to define a normal range of composite score and add robustness in detecting anomalies. If the composite score for a review falls in that range then its okay and if falls outside then it is an anomaly. There may be situations where false reviews and ratings would be given, like the rating would be high and review is also very good but it may be a false marketing technique used.
- **Limitations:** It may not be a perfect way to identify anamoly, some anamolies detection may be false but it is better than to just compare sentiment of reviews and ratings.

4.1 Z-score normalization

```
df['rating_z'] = (df['rating'] - df['rating'].mean()) / df['rating'].std()
df['sentiment_z'] = (df['sentiment'] - df['sentiment'].mean()) / df['sentiment'].std()
```

- Z-score normalization standardizes a variable by subtracting the mean and then dividing by the standard deviation. This process adjusts the data so that it has a mean of 0 and a standard deviation of 1, allowing values on different scales to be compared more easily. This method ensures that both ratings and sentiment scores are treated equally when calculating the composite score for anomaly detection.

$$Z = \frac{X - \mu}{\sigma}$$

- Where:
  X: The original value.
  μ: The mean of the column.
  σ: The standard deviation of the column.

4.2 Composite Score and IQR-based Threshold Computation

```
df['composite_score'] = df['rating_z'] + df['sentiment_z']
median_composite = df['composite_score'].median()
q1 = df['composite_score'].quantile(0.25)
q3 = df['composite_score'].quantile(0.75)
iqr = q3 - q1
```

- The interquantile range (IQR) represents the difference between the 75th percentile (q3) and the 25th percentile (q1) of a dataset. It helps measure the spread of the middle 50% of the data, making it a useful tool for identifying outliers. Since the IQR focuses on the central portion of the dataset, it is less affected by extreme values.
- We can change q1 & q3, which makes the range either wider or narrower making anamoly detection vulnerable, hence we used 75th & 25th percentile so we measure the middle 50% spread.

4.3 Setting the Anamoly Boundary and Assigning Anamoly Flag

- We used median composite score here and not mean composite score to get more robustness in our results.
- A multiplier k is used to set the range, we used $k = 2.5$ here which was giving satisfactory result. Earlier we used $k = 1.5$ which made the range narrower giving too much anomalies.
- If composite score for a review is outside this range then it is marked as 1 (anomaly), otherwise 0.

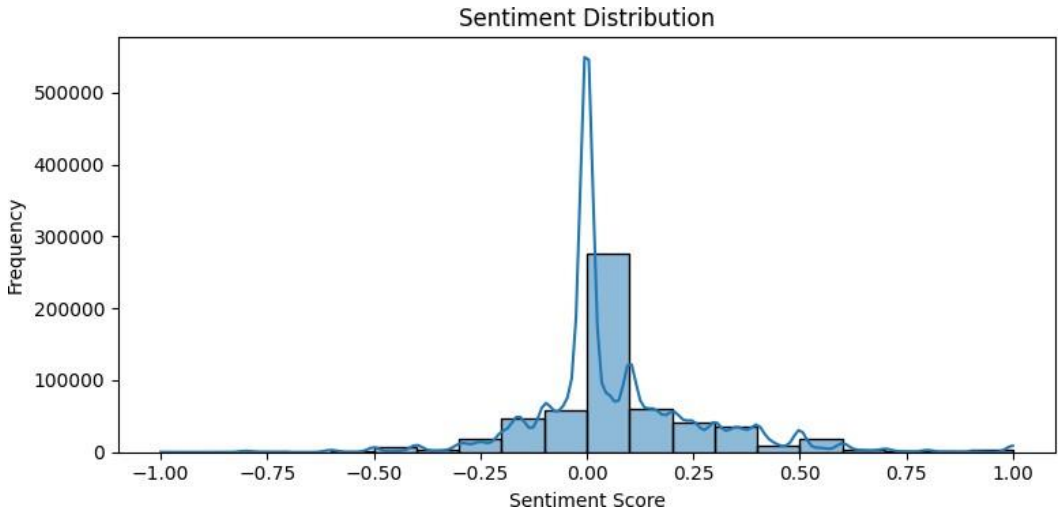
```
k = 2.5
lower_bound = median_composite - k * iqr
upper_bound = median_composite + k * iqr

df['anomaly_optimized'] = df['composite_score'].apply(
    lambda x: 1 if x < lower_bound or x > upper_bound else 0
)
```

5 Script Workflow

- We used different libraries for different uses as referenced above in section 3 - Python Libraries Used and creates a output folder to save visualisations and the processed data.
- The script imports the merged CSV into a DataFrame, detects the review text column automatically (such as ”text,” ”review-text,” or ”title”), and processes the content by eliminating HTML tags, newlines, and unnecessary spaces.
- Then it analyzes the sentiment of each review using TextBlob, standardize both the rating and sentiment with z-scores, and merge them into a composite score. Reviews are marked as anomalies if their composite score deviates beyond the median ± 2.5×IQR.
- Then the script exports the processed DataFrame as a CSV file and creates essential visualizations, which includes a sentiment histogram, a rating boxplot, a scatter plot for price vs. anomalies, and a correlation heatmap.

6 Results and Analysis

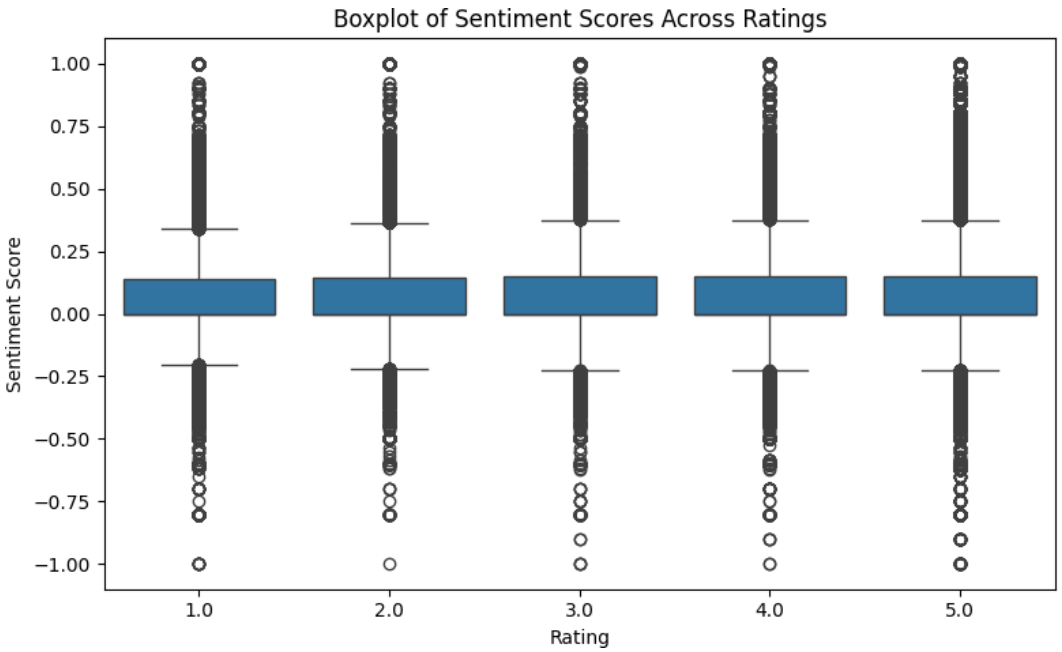
6.1 Sentiment Histogram



- The x-axis represent sentiment score and y-axis represent the frequency of reviews.
- We have larger spikes near zero (0) suggesting that most of the reviews are neutral and not using strongly emotional words.
- Some reviews are positively polarised and very less reviews are negatively polarised suggesting some emotional words are used in them.
- **Interpretation:** The curve above histogram suggests the same things that most reviews do not have strong emotional words in them (i.e, they are neutral) while some have emotional words

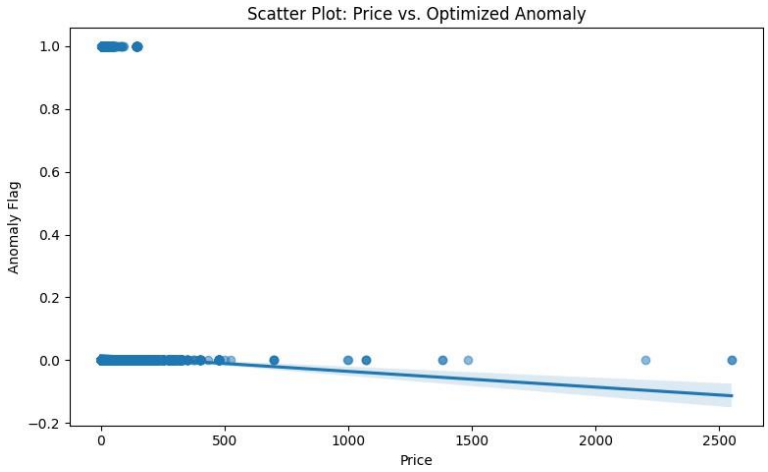
used in them but they are less in numbers.

6.2 Boxplot Sentiment Rating



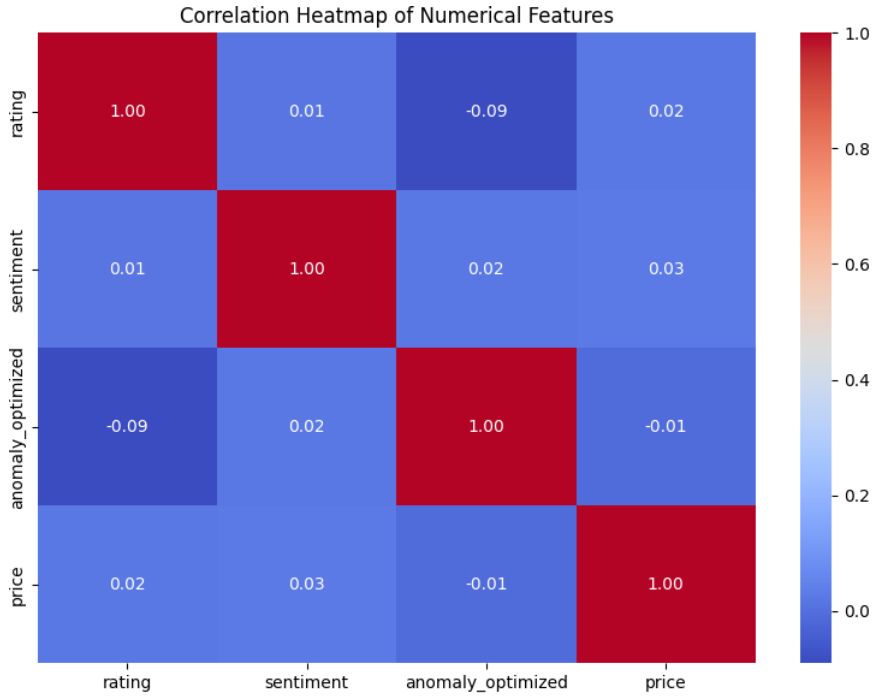
- The x-axis represent rating and y-axis represent sentiment scores.
- Each box represents the middle 50% (IQR) of sentiment scores for each rating, with the horizontal line marking the median. The extensions reach up to 1.5×IQR, while any points beyond them are identified as outliers.
- In this boxplot, each rating category (from 1 to 5) has a median sentiment close to zero—indicating overall neutrality—with similar interquartile ranges, while the presence of outliers shows that some reviews exhibit markedly strong positive or negative sentiment regardless of their rating.
- **Interpretation**
  - Even though higher ratings (like 4.0 or 5.0) might be expected to have more positive text, the boxplot shows a broad overlap of sentiment scores across all rating levels.
  - The concentration of medians near zero aligns with the idea that many reviews use relatively neutral language, even when giving higher or lower ratings.

6.3 Anomaly vs Price



- The x-axis represents product prices (ranging from 0 to 2500), while the y-axis represents a continuous anomaly measure, which can occasionally dip below 0 or exceed 1.
- Most data points are concentrated below a price of 500, with some anomaly values approaching 1. For products with high price reviews are more scattered toward the right side of the graph.
- The line represents a linear trend, while the shaded region indicates the confidence interval, which expands where data points are sparse.
- As product prices goes up, the anomaly measure declines, suggesting a negative correlation between cost and flagged anomalies.
- **Interpretation:** This graph tells that product with high price have genuine reviews and products with less price have more fake reviews.

6.4 Correlation Heatmap



- **Diagonal of 1.00:** Each feature have perfect correlation with itself which is obvious.
- **Rating vs. Sentiment (0.01):** Low correlation score indicates a neutral relationship between Ratings and text sentiment.
- **Rating vs. Anomaly (-0.09):** There is a low negative correlation between higher ratings and anomalies.
- **Sentiment vs. Anomaly (0.02):** Anomaly detection isn't strongly influenced by sentiment alone, showing minimal dependence, which was the main logic behind our idea.
- **Sentiment vs. Price (0.03):** The positivity or negativity of reviews has little to no connection with product cost.
- **Anomaly vs. Price (-0.01):** Higher-priced items exhibit slightly fewer anomalies, but the relationship is negligible.

7 Discussion

7.1 Limitations

- There may be other factors need to be taken in account for good results while detecting the anomalies as only rating and sentiment are not sufficient to detect anomaly.

- The chosen threshold (median $\pm$ $k \times$IQR) is sensitive to the value of k as it changes the range value, it may happen that anomalies are not detected properly or wrongly marked, it should be optimized well for good results.
- TextBlob's sentiment analysis may not capture nuanced language or context effectively.
- We did not have a labeled dataset i.e, wheather the review and rating are anomaly or not, having it may help us to cross validate.

7.2 Future Directions

- We can take other factors in account other than ratings and sentiment scores to get good results and increase accuracy.
- We can use advanced NLP techniques to do much better sentiment analysis for the given reviews.
- We can use other machine learning-based anomaly detection methods like Isolation Forest, clustering, etc.
- Develop a labeled dataset to evaluate and refine the anomaly detection accuracy. Having labeled dataset will help to optimize the threshold parameter k, by parallely cross validating it with the labeled dataset.
- Optimize the q1 and q3 values to get more robust results.

7.3 Challenges

- It was a bit tricky to get the review dataset and item-metadata from the provided link.
- Faced the problems as discussed in Limitations part, optimizing the k value and cross validation.
- Both the merged csv file and the processed csv file exceeded 1 GB each, so we had to use Git LFS for pushing files on github.

End of Report

Contributions

- **Diwakar Prajapati (220382):** Scraping the data from provided link, developed the python script, write the LaTex code for PDF, contributed in discussion section and did the analysis for Boxplot Sentiment Rating and Correlation Heatmap.
- **Shivam Kumar (221013):** Helped in writing the code with minor corrections, developed the main logic of script while referencing to **Tukey's Outlier Methods** and optimizing the parameters like k value with the help of internet (saw 2 research papers for good results), did analysis for sentiment histogram and price vs anomaly scatter plot, and helped in discussion section.

Please refer to the title page (first page of the PDF) for GitHub link to review Codes.