**Amazon Reviews Sentiment Analysis Project**

**Member 1 - Shivpriya Mane & Member 2 - Vaishnavi Bhavesh Patel**

Team 20

### **Introduction**

Sentiment analysis of Amazon book reviews is a specialized branch of Natural Language Processing (NLP) that aims to understand the emotional undertones conveyed through textual feedback on literature. As the volume of e-commerce continues to swell, book reviews on platforms like Amazon offer a critical source of information. These reviews not only influence potential readers' decisions but also provide authors and publishers with valuable insights into audience reactions. Analyzing these sentiments can uncover trends and preferences, guiding both future writing and marketing decisions.

### **Background**

The field of sentiment analysis in the context of book reviews requires handling unique challenges such as the diverse literary styles, nuanced language, and the deep emotional engagement often found in book reviews. From initial techniques like keyword spotting to more advanced machine learning methods including neural networks, sentiment analysis has evolved to capture complex emotional responses. Tools such as NLTK and other Python libraries are instrumental in categorizing reviews beyond simple positive or negative sentiments. This analysis not only aids in understanding reader satisfaction but also helps in refining publishing strategies and addressing broader reading trends. Furthermore, it raises ethical considerations about data privacy and the potential biases in automated systems, emphasizing the need for careful, respectful management of reader feedback.

### **Motivation**

Our project is motivated by the desire to delve into the rich emotional tapestry woven by reader reviews on Amazon. By applying sentiment analysis to these book reviews, we aim to uncover detailed insights into readers' emotional journeys through various narratives, which can empower authors and publishers to enhance their engagement with readers and adapt to their preferences and critiques.

### **Goal**

Our goal is to perform sentiment analysis on Amazon book reviews to ascertain the sentiments expressed by readers towards different literary works. We plan to classify these sentiments as positive, negative or neutral and provide aggregated sentiment scores for individual books. By comparing the performance of various machine learning models, based on classification accuracy and ROC scores, we aim to utilize these models to derive meaningful conclusions about the pros and cons of the books as perceived by the readers.

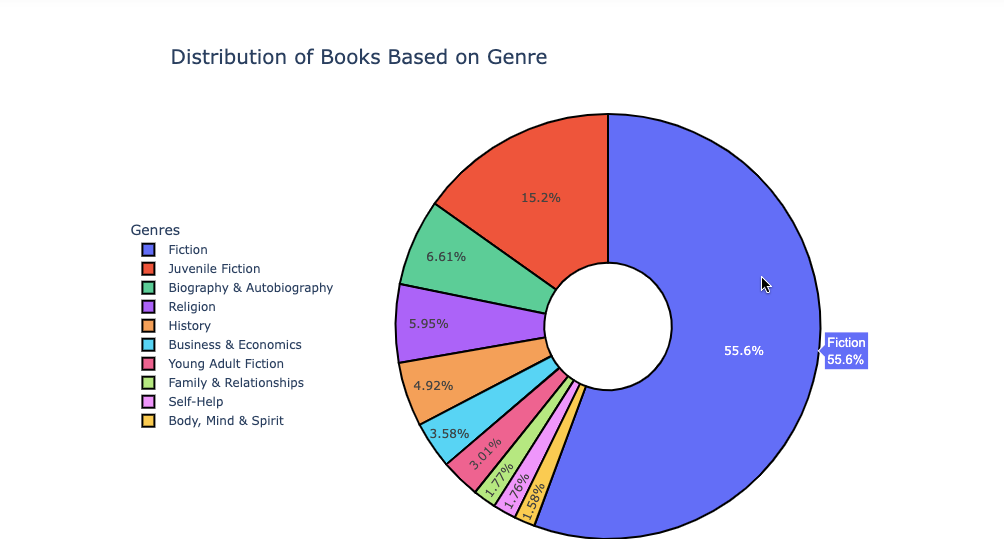
### **Methodology**

### **Data Preprocessing and Cleaning**

* **Data Loading**: Initially, we imported two distinct datasets using Pandas; one with ratings of books and the other with comprehensive book details.
* **Merging Datasets**: We merged these datasets based on the common attribute, 'Title'. This integration allowed us to connect each book's ratings with its corresponding metadata for a unified analysis framework.
* **Column Selection**: From the merged dataset, we carefully selected columns that were pertinent to our analysis goals. These included 'Title', 'review/score', 'review/summary', 'review/text', 'authors', 'categories', and 'ratingsCount'.
* **Duplicate Removal**: We removed duplicate entries to enhance the dataset’s integrity, ensuring each data point was unique and thus reliable for analysis.
* **Null Value Handling**: We also eliminated any rows containing null values across essential columns to avoid any bias or error in subsequent analyses.
* **Data Sampling**: To manage computational resources effectively, we sampled approximately 30% of the data. This subset was representative yet manageable for experimental analysis.
* **Text Cleaning**: Added text cleaning included extracting clean author names and categories by removing extraneous formatting and focusing on meaningful content. We also computed a 'word\_count' for each review to help detailed textual analysis.

**Exploratory Data Analysis**

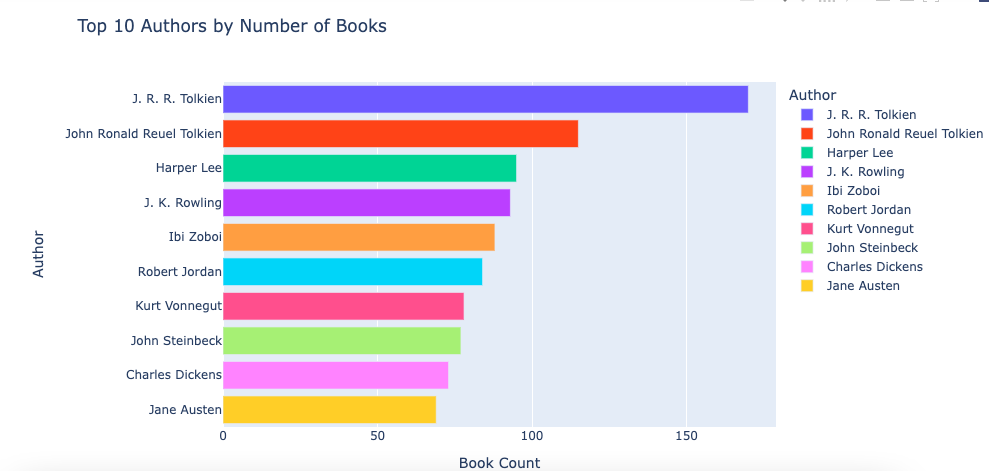
* **Sentiment Assignment**: We transformed 'review/score' into a binary 'Sentiment' category, designating scores of 0.05 or higher as 'Positive' and score less than –0.05 as 'Negative' and rest as 'Neutral'. This classification allowed us to focus on the polarity of feedback.
* **Visualizing Sentiment Distribution**: We created visual representations to explore the distribution of sentiments across various books, identifying trends and outliers in reader responses.
* **Thematic Analysis Using Word Clouds**: To visually capture prevalent themes and expressions in reviews, we generated word clouds for both positive and negative sentiments. This helped highlight the most frequent terms and topics mentioned by readers.
* **Review Length Insights**: We analyzed the length of reviews to investigate any potential correlation between the depth of review content and the sentiment expressed, aiding in understanding how textual length might influence reader sentiment.



The pie chart indicates that Fiction dominates the book reviews on Amazon, making up over half of the genre distribution, which suggests that fiction books are either more popular or more frequently reviewed than other genres. Juvenile Fiction is the next most reviewed genre, while other genres like Religion, History, and Self-Help constitute smaller fractions, indicating more niche audiences or fewer reviews

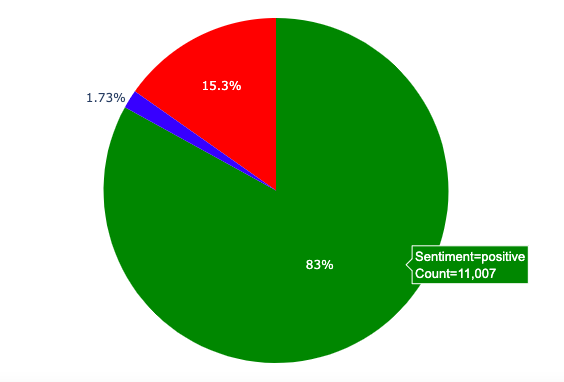


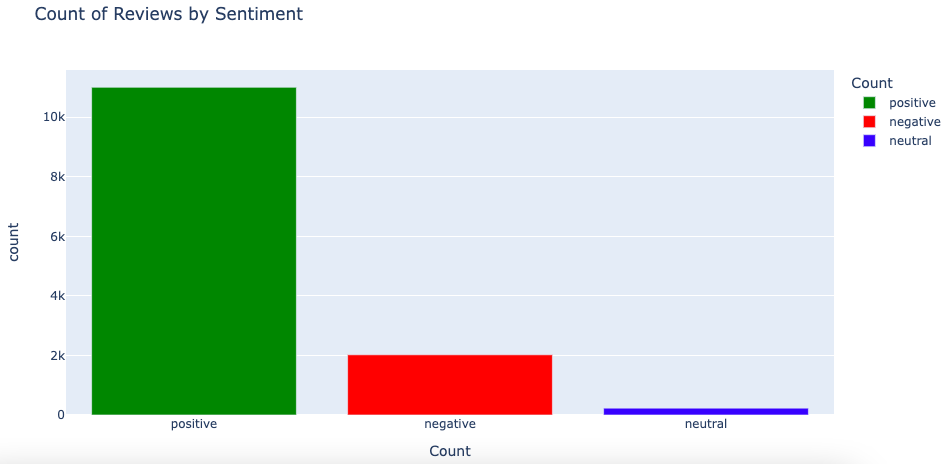
The bar chart titled "Most Frequent Words in Reviews with Score > 3" shows that the most common words in positive book reviews are mostly common English stop words such as "the," "and," "of," "to," indicating that the analysis might benefit from stop-word removal for more insightful results. Despite this, the presence of words like "book" suggests that positive reviews often directly reference the item being reviewed.



The bar chart "Top 10 Authors by Number of Books" reveals that J.R.R. Tolkien, John Ronald Reuel Tolkien, and Harper Lee are among the most prolific authors in this dataset. Notably, J.R.R. Tolkien appears twice, potentially indicating a duplication issue that may require data cleaning. The list includes classic authors like Jane Austen and Charles Dickens, reflecting their enduring popularity.

Sentiment Analysis





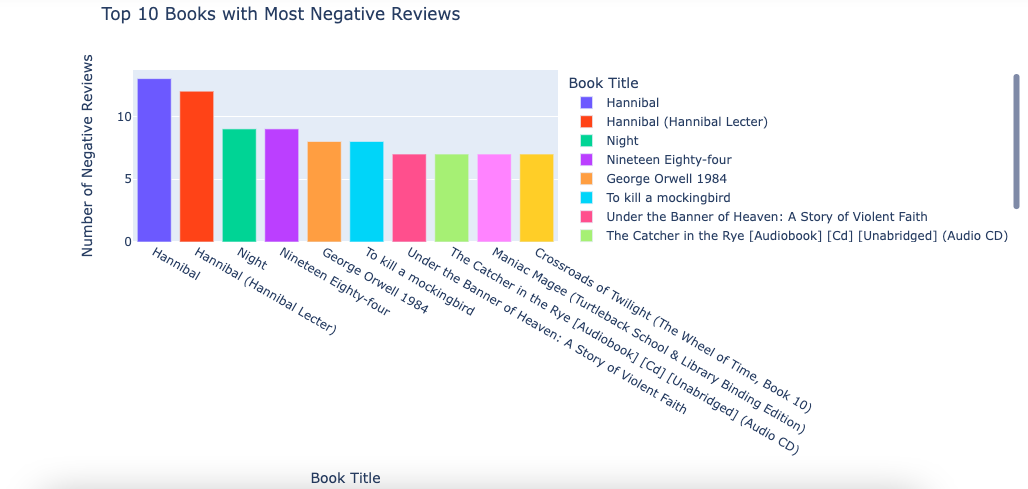
We illustrated the sentiment score distribution using a pie chart and a bar chart. The chart showed the massive imbalance between positive, negative and neutral reviews.

The graph is a bar chart titled "Count of Reviews by Sentiment," showing bars corresponding to sentiment score categories labeled ‘positive’, ‘negative’ and ‘neutral’ on the x-axis. The y-axis represents the count of reviews.

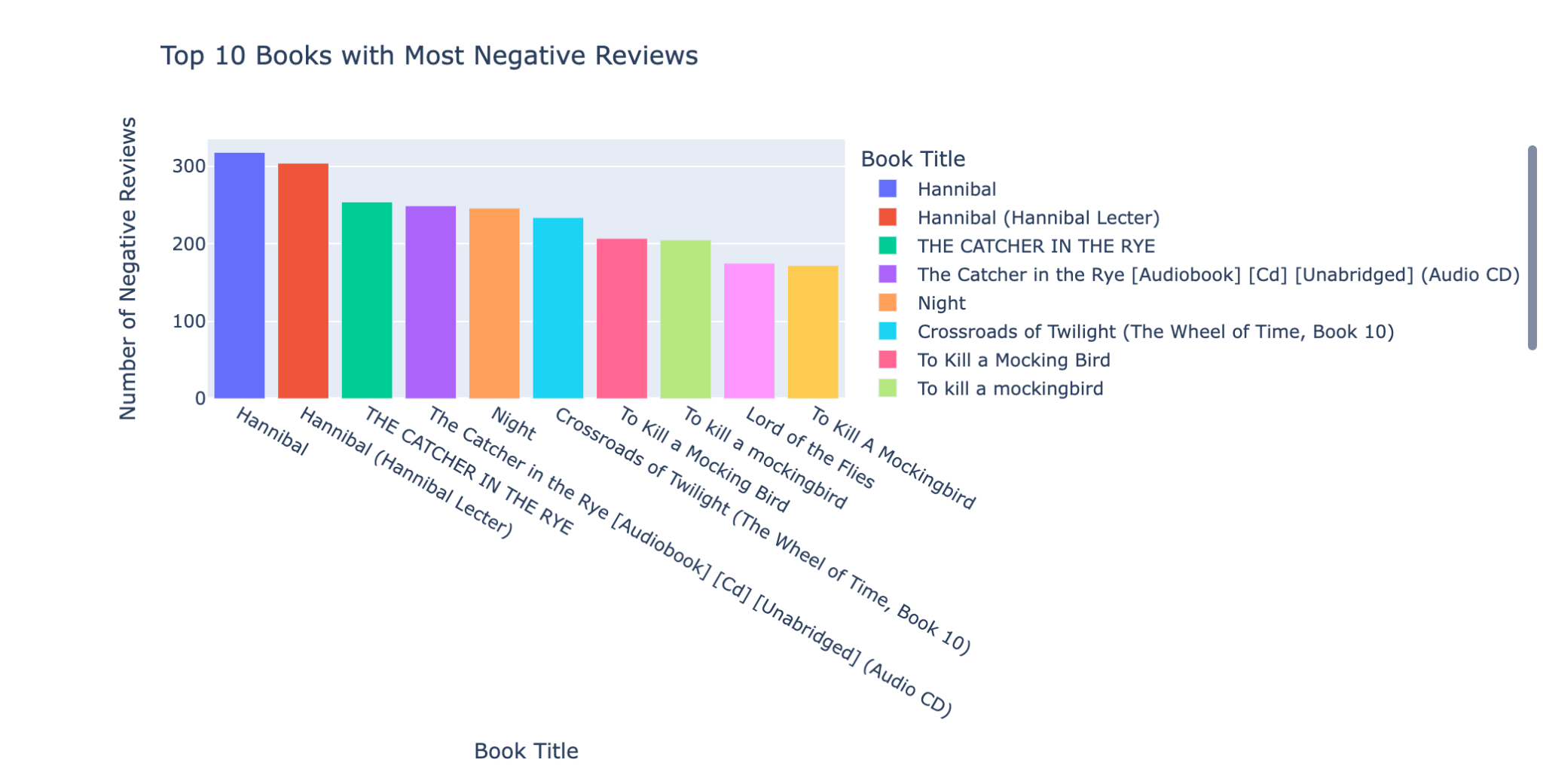
Top 10 Positive Reviews -



Top 10 Negative Reviews -



Top 10 Neutral Reviews -



### **Creating and Interpreting the Correlation Matrix**

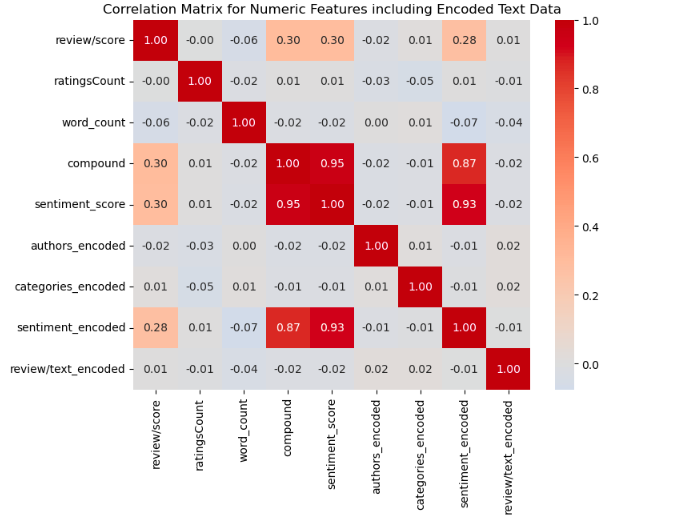
To further understand the relationships between numerical features in our dataset, we constructed a correlation matrix. This matrix helped us explore how different numerical attributes, such as 'review/score', 'word\_count', and 'ratingsCount', relate to each other, particularly how they might influence the 'Sentiment' derived from book reviews.

#### Process:

* **Selection of Numerical Features**: We first identified all numerical columns in our dataset that could potentially influence or reflect the sentiments expressed in the reviews.
* **Computation of Correlation Coefficients**: Using the Pearson correlation method, we calculated the correlation coefficients between each pair of selected features. This method quantifies the degree to which a change in one feature could linearly predict a change in another.
* **Visualization with a Heatmap**: To make the correlation matrix easier to interpret, we visualized the matrix as a heatmap using libraries such as Matplotlib and Seaborn. This visualization highlighted stronger correlations in distinct colors, making it straightforward to identify significant relationships at a glance.

#### Interpretation:

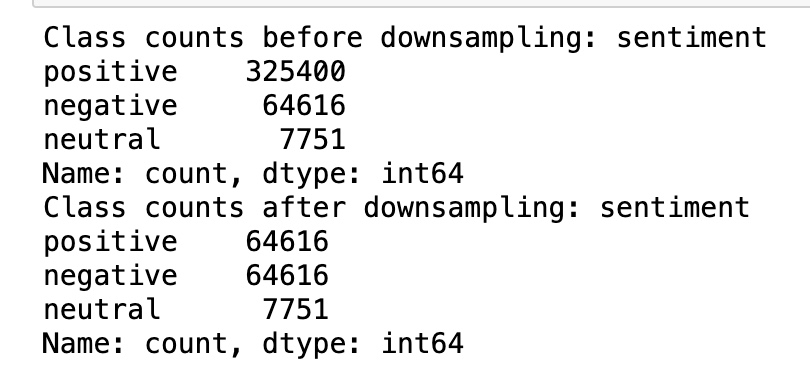
* **Positive Correlations**: High positive values in the matrix indicate features that move in tandem, suggesting that as one feature increases, the other does as well. For instance, if 'word\_count' and 'review/score' show a strong positive correlation, it suggests that longer reviews tend to have higher scores.
* **Negative Correlations**: High negative values suggest an inverse relationship. For example, if 'ratingsCount' and 'Sentiment' are negatively correlated, it could imply that books with more ratings do not necessarily receive more positive reviews, potentially due to higher critical scrutiny.
* **No Correlation**: Values near zero imply that there is no linear relationship between those features. This can be crucial for deciding which variables may not be useful in predictive modeling.



**Results and Analysis**

**Feature Engineering**

* We are using TF-IDF vectorization for Logistic Regression, and Random Forest and XGBoost because it effectively captures the importance of words in relation to the entire dataset. TF-IDF, which stands for Term Frequency-Inverse Document Frequency, helps in diminishing the weight of terms that occur very frequently in the dataset and increases the weight of terms that occur rarely. This method is particularly beneficial for these algorithms as it provides a numerical representation of text data, enabling them to process and classify the data more accurately. Additionally, TF-IDF vectorization helps in handling high-dimensional data, which is common in text processing, making it ideal for these machine learning models.
* One takeaway was, we needed to down sample our majority class (Positive sentiments) which was done using the resample function from sklearn.utils. This balanced dataset will be crucial for training our machine learning models more effectively. By ensuring that the number of positive and negative sentiment examples is equal, we prevent the model from being biased towards the majority class.
* This balancing act helps in improving the overall accuracy and generalizability of the model, making it more adept at correctly classifying both positive and negative sentiments in new, unseen data. In turn, this enhances the model's utility in real-world applications where sentiment distribution might not always be evenly split.

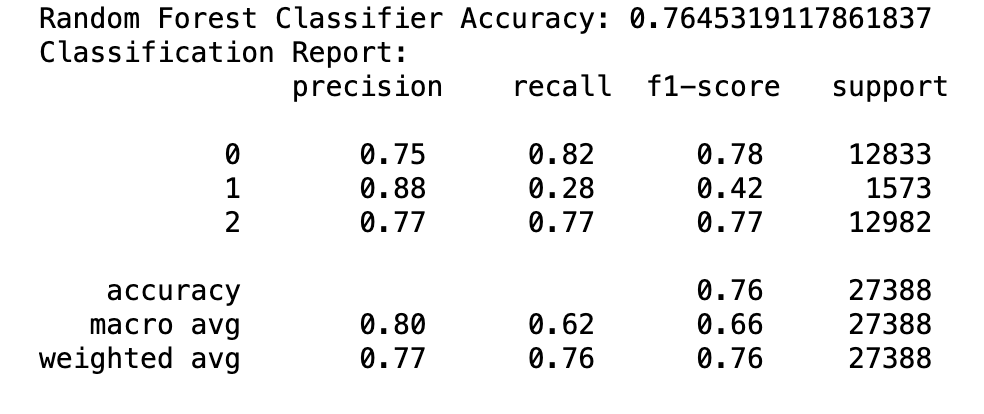


**Model Selection**

We explored various Machine Learning models suitable for sentiment analysis, such as Logistic Regression, Random Forrest and XGBoost. The models were evaluated based on their accuracy, precision, recall, ROC-AOC, confusion matrices and F1-score.

**Random Forrest**

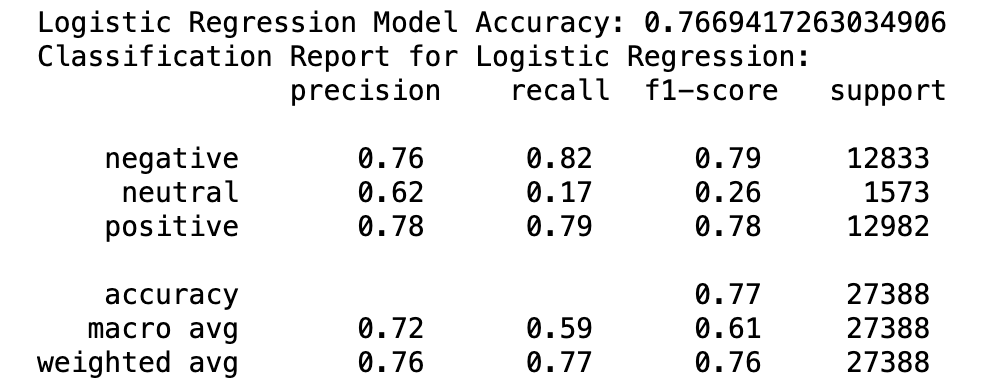
* **Accuracy:** Random Forests often produce a highly accurate model by combining the output of multiple decision trees to reduce overfitting.
* **Robustness:** They are less prone to overfitting than individual decision trees and can handle outliers and non-linear data effectively.
* **Versatility:** Capable of performing both regression and classification tasks, and they work well with both categorical and continuous data.
* **Feature Importance:** They provide insights into the importance of each feature in prediction, helping in feature selection and understanding the data.
* **Handling Missing Values:** Random Forest can handle missing values in the data, either through imputation or by using the intrinsic properties of the trees.
* **Ease of Use:** They require very little tuning of hyperparameters and can be used relatively easily without the need for scaling data.



**Logistic Regression**

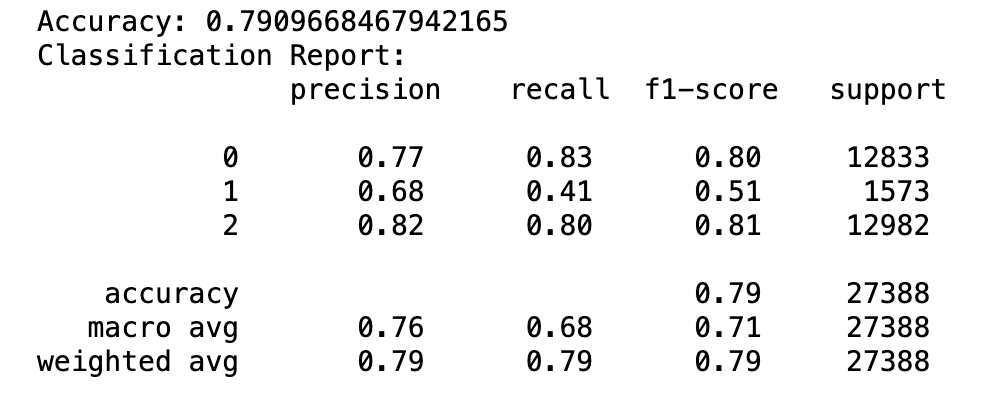
* **Efficiency**: Fast and computationally efficient for large datasets.
* **Clarity**: Offers clear interpretability of feature influence on sentiment.
* **Probabilistic Output:** Provides likelihood scores for sentiment categories.
* **Flexibility:** Capable of binary and multinomial classification tasks.

**Baseline Comparison:** Serves as a reliable baseline for evaluating other models.



**XGBoost**

* **High Performance**: XGBoost is known for delivering excellent accuracy and efficiency on large datasets.
* **Handling Sparse Data:** It efficiently manages sparse data, which is common in text processing.
* **Scalability:** XGBoost scales well with large volumes of data, making it suitable for extensive review datasets.
* **Regularization Features:** It includes built-in L1 and L2 regularization to help prevent overfitting.
* **Model Interpretability:** Provides feature important scores, aiding in understanding which words influence sentiment.
* **Flexibility:** Supports custom optimization objectives and criteria, adaptable to specific sentiment analysis needs.
* **Robustness:** Performs well across various data distributions, reducing the need for data preprocessing.



**Conclusion**

Comprehensive Data Handling and Analysis:

The project adeptly processed and analyzed Amazon book reviews, starting from data loading to merging, cleaning, and sampling for optimal analysis efficiency. Extensive data preprocessing ensured the integrity and usefulness of the dataset for sentiment analysis.

Robust Methodology and Explorative Data Analysis:

Sentiment categorization was effectively achieved by transforming review scores into binary and neutral categories, complemented by thematic insights from word clouds and sentiment distribution visualizations. This explorative approach provided a foundational understanding of the underlying sentiments in the reviews.

Effective Use of Machine Learning Models:

- Random Forest showcased high accuracy, robustness to overfitting, and valuable feature insights, proving its efficacy in handling complex, non-linear sentiment analysis tasks.

- Logistic Regression was utilized for its computational efficiency and clarity in interpreting features, serving as a reliable baseline for model comparison.

- XGBoost excelled in performance, especially in handling sparse text data, scalability, and offering deep interpretative insights into feature importance due to its advanced regularization and optimization features.

The sentiment analysis revealed a significant imbalance among positive, negative, and neutral sentiments, emphasizing the critical nature of nuanced sentiment detection in literary reviews. The models employed were able to distinguish between various sentiments effectively, providing a granular view of consumer opinions.

Strategic Implications and Model Effectiveness:

The analysis not only aids publishers and authors in understanding reader sentiments but also highlights the importance of nuanced language processing in reviews. The project demonstrated the effectiveness of advanced machine learning techniques, particularly XGBoost, in deriving meaningful insights from textual data.

The project effectively harnessed the power of NLP and machine learning to offer actionable insights into Amazon book reviews, with XGBoost standing out for its robust handling of the challenges posed by high-dimensional, sparse text data. The findings can significantly influence future publishing strategies and enhance reader engagement by tailoring offerings to meet the revealed reader preferences and sentiments.

This project underscores the transformative potential of machine learning in interpreting complex human emotions and preferences through textual analysis, setting a benchmark for future sentiment analysis projects in the literary domain.

**References**

[1] Kaggle Amazon Book Review: [Link](https://www.kaggle.com/code/shubham2703/amazon-books-review-eda-sentiment-analysis)

[2] Class notes, Jupyter notebook from Professor Handan Liu

[3] Medium Article: **Sentiment Analysis on Amazon Reviews using TF-IDF Approach** [**Link**](https://medium.com/analytics-vidhya/sentiment-analysis-on-amazon-reviews-using-tf-idf-approach-c5ab4c36e7a1)