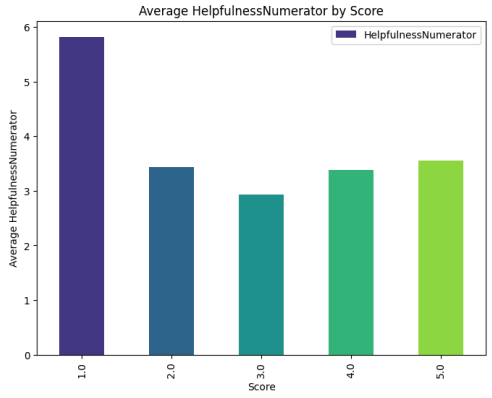
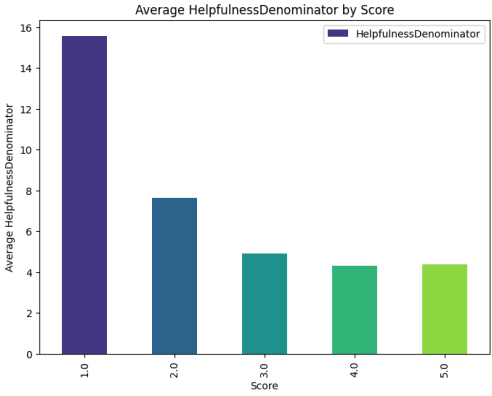
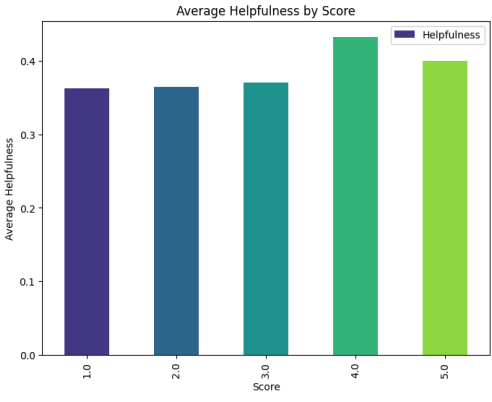
**Caroline Muscara**
**U96730221**
**CS506 Midterm Writeup**

For feature extraction, I chose to add review length and sentiment features because they offer valuable insights into the nature of the review content, which can be indicative of user ratings. The length of a review(measured by the word count) can provide context about how detailed or comprehensive the review is, as users who leave longer reviews might be more invested in sharing their experiences, whether positive or negative. While review length alone showed a weaker correlation with score, it was still expected to complement other features and provide additional predictive value. On the other hand, sentiment was included as it directly measures the emotional tone of the review text, ranging from negative to positive. Since higher review scores are generally associated with more positive sentiment, incorporating this feature helped capture a critical aspect of the data that is closely linked to the target variable. Both features were thus selected for their potential to enhance the model's ability to differentiate between varying review scores, making the predictions more accurate and contextually informed.

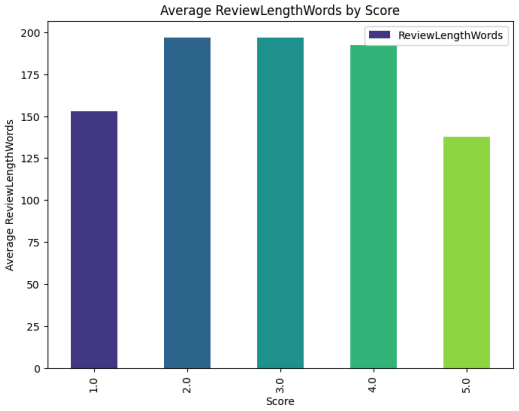


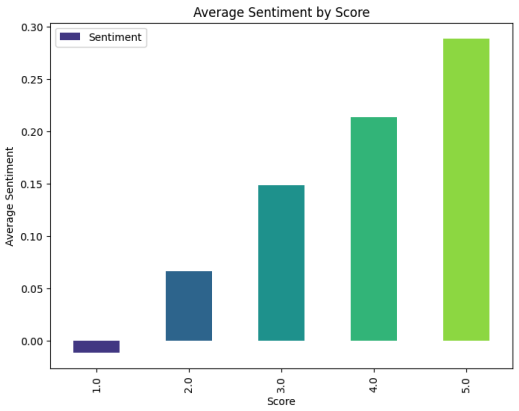




In addition to feature extraction, I implemented a function to analyze the relationship between individual features and review scores. This function grouped data by score, computed the average value of each feature for each score, and plotted the results. It also calculated the correlation between each feature and the score, providing insights into which features might serve as effective predictors. The analysis revealed varying degrees of association among features. Sentiment, with a correlation of 0.422, emerged as a promising predictor,



**Caroline Muscara**
**U96730221**
**CS506 Midterm Writeup**




indicating that more positive sentiment in reviews generally corresponds with higher scores. In contrast, helpfulness-related features, such as HelpfulnessNumerator and HelpfulnessDenominator, demonstrated weaker correlations, signaling limited predictive value. Similarly, ReviewLengthWords showed a slight negative correlation (-0.079), suggesting that longer reviews do not necessarily translate to higher scores. This analysis guided the feature selection process, confirming sentiment as a valuable predictor while highlighting the limited utility of other features when used independently.

The feature selection process focused on numeric features that had some predictive potential, including Helpfulness, ReviewLengthWords, Sentiment, and the target variable, Score. Non-numeric features were excluded to ensure compatibility with the model, while missing values were set to zero to prevent errors during training and maintain the integrity of the dataset. To address class imbalance, SMOTE (Synthetic Minority Over-sampling Technique) was applied to the training set. SMOTE generated synthetic samples for minority classes, effectively balancing the class distribution and preventing bias toward the majority class. This approach improved model performance by ensuring adequate representation of all classes during training. To manage the size of the dataset and expedite testing, I implemented a data reduction strategy. Both the balanced training dataset and the original training set were down-sampled by 50%. This approach maintained the balanced class distribution achieved by

**Caroline Muscara**
**U96730221**
**CS506 Midterm Writeup**

SMOTE while significantly reducing the dataset's size, making it more manageable for faster iteration and testing.

While I explored several alternative approaches to improve performance, time constraints limited their complete implementation. I initially attempted to optimize the KNN model using GridSearchCV for hyperparameter tuning. However, the identified n_neighbors value resulted in lower accuracy on the testing set, leading me to revert to a fixed n_neighbors=175, which had performed better during preliminary testing. Additionally, I experimented with scaling the features using StandardScaler to enhance model convergence and accuracy, but this approach increased runtime and introduced indexing issues. Dimensionality reduction using PCA was also considered, but it faced similar runtime challenges and column indexing complications, preventing full integration within the given time frame.

Despite these challenges, the final approach relied on a combination of feature engineering, data balancing, and careful selection of model parameters to improve performance. Correlation analysis was pivotal in identifying sentiment as a key feature, while SMOTE addressed class imbalance effectively. Down-sampling and selective feature inclusion improved training efficiency without compromising accuracy. The methodology involved iterative testing and adjustments, allowing the model to achieve optimal results within the constraints. Future improvements could include more comprehensive hyperparameter tuning, refined feature scaling, and dimensionality reduction to enhance both accuracy and runtime.