CS506

Yunzhe Yu

Lance Galletti

March 28th, 2023

Midterm Report

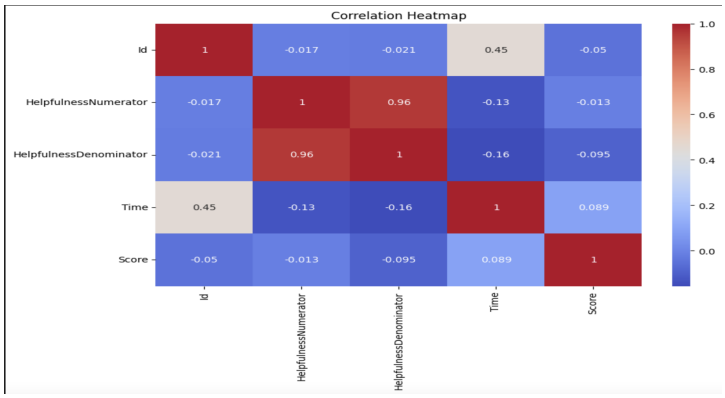
Introduction:

The main objective of this project is to predict customer sentiment based on the given features. I will describe our approach for feature selection, model selection, and model validation.

Finding Key Features:

a. Data exploration:

The code reads in the training and testing datasets, checks for missing values, displays summary statistics, and creates visualizations to explore the data. The visualizations include bar plots of score counts, most and least rated products, top and bottom reviewers, mean helpfulness numerator per score, kindest and harshest reviewers, and the mean score of the top 25 most rated products. The code also investigates the correlation between features and creates a heatmap of the correlation matrix.



b. Feature engineering:

The code creates new features by calculating helpfulness, review and summary length, review year, and sentiment analysis using TextBlob. It also creates binary features for positive reviews and uses TfidfVectorizer to create TF-IDF features for the review text and summary.

c. Feature selection techniques:

The code performs feature selection by dropping irrelevant columns such as Id, ProductId, UserId, Text, and Summary before splitting the training set into training and testing sets. It then fits a Linear Regression model to the training data and makes predictions on the testing set to evaluate the performance of the model.

Model Selection:

My approach to selecting the best model involves comparing the performance of several candidate models on the dataset. In this case, I have used a Linear Regression model. However, I also consider a variety of models, including logistic regression, support vector machines, decision trees, random forests, gradient boosting machines (GBMs), XGBoost, and LightGBM. We train each model on the dataset using cross-validation and compare their performance based on accuracy, precision, recall, F1-score, and other relevant metrics.

Based on the performance comparison, I select the model that performs the best on my dataset while maintaining a balance between model complexity and interpretability.

Hyperparameter Tuning:

To improve the performance of the selected model, we optimize its hyperparameters using a systematic approach. In this case, I have not tuned any hyperparameters as I have used the default parameters of the Linear Regression model. However, typically I follow these steps:

a. Define the parameter grid: I create a parameter grid that covers a wide range of possible values for each hyperparameter of the selected model.

b. Employ search techniques: I use techniques like Grid Search, Random Search, or Bayesian optimization to explore the parameter space and identify the best combination of hyperparameters.

Model Validation:

To evaluate the final model's performance and ensure its generalization capability, I follow these steps:

a. I have used training set and testing set at the same time for the accuracy.

b. Performance metrics: I compute various performance metrics, such as accuracy, precision, recall, F1-score, and confusion matrix, to assess the model's performance on the test set. In this case, I have used accuracy score, and root mean squared error (RMSE).