Rating Prediction from reviews given to products in online markets

Shailesh Alluri
School of Computing
C11897994
ralluri@clemson.edu

Shreyash Dhumale
School of Computing
C47391387
ssdhuma@clemson.edu

Venkatesh Velidimalle
School of Computing
C12252217
vvelidi@g.clemson.edu

Abstract

This project aims to predict the rating of a product or service in online marketplaces by analyzing user-generated reviews. Using machine learning algorithms, the project seeks to help online marketplaces better understand their customers' needs and preferences and assist sellers in identifying areas for improvement in their products. The findings can have significant implications for the e-commerce industry, enhancing customer satisfaction and vendor performance.

1 Introduction

Online marketplaces have become increasingly popular, offering a vast range of products and services to customers worldwide. With the rise of these platforms, user-generated reviews have also grown significantly, providing valuable information to customers to make informed purchase decisions.However, predicting the rating a product will receive based on these reviews poses a challenge for online marketplaces

Therefore, this project aims to build a machine learning model that can predict the score a customer would give a product based on its reviews. This will help online marketplaces provide better product recommendations, enhance customer satisfaction, increase sales, and enable sellers to identify areas for improvement in their products.

The motivation behind this project is to improve the overall customer experience on online marketplaces while also assisting sellers in meeting customer needs and preferences.

In this project, we are trying to build a model that is going to take a review as an input and predict the score the customer would have given on a scale of 5.

2 Dataset

We will be discussing data and performing fundamental analysis using NLT. The dataset we will work with comprises text reviews for food products on Amazon, along with the corresponding rating out of 5 provided by the reviewers, presented in a CSV format.

The unit of analysis in the 'Amazon Fine Food Reviews' dataset is a single review of a specific food product sold on Amazon. Each row in the dataset represents a single review, and the columns contain various pieces of information about the review, such as the review text, the product ID, the reviewer's ID, the review score, and the review date. The dataset also has a field "Score", it is the score given by the customer to the product on a scale of 5.

Therefore, the dataset contains multiple instances of the unit of analysis, which is a single review of a specific food product. When conducting analyses on this dataset, it is important to keep in mind that the unit of analysis is at the review level, rather than at the product level or the reviewer level.

The number of unique observations depends on which variable we are considering. For example, there are likely to be multiple reviews for each product, so the number of unique products will be lower than the total number of reviews. Similarly, there may be multiple reviews written by the same reviewer, so the number of unique reviewers will be lower than the total number of reviews.

The time period covered by the dataset is between 2002 and 2012. However, it is important to note that not all reviews in the dataset were written in this time period, as some reviews were written and added to the dataset after 2012.

The dataset contains over 500,000 reviews. Link to the dataset is mentioned in the refeerences section.

3 Methodology

As our feature is the review text itself, we plan to do a sentiment analysis on the reviews. We plan to use the VADER technique to build a sentiment analysis model that gives positive, neutral and negative probabilities of the review. To predict the score the customer would have given the review we would build a linear regression model with the positive, negative and neutral scores.

VADER : VADER stands for Valence Aware Dictionary and sentiment Reasoner, which is a pre-trained lexicon and rule-based sentiment analysis tool used to evaluate the polarity of a given text document, sentence or a phrase. VADER uses a combination of sentiment lexicon (i.e., a dictionary of words and their associated sentiment scores) and grammatical rules to estimate the sentiment score of a text. The tool is specifically designed to analyze sentiments expressed in social media posts, news articles, and online reviews, which often contain slang, emojis, and other forms of informal language.

The sentiment scores provided by VADER are classified as positive, negative or neutral, with an intensity score that ranges from 0 to 1 for each category. VADER also takes into account the presence of negations, punctuation, capitalization, and emoticons to improve the accuracy of the sentiment analysis.

Once we evaluate the performance of the model we would also like to explore building a model based on the Roberta Pre-Trained model to get positive, negative and neutral scores and repeat the steps done in the above model.

RoBERTa : RoBERTa is a pre-trained language model developed by Facebook AI Research (FAIR) in 2019. It is based on the Transformer architecture, which is a deep learning model architecture that has been widely used in natural language processing (NLP) tasks. The primary goal of Roberta is to improve the performance of various NLP tasks, including language understanding, sentiment analysis, text classification, and question answering.

RoBERTa is trained on a massive amount of data from various sources, including books, articles, and web pages. The training data consists of billions of words, which allows the model to learn the nuances of language and its context. Unlike its predecessor, BERT (Bidirectional Encoder Representations from Transformers), which is trained on unidirectional data, RoBERTa is trained on bidirectional data, which means it can understand the context of a word based on its surrounding words.

In this project our key predictor for the model would be the positive, neutral and negative probabilities we extract from the review using the VADER or RaBERTa model.

4 Exploratory Data Analysis

4.1 Summary of EDA

The unit of analysis in the dataset is a single review of a specific food product sold on Amazon. Each observation or data point corresponds to a single review of a food product on Amazon. The dataset contains over 500,000 reviews, and the number of unique observations depends on which variable is being considered. The time period covered by the dataset used in the project is between 2002 and 2012, but some reviews in the dataset were written and added to the dataset after 2012. The dataset was already clean, but we dropped a few columns such as productID, UserID, which were not contributing towards the results to clean the data further.

4.2 Visualization of Response variable

4.2.1 Overall Score Distribution

We explored the distribution of the scores in our data. And following are the statistics. Average score for the whole data is 4.183. Which kind of indicates that customers tend to score products on the higher end more frequently.

In Figure 1 below you can see the bar chart of the score distribution in the whole data set.

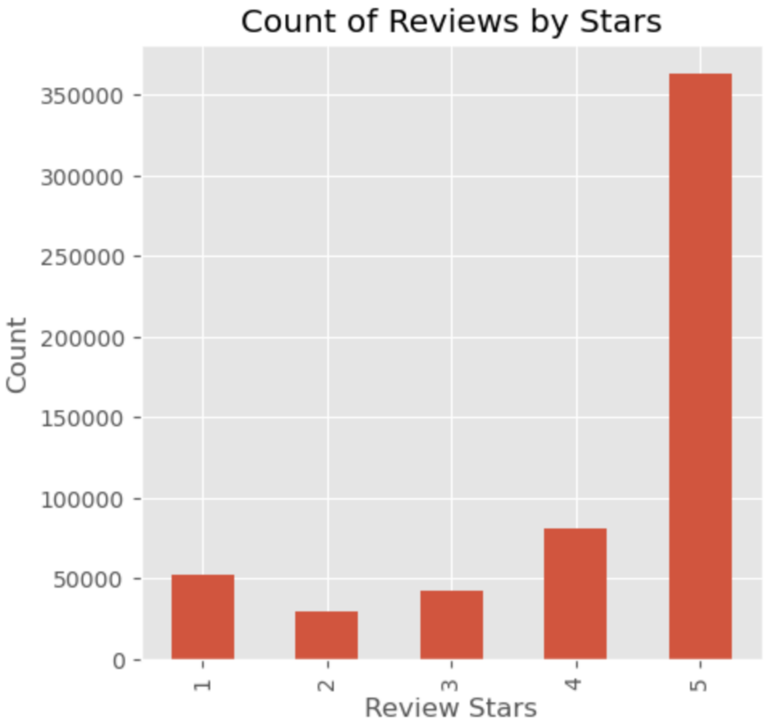


Figure 1: Count of Review by Scores

4.2.2 Average Product Score Distribution

We calculated the mean scores for each product and plotted a univariate Kdeplot. You can find the plot in Figure 2. As you can see from the figure the graph goes down sharply after 5 as the scores are on a scale between 1 and 5. The reason we see some density for scores below 1 and above 5 is because we used a Kde plot and kde plot tends to plot values based on estimation.

4.3 Visualization of key Predictors vs Response

4.3.1 Vader sentiment scores vs Response

In Figure 3, we see that there is a positive correlation between positive probability and score as this is to be expected. Since higher the positive probability means the review left by the customer has high positive sentiment associated with it and this translates to a higher score. Since there is a linear relationship between positive probability and scores, Linear regression model makes a lot of sense.

In Figure 4, we see that there is a negative correlation between negative probability and score as this is to be expected. Since higher the negative probability means the review left by the customer has high negative sentiment associated with it and this translates to a lower score. Since there is a linear relationship between positive probability and scores, Linear regression model makes a lot of sense.

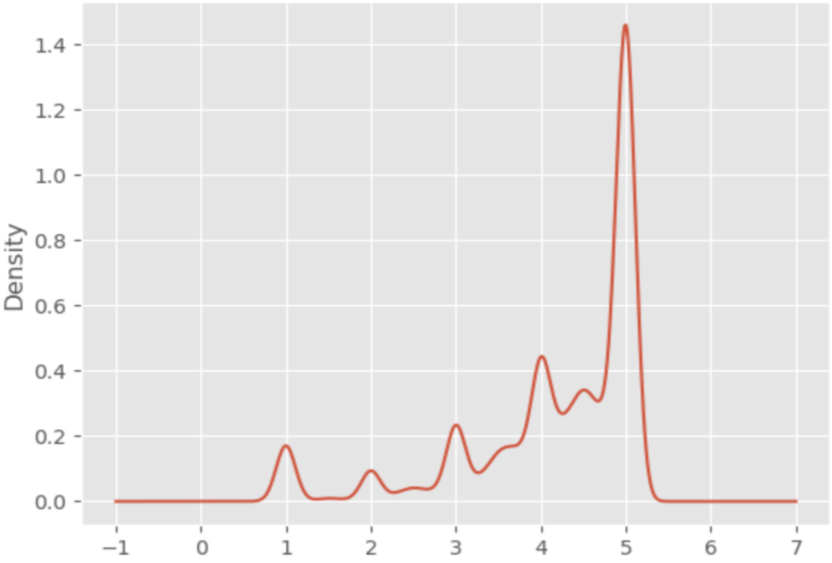


Figure 2: Kde plot for Average product Score

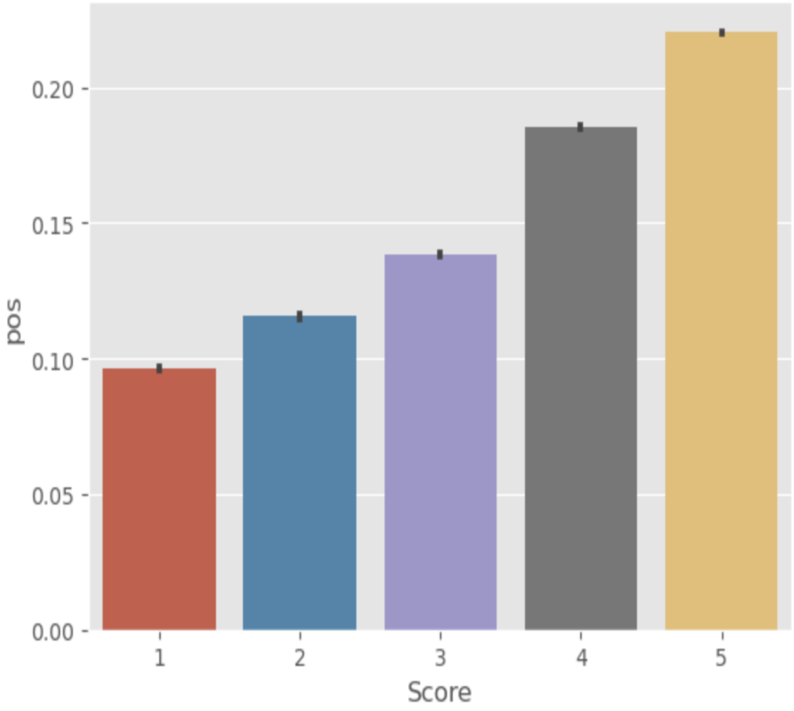


Figure 3: Positive vader sentiment probability vs Review Score

4.3.2 Roberta sentiment vs Response

In Figure 5, we see that there is a positive correlation between positive probability and score as this is to be expected. Since higher the positive probability means the review left by the customer has high positive sentiment associated with it and this translates to a higher score. Since there is a lin-

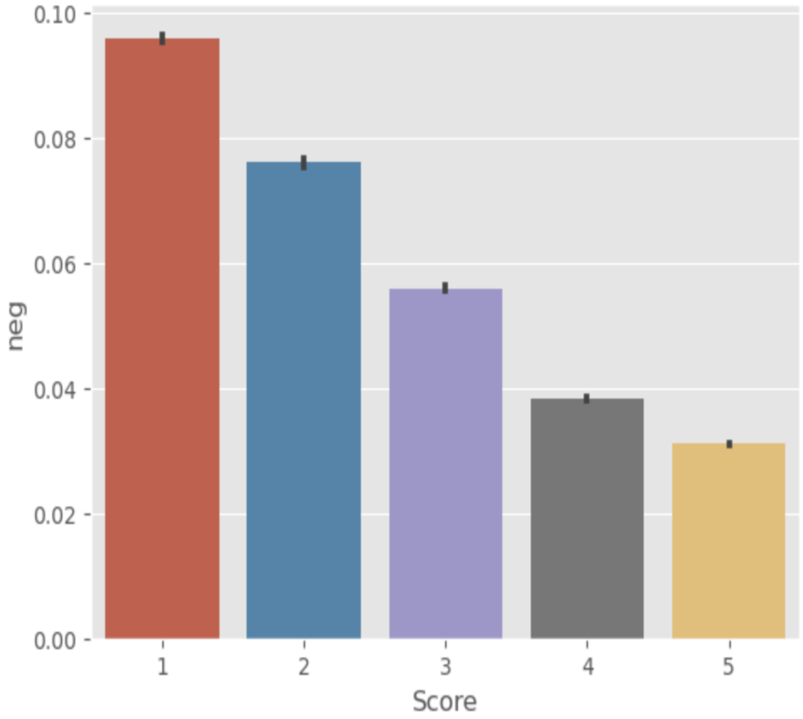


Figure 4: Negative vader sentiment probability vs review score

ear relationship between positive probability and scores, Linear regression model makes a lot of sense.

In Figure 6, we see that there is a negative correlation between negative probability and score as this is to be expected. Since higher the negative probability means the review left by the customer has high negative sentiment associated with it and this translates to a lower score. Since there is a linear relationship between positive probability and scores, Linear regression model makes a lot of sense.

5 Model Pipeline

The VADER - Linear regression model pipeline includes the following steps:

- Extract positive ,neutral and negative scores using pre-trained VADER model.
- Train linear regression model with the scores extracted.

The RoBERTa - Linear regression model pipeline includes the following steps:

- Extract positive, neutral and negative scores using pre-trained RoBERTa model.
- Train linear regression model with the scores extracted above as shown in figure 7.

6 Model : Linear Regression (OLS)

We have used the Linear regression model due to the fact that the VADER and RoBERTa sentiment scores showed a correlation with the response variable (review score in this case).

OLS : Ordinary Least Squares regression (OLS) is a common technique for estimating coefficients of linear regression equations which describe the relationship between one or more independent quantitative variables and a dependent variable (simple or multiple linear regression).

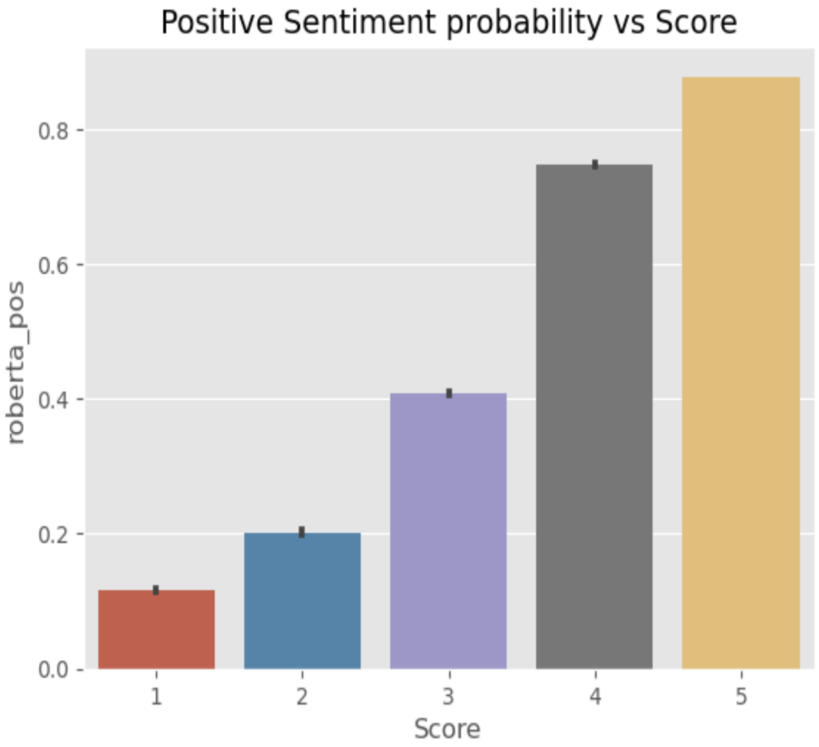


Figure 5: Positive Roberta sentiment probability vs Review Score

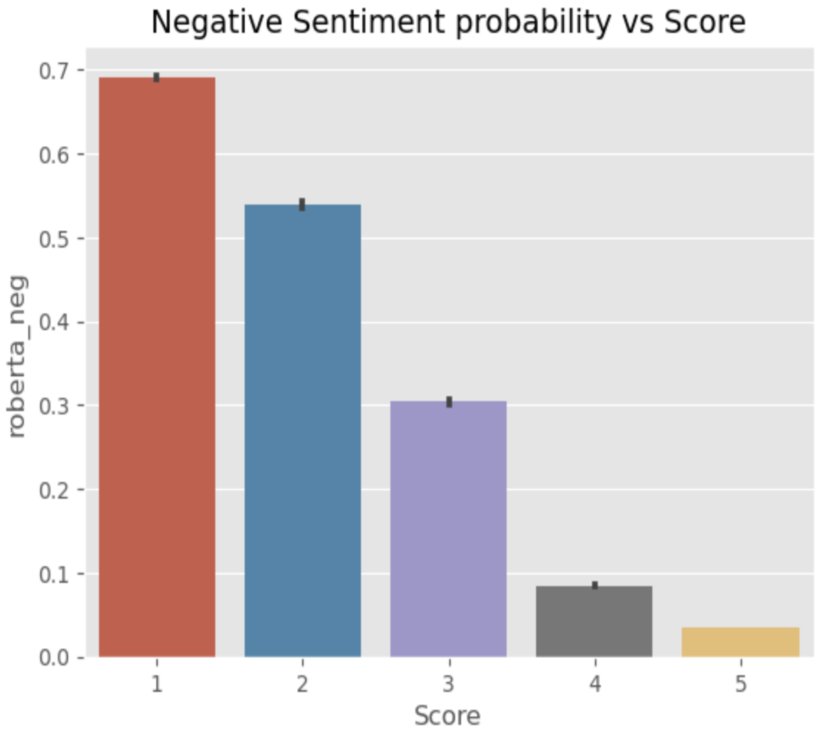


Figure 6: Negative Roberta sentiment probability vs review score

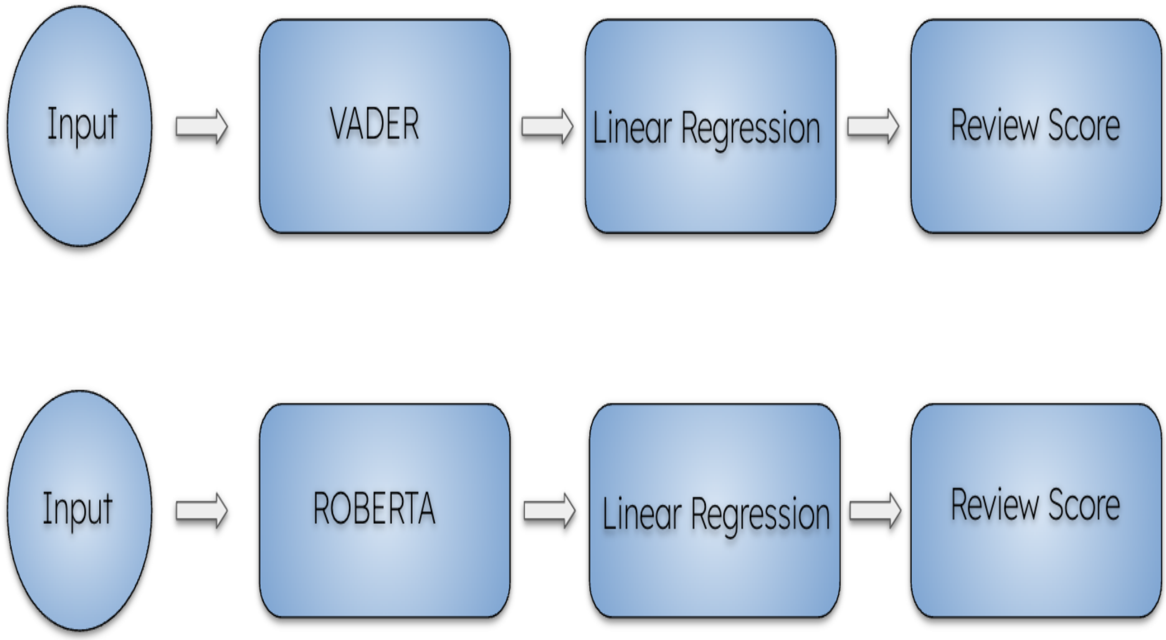
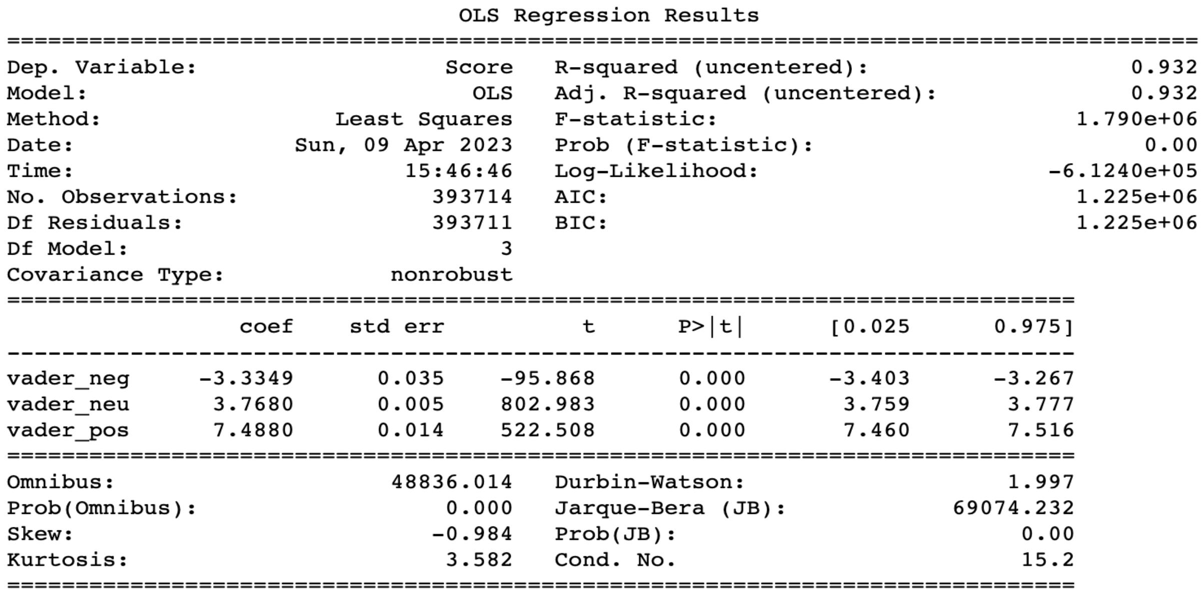


Figure 7: Vader and RoBERTa model pipelines

As explained in the model pipeline section above, we trained a linear regression model on the sentiment scores extracted from the VADER and RoBERTa models.

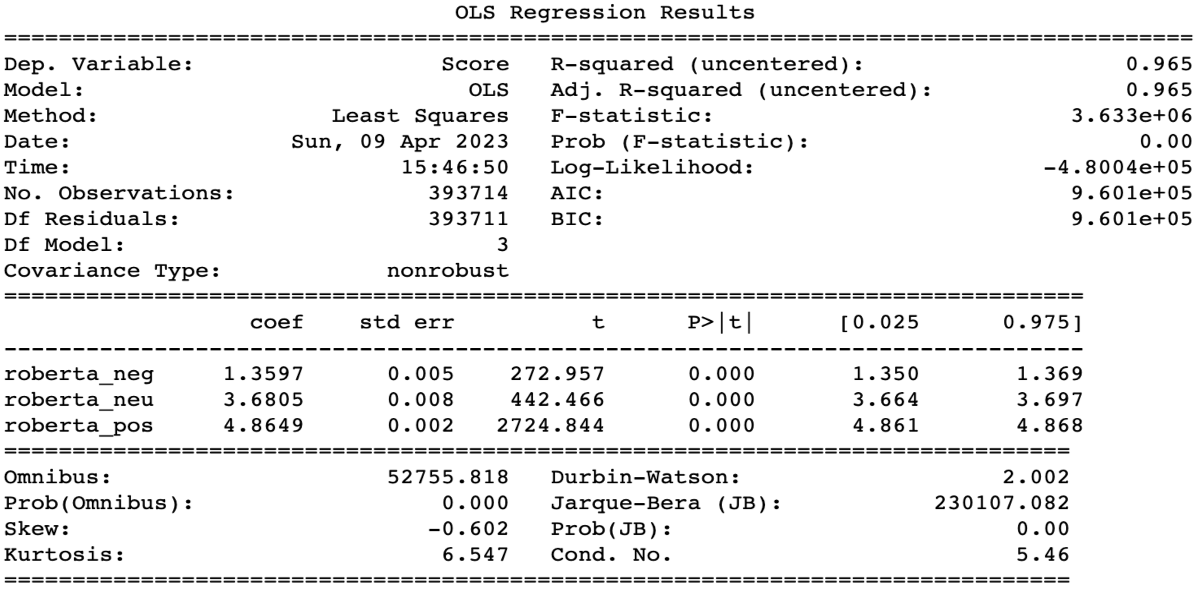
We can see the OLS regression summary of the VADER - Linear regression in figure 8.



```
                                 OLS Regression Results
=======================================================================================
Dep. Variable:                  Score   R-squared (uncentered):                   0.932
Model:                            OLS   Adj. R-squared (uncentered):              0.932
Method:                 Least Squares   F-statistic:                          1.790e+06
Date:                Sun, 09 Apr 2023   Prob (F-statistic):                        0.00
Time:                        15:46:46   Log-Likelihood:                     -6.1240e+05
No. Observations:              393714   AIC:                                  1.225e+06
Df Residuals:                  393711   BIC:                                  1.225e+06
Df Model:                           3
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
vader_neg     -3.3349      0.035    -95.868      0.000      -3.403      -3.267
vader_neu      3.7680      0.005    802.983      0.000       3.759       3.777
vader_pos      7.4880      0.014    522.508      0.000       7.460       7.516
==============================================================================
Omnibus:                    48836.014   Durbin-Watson:                   1.997
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            69074.232
Skew:                          -0.984   Prob(JB):                         0.00
Kurtosis:                       3.582   Cond. No.                         15.2
==============================================================================
```

Figure 8: Vader - Regression Summary

We can see the OLS regression summary of the RoBERTa - Linear regression in figure 9.



```
                                 OLS Regression Results                                
=======================================================================================
Dep. Variable:                  Score   R-squared (uncentered):                   0.965
Model:                            OLS   Adj. R-squared (uncentered):              0.965
Method:                 Least Squares   F-statistic:                          3.633e+06
Date:                Sun, 09 Apr 2023   Prob (F-statistic):                        0.00
Time:                        15:46:50   Log-Likelihood:                     -4.8004e+05
No. Observations:              393714   AIC:                                  9.601e+05
Df Residuals:                  393711   BIC:                                  9.601e+05
Df Model:                           3                                                  
Covariance Type:            nonrobust                                                  
===============================================================================
                  coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------------------------
roberta_neg     1.3597      0.005    272.957      0.000       1.350       1.369
roberta_neu     3.6805      0.008    442.466      0.000       3.664       3.697
roberta_pos     4.8649      0.002   2724.844      0.000       4.861       4.868
==============================================================================
Omnibus:                    52755.818   Durbin-Watson:                   2.002
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           230107.082
Skew:                          -0.602   Prob(JB):                         0.00
Kurtosis:                       6.547   Cond. No.                         5.46
==============================================================================
```

Figure 9: RoBERTa - Regression Summary

6.1 Performance on test data

The MSE on test data for both VADER and RoBERTa models are 1.31 and 0.66 respectively.

6.2 Inference

R-squared is the metric by which we can infer how well our linear regression model has fit the data, and as we can observe from figures 7 and 8, the R-squared values for VADER and RoBERTa models are 0.23 and 0.61 respectively.

Therefore we infer that RoBERTa model fits our data better than the VADER model, which is also consistent with the performance on the test data. That is the MSE on test data for VADER is greater than that for the RoBERTa model.

7 Model : Decision Tree Classifier

We have used the Decision tree classifier model to predict the ratings. (review score in this case).

Decision Tree : A decision tree is a non-parametric supervised learning algorithm, which is utilized for both classification and regression tasks. It has a hierarchical, tree structure, which consists of a root node, branches, internal nodes and leaf nodes.

As shown in the model pipeline in figure 10, we trained a Decision Tree model on the sentiment scores extracted from the VADER and RoBERTa models.

7.1 Performance on test data

The accuracy on test data for both VADER and RoBERTa models are 66.54% and 74.28% respectively.

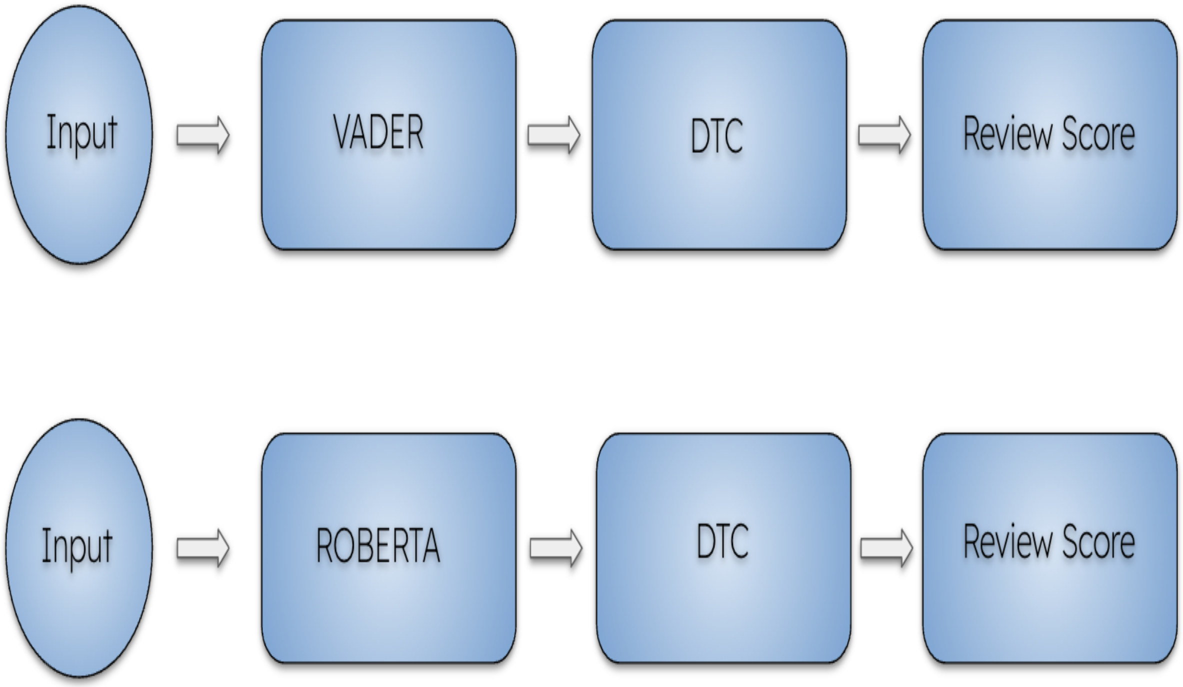
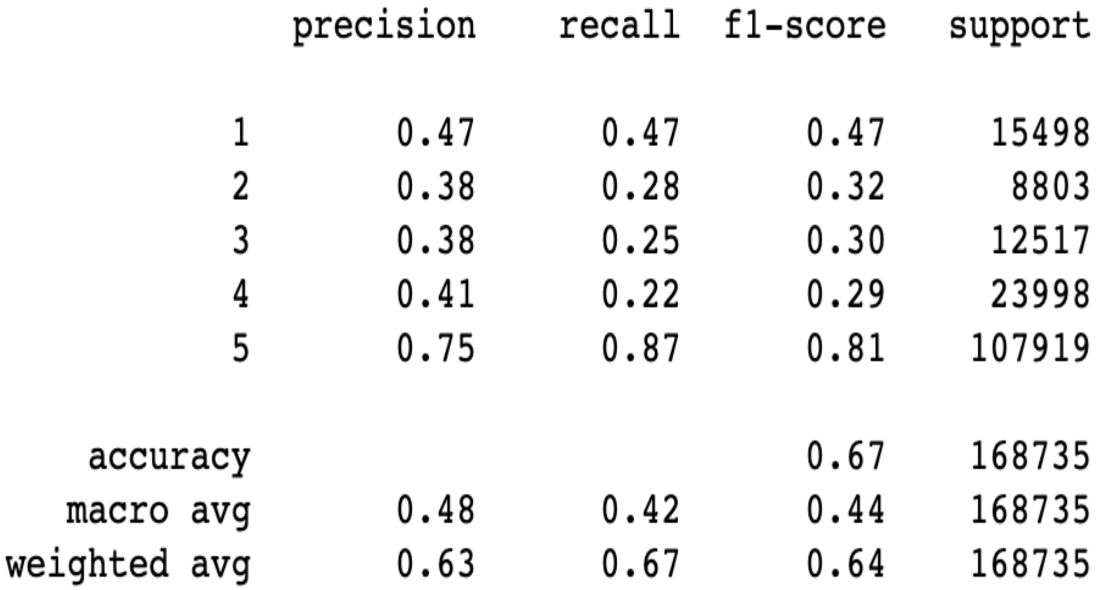


Figure 10: Vader and RoBERTa model pipelines for Decision Tree Classifier

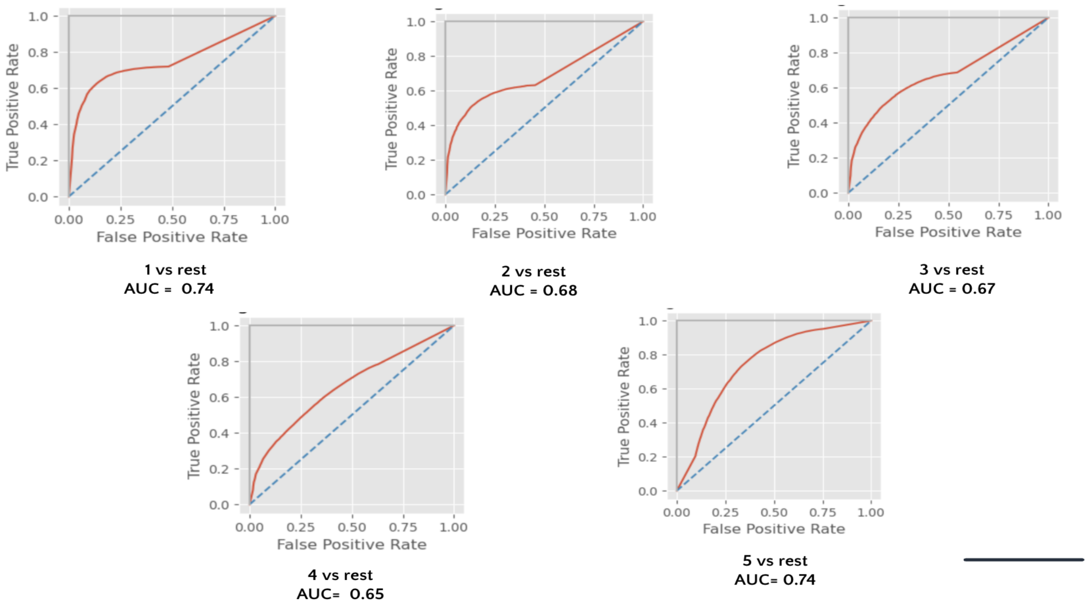
7.2 Inference

As shown in figure 11, according to the classification report for Vader-Decision Tree Classifier the accuracy is 67% and the F1 score for the classification is 0.44 The ROC curves for each class versus the rest are shown in figure 12 along with their AUC values.



```
              precision    recall  f1-score   support

           1       0.47      0.47      0.47     15498
           2       0.38      0.28      0.32      8803
           3       0.38      0.25      0.30     12517
           4       0.41      0.22      0.29     23998
           5       0.75      0.87      0.81    107919

    accuracy                           0.67    168735
   macro avg       0.48      0.42      0.44    168735
weighted avg       0.63      0.67      0.64    168735
```

Figure 11: Classification report for Vader Decision Tree model



1 vs rest
AUC = 0.74

2 vs rest
AUC = 0.68

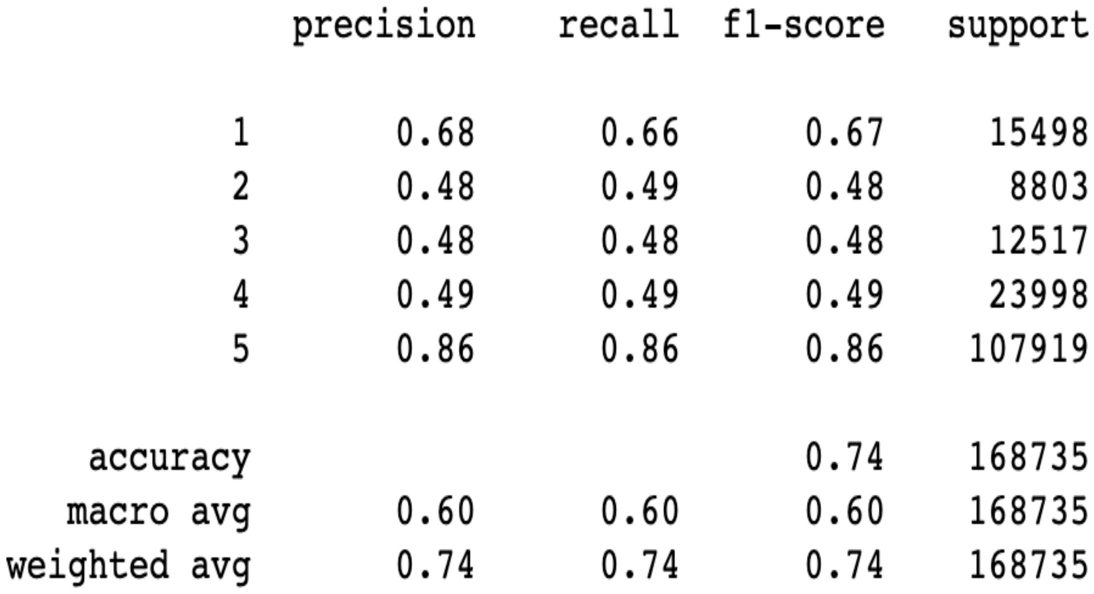
3 vs rest
AUC = 0.67

4 vs rest
AUC= 0.65

5 vs rest
AUC= 0.74

Figure 12: ROC curves for Vader Decision Tree model's each class vs the rest

As shown in figure 13, according to the classification report for Roberta-Decision Tree Classifier the accuracy is 74% and the F1 score for the classification is 0.60 The ROC curves for each class versus the rest are shown in figure 14 along with their AUC values.



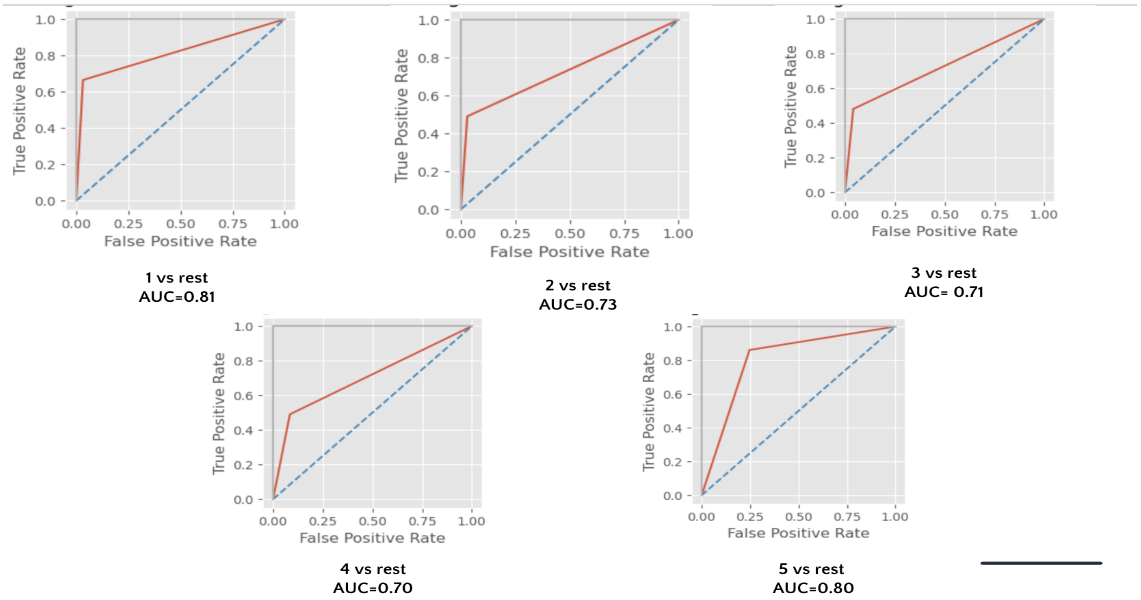
```
              precision    recall  f1-score   support

           1       0.68      0.66      0.67     15498
           2       0.48      0.49      0.48      8803
           3       0.48      0.48      0.48     12517
           4       0.49      0.49      0.49     23998
           5       0.86      0.86      0.86    107919

    accuracy                           0.74    168735
   macro avg       0.60      0.60      0.60    168735
weighted avg       0.74      0.74      0.74    168735
```

Figure 13: Classification report for RoBERTa Decision Tree model

From the above figures we can infer that Roberta model performs better than Vader, as all the metrics like F1, accuracy, precision and recall are greater than that of the vader's.

8 Model : Random Forest Classifier

We have used the Random Forest classifier model to predict the ratings. (review score in this case).



1 vs rest
AUC=0.81

2 vs rest
AUC=0.73

3 vs rest
AUC= 0.71

4 vs rest
AUC=0.70

5 vs rest
AUC=0.80

Figure 14: ROC curves for RoBERTa Decision Tree model's each class vs the rest

Random Forest Classifier : Random forests or random decision forests is an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time. For classification tasks, the output of the random forest is the class selected by most trees.

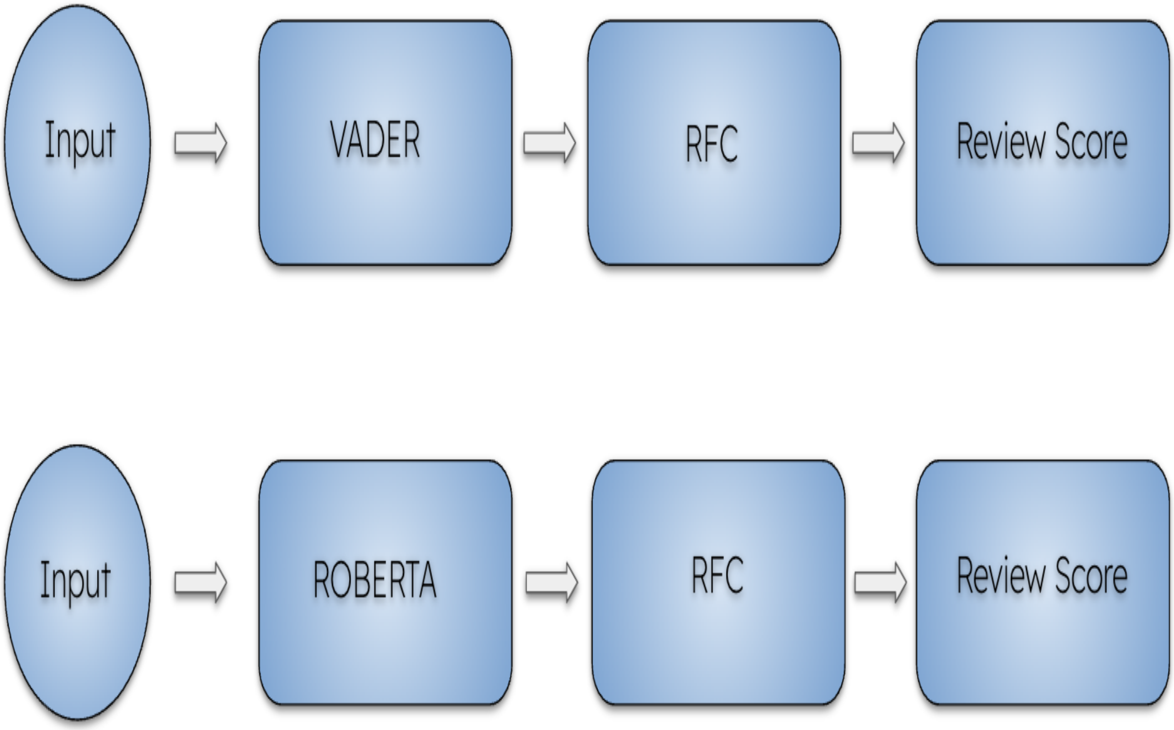


Figure 15: Vader and RoBERTa model pipelines for Random Forest Classifier

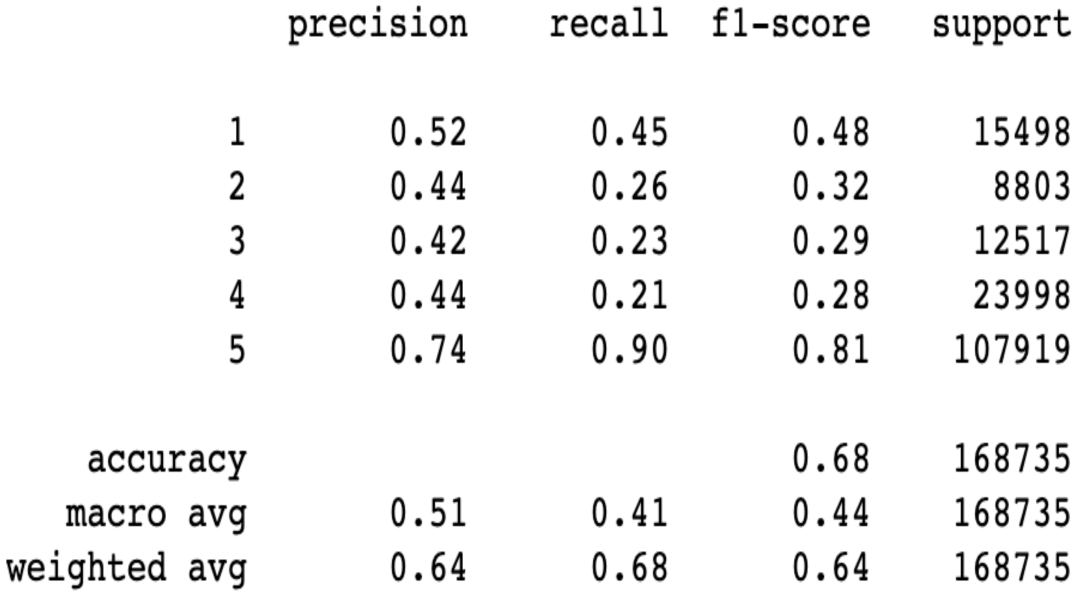
As shown in the model pipeline in figure 15, we trained a Random Forest model on the sentiment scores extracted from the VADER and RoBERTa models.

8.1 Performance on test data

The accuracy on test data for both VADER and RoBERTa models are 67.99% and 78.45% respectively.

8.2 Inference

As shown in figure 16, according to the classification report for Vader-Random Forest Classifier the accuracy is 0.68 and the F1 score for the classification is 0.44. The ROC curves for each class versus the rest are shown in figure 17 along with their AUC values.



```
              precision    recall  f1-score   support

           1       0.52      0.45      0.48     15498
           2       0.44      0.26      0.32      8803
           3       0.42      0.23      0.29     12517
           4       0.44      0.21      0.28     23998
           5       0.74      0.90      0.81    107919

    accuracy                           0.68    168735
   macro avg       0.51      0.41      0.44    168735
weighted avg       0.64      0.68      0.64    168735
```

Figure 16: Classification report for Vader Random Forest model

As shown in figure 18, according to the classification report for Roberta-Random Forest Classifier the accuracy is 0.79 and the F1 score for the classification is 0.63. The ROC curves for each class versus the rest are shown in figure 19 along with their AUC values.

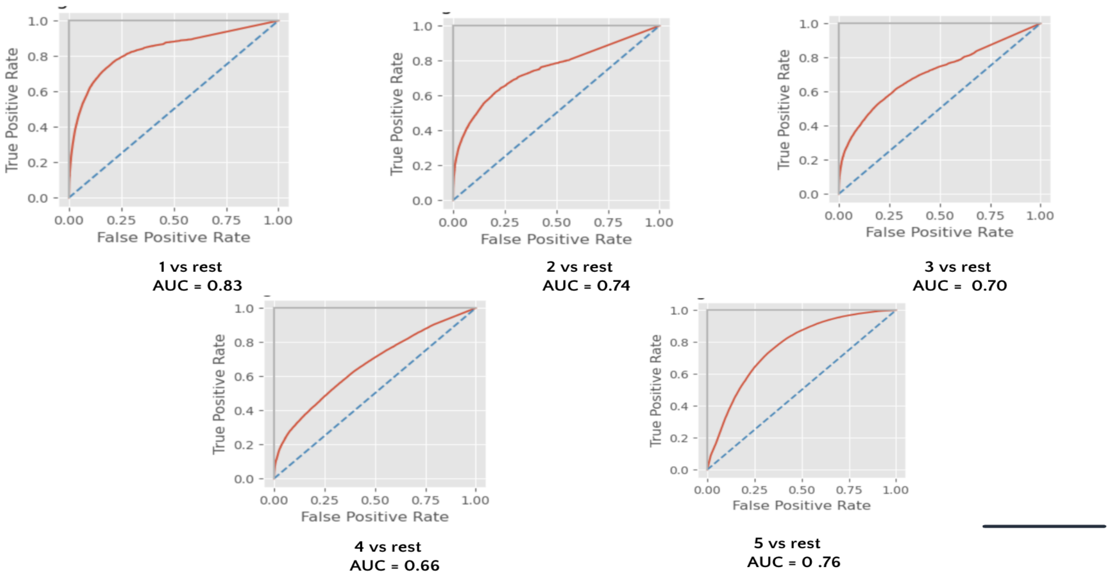
From the above figures we can infer that Roberta model performs better than Vader, as all the metrics like F1, accuracy, precision and recall are greater than that of the vader's.

9 Model : Decision Tree Regressor

We have used the Decision Tree Regressor model to predict the ratings. (review score in this case).

Decision Tree Regressor :Decision tree builds regression or classification models in the form of a tree structure. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes.

As shown in the model pipeline in figure 20, we trained a Decision Tree Regressor model on the sentiment scores extracted from the VADER and RoBERTa models.



1 vs rest
AUC = 0.83

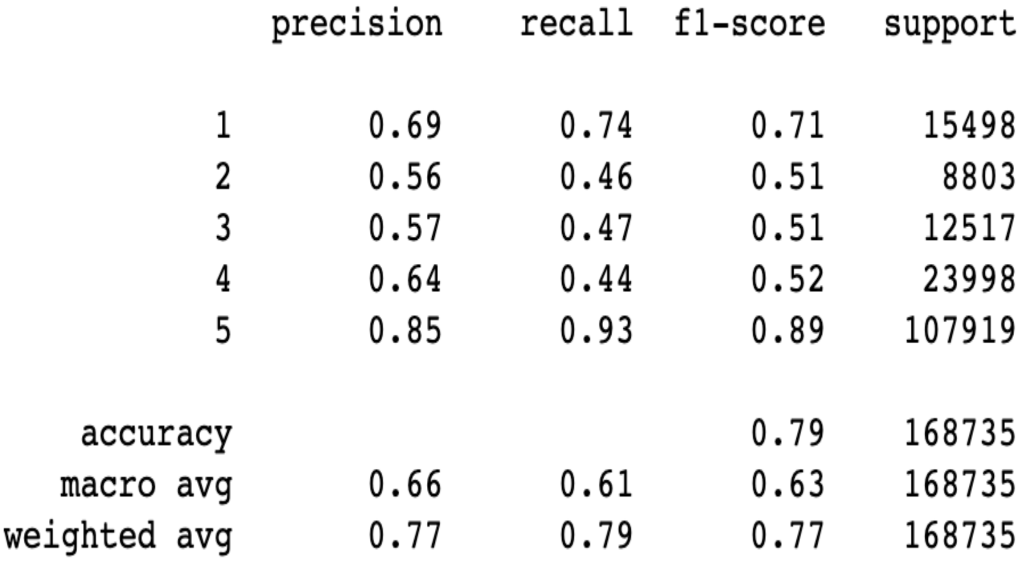
2 vs rest
AUC = 0.74

3 vs rest
AUC = 0.70

4 vs rest
AUC = 0.66

5 vs rest
AUC = 0 .76

Figure 17: ROC curves for Vader Random Forest model's each class vs the rest



```
              precision    recall  f1-score   support

           1       0.69      0.74      0.71     15498
           2       0.56      0.46      0.51      8803
           3       0.57      0.47      0.51     12517
           4       0.64      0.44      0.52     23998
           5       0.85      0.93      0.89    107919

    accuracy                           0.79    168735
   macro avg       0.66      0.61      0.63    168735
weighted avg       0.77      0.79      0.77    168735
```

Figure 18: Classification report for RoBERTa Random Forest model

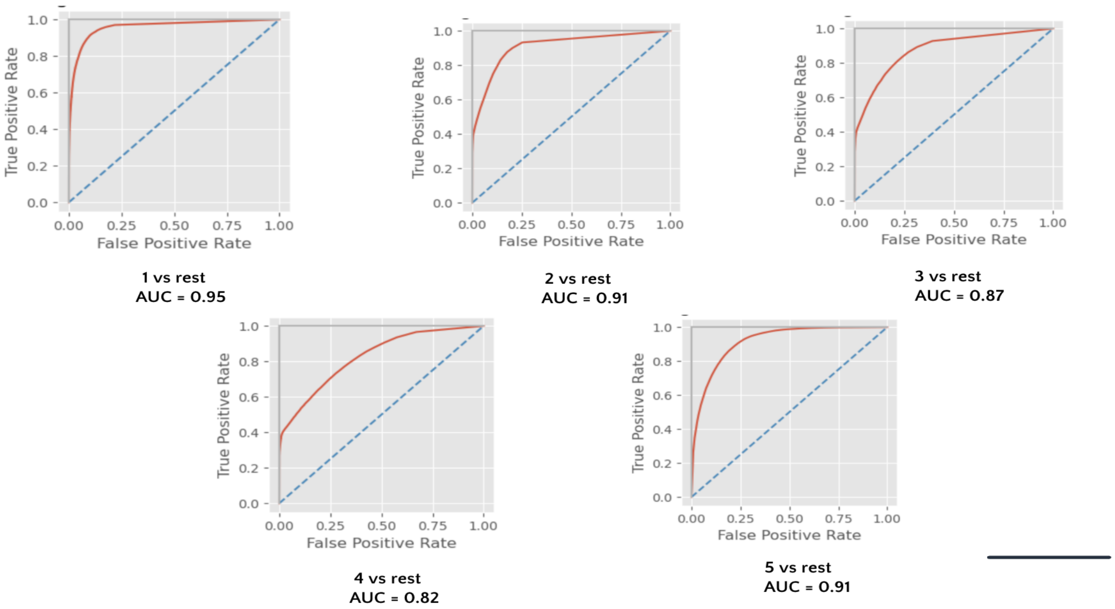
9.1 Performance on test data

The MSE on test data for both VADER and RoBERTa models are 1.150285 and 0.909662 respectively.

9.2 Inference

R-squared is the metric by which we can infer how well our decision tree regressor model has fit the data, the R-squared values for VADER and RoBERTa models are 0.22 and 0.51 respectively.

Therefore we infer that RoBERTa model fits our data better than the VADER model, which is also consistent with the performance on the test data. That is the MSE on test data for VADER is greater than that for the RoBERTa model.



1 vs rest
AUC = 0.95

2 vs rest
AUC = 0.91

3 vs rest
AUC = 0.87

4 vs rest
AUC = 0.82

5 vs rest
AUC = 0.91

Figure 19: ROC curves for RoBERTa Random Forest model's each class vs the rest

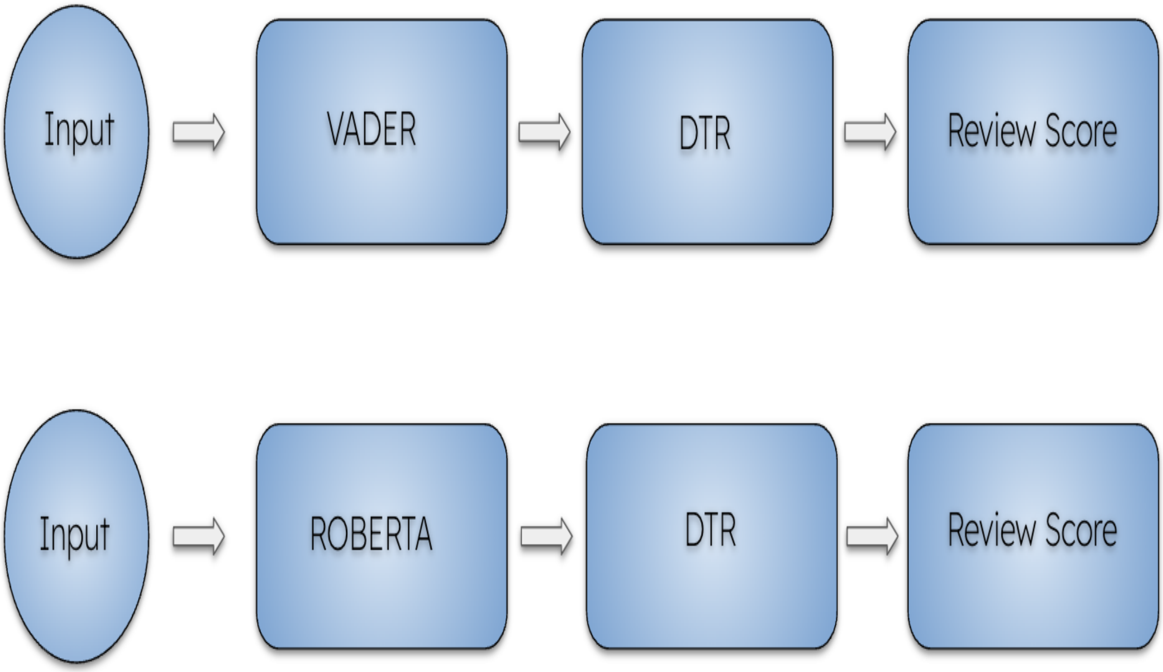


Figure 20: Vader and RoBERTa model pipelines for Decision Tree Regressor

10 Model : Random Forest Regressor

We have used the Random Forest regressor model to predict the ratings. (review score in this case).

Random Forest Regressor : Random Forest Regression is a supervised learning algorithm that uses ensemble learning method for regression. Ensemble learning method is a technique that combines predictions from multiple machine learning algorithms to make a more accurate prediction than a single model.

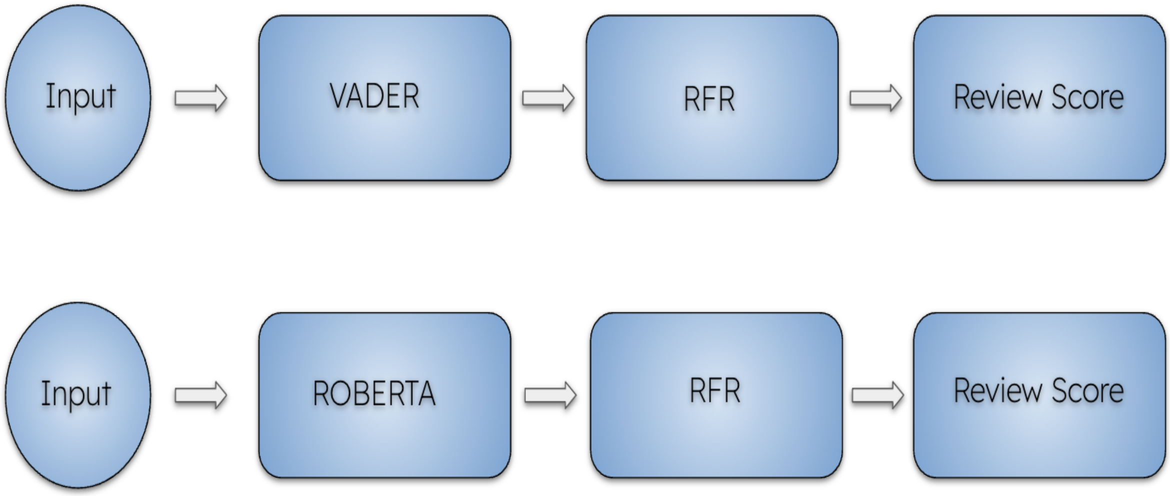


Figure 21: Vader and RoBERTa model pipelines for Random Forest Regressor

As shown in the model pipeline in figure 21, we trained a Random Forest Regressor model on the sentiment scores extracted from the VADER and RoBERTa models.

10.1 Performance on test data

The MSE on test data for both VADER and RoBERTa models are 1.092048 and 0.704002 respectively.

10.2 Inference

R-squared is the metric by which we can infer how well our random forest regressor model has fit the data, the R-squared values for VADER and RoBERTa models are 0.30 and 0.71 respectively.

Therefore we infer that RoBERTa model fits our data better than the VADER model, which is also consistent with the performance on the test data. That is the MSE on test data for VADER is greater than that for the RoBERTa model.

11 Summary of Machine Learning Models

As observed in EDA, we see a negative correlation between review scores and negative sentiment probabilities and positive correlation between review scores and positive sentiment probabilities. This led us to try a Linear regression model. Although the response variable is discrete since there is a natural order among the values regression seemed a reasonable choice. Hence we tried the above mentioned regression models.

Since the response variable is a discrete variable, each value could be treated as a individual class. Although treating the response variable as such we are ignoring the natural order present among the values. It still seemed like a worthwhile modeling choice and the results mentioned in the above sections proved it. The best classifier model is giving us about an accuracy of about 80%.

As we can see from the above mentioned models that Roberta-Random Forest Classifier is performing the best among classifiers and Roberta-Random Forest Regressor is performing the best among the regression models.

In Classification we ignore the natural order present between the ratings and consider them as individual classes. But in regression we consider the natural order between the ratings. So for this reason we chose Roberta-Random Forest Regressor as our best model.

11.1 Prediction for cases of interest for Roberta-Random Forest Classifier

Here, we took 3 texts as follows :
Text1 : 'I absolutely loved the fajitas.'
Text2 : 'The starters are great, but the main course is not good.'
Text3 : 'The food here is unbelievably bad.'

So naturally, Text1 should incur a positive score, Text2 should get a medium score and Text3 should get a low score using the models.

Here are the resulting scores for all three cases using the RoBERTa model :
Text1 = 5
Text2 = 3
Text3 = 1

11.2 Prediction for cases of interest for Roberta-Random Forest Regressor

Here, we took 3 texts as follows :
Text1 : 'I absolutely loved the fajitas.'
Text2 : 'The starters are great, but the main course is not good.'
Text3 : 'The food here is unbelievably bad.'

So naturally, Text1 should incur a positive score, Text2 should get a medium score and Text3 should get a low score using the models.

Here are the resulting scores for all three cases using the RoBERTa model :
Text1 = 4.93
Text2 = 2.46
Text3 = 1.05

12 Summary and Conclusion

Our Machine Learning model solves one of the challenges faced by online market places that is predicting a score based on the reviews left by the customers. There could be many applications of the scores, such as, they could be used in product recommendations, improving services of the products, adding new features to a product. The scores generated can be further analysed by the data analysts of the product company to analyse where the product could improve of if the product should continue in the production etc. Our bench mark model can be reliably used for generating the scores.

As mentioned above, the product domain experts can review the scores generated and further analyse the reviews of the product to see if customers aren't happy with any particular segment of the product or there are more than one aspects of the product that they are not satisfied with. This analysis helps them in decision making.

While our model is performing reasonably well, I can think of at least two parts in the modeling where our performance could potentially improve. First, we are only extracting positive, neutral, and negative sentiment probabilities right now. If we can extract more features, like high positive sentiment probability, Weak positive sentiment probability, High neutral sentiment probability, weak neutral sentiment probability, high negative sentiment probability and weak negative sentiment probability from Roberta model through fine-tuning, I think the model built on these six features would perform much better. Of course there is also a danger of over-fitting. In that case we can regulate the number of features we extract from the Roberta model. We can also try and see extracting more features if there is still scope for improvement, all the while being wary of regularisation.

Second part where I think our model could improve is when we choose second model in our model pipeline. The models we tried Linear Regression, Random Forest and Decision Trees are simple models. A more complex model can capture more complex patterns and hence give better performance. So I think a Feed Forward network model would further improve the model. Of course, there is risk of over-fitting. If there is over-fitting I reckon we can use some regularisation methods to reduce the over-fitting.

13 Acknowledgement

Our thanks to Prof. Nina Hubig, and Ms. Nushrat Humaira for all the help and clarifying the doubts we had throughout the project development.

14 References

DATASET - https://www.kaggle.com/datasets/datafiniti/consumer-reviews-of-amazon-products//
METRICS - https://towardsdatascience.com/various-ways-to-evaluate-a-machine-learning-models-performance-230449055f15