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

Reviews left on products on popular e-commerce websites often describe characteristics of the service or the product itself. Reviews related to the offered service/transportation impact the seller's reputation in general, and reviews related to the product should only reflect the product's performance or quality. In this project, the students will collect public reviews and infer the rating from the seller and the product itself.

1 Introduction

In recent years, online markets have become increasingly popular, providing a vast range of products and services to customers worldwide. With the rise of these platforms, the amount of user-generated reviews has also grown significantly. These reviews can be a valuable source of information for customers, allowing them to make informed decisions about products they wish to purchase.

One of the challenges faced by online marketplaces is predicting the rating that a product will receive based on the reviews it has received. This is an important task as it allows for better product recommendations, improved customer satisfaction, and increased sales. Machine learning algorithms can be used to analyze and extract insights from the reviews, and predict the rating that a product is likely to receive.

This topic is of significant interest to online marketplaces, as it can help them better understand their customers' needs and preferences, and provide more personalized recommendations. Additionally, it can help sellers to identify areas for improvement in their products, and adjust their marketing strategies accordingly.

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 Data Set

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.

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 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.

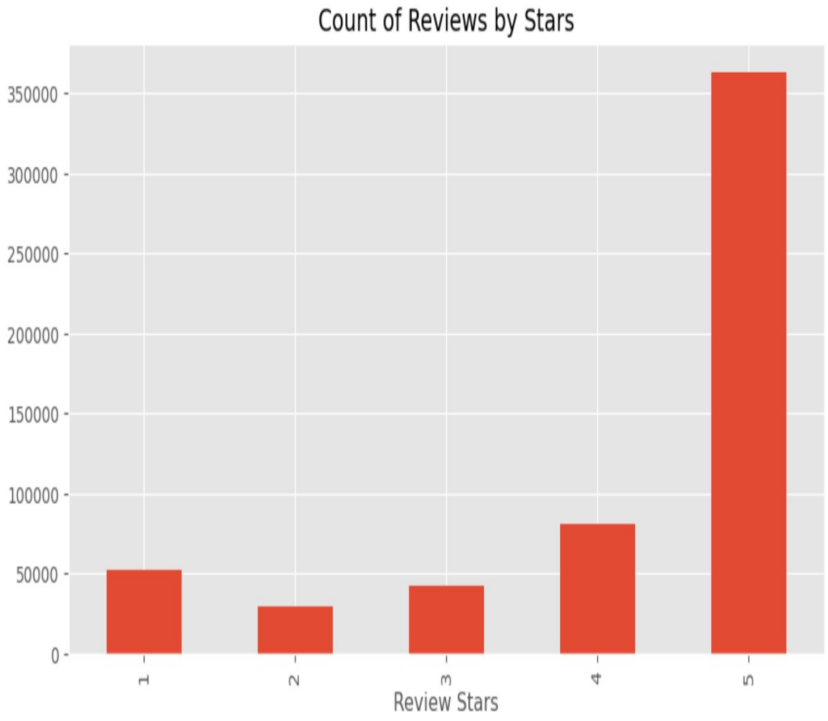


Figure 1: Count of Review by Scores

4.2 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 Score vs Positive Ratings for Vader Model

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.

4.4 Score vs Neutral Ratings for Vader Model

In Figure 4, We see that there is almost no correlation between neutral probability and score as this is to be expected. Since the neutral probability means the review left by the customer has a neutral sentiment associated with it and this translates to a higher score.

4.5 Score vs Negative Ratings for Vader Model

In Figure 5, 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.

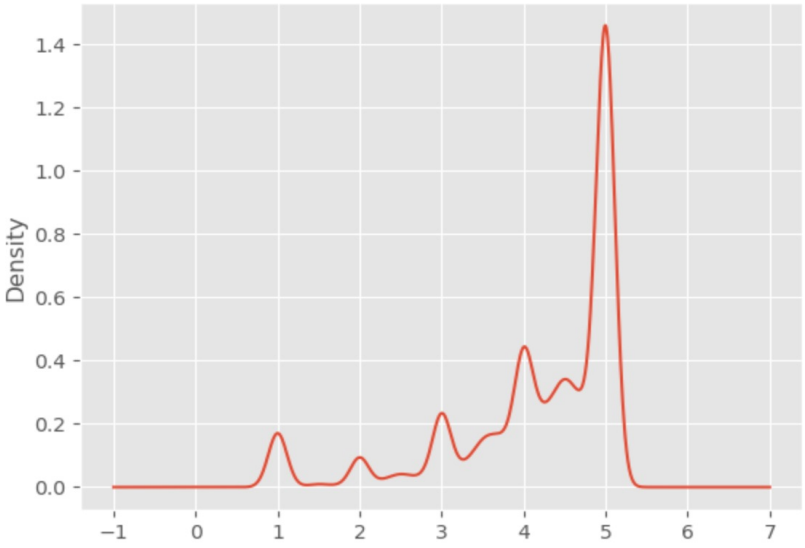


Figure 2: Kde plot for Average product Score

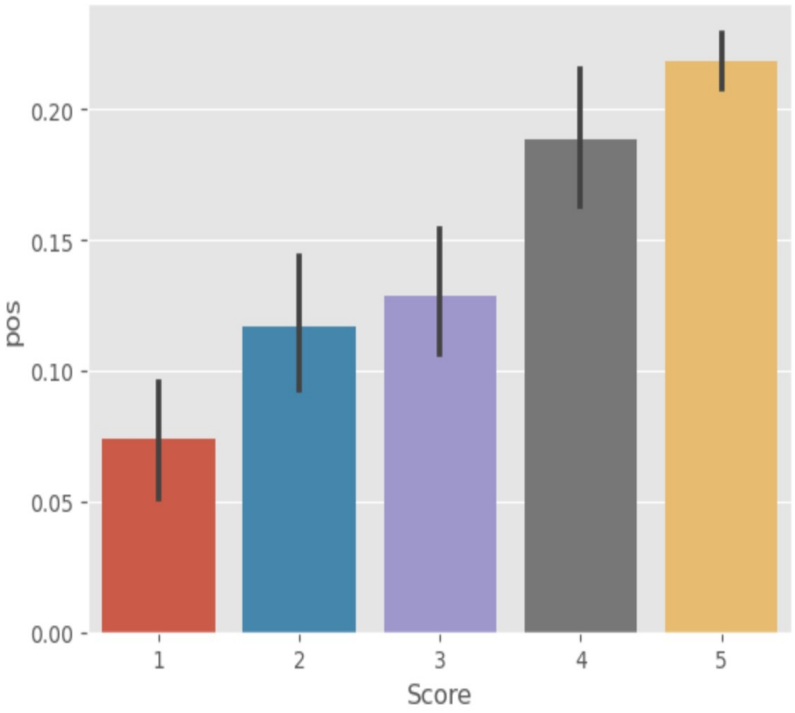


Figure 3: Positive probability vs Score

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 above as shown in figure 6.

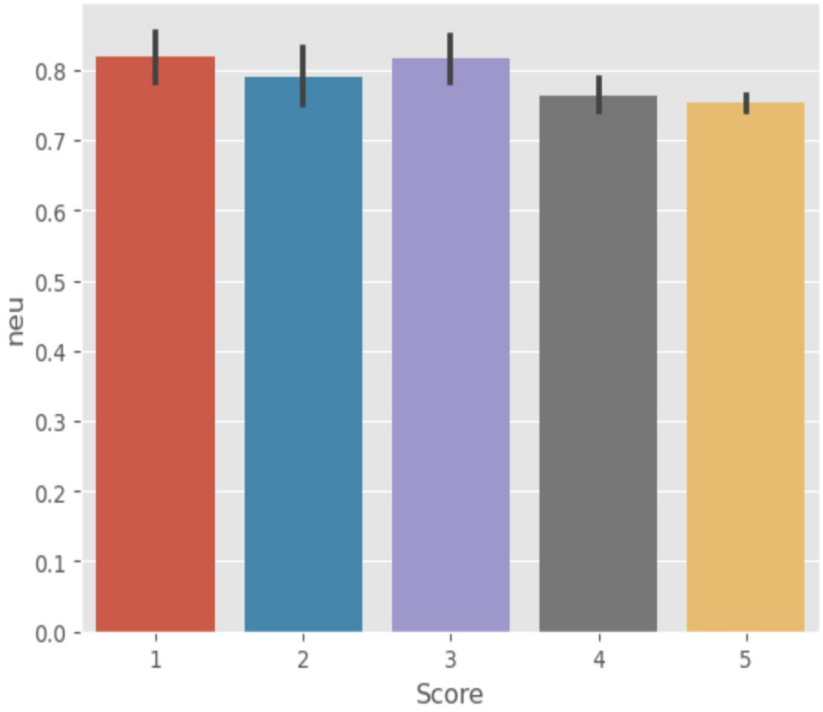


Figure 4: Neutral probability vs Score

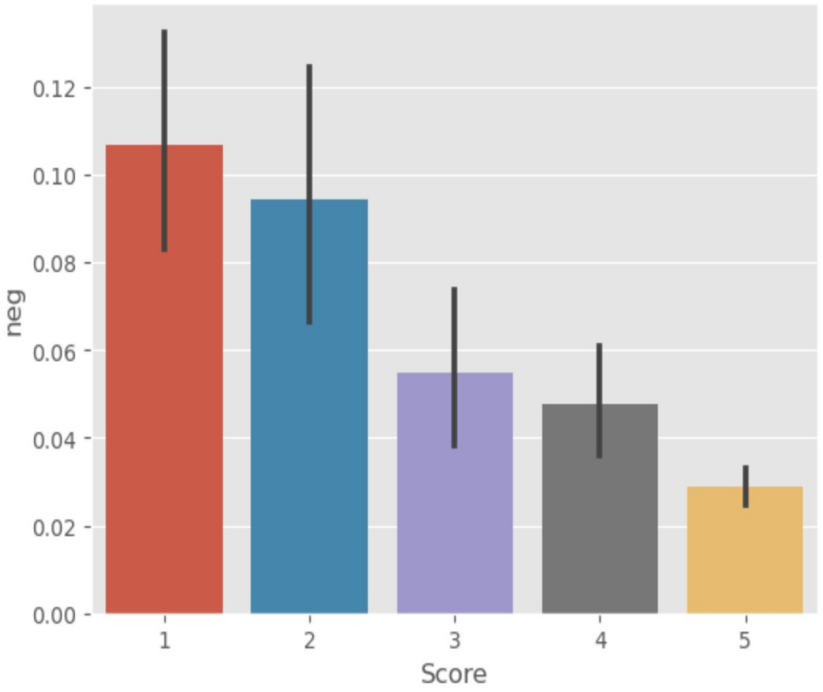


Figure 5: Negative probability vs Score

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 6.

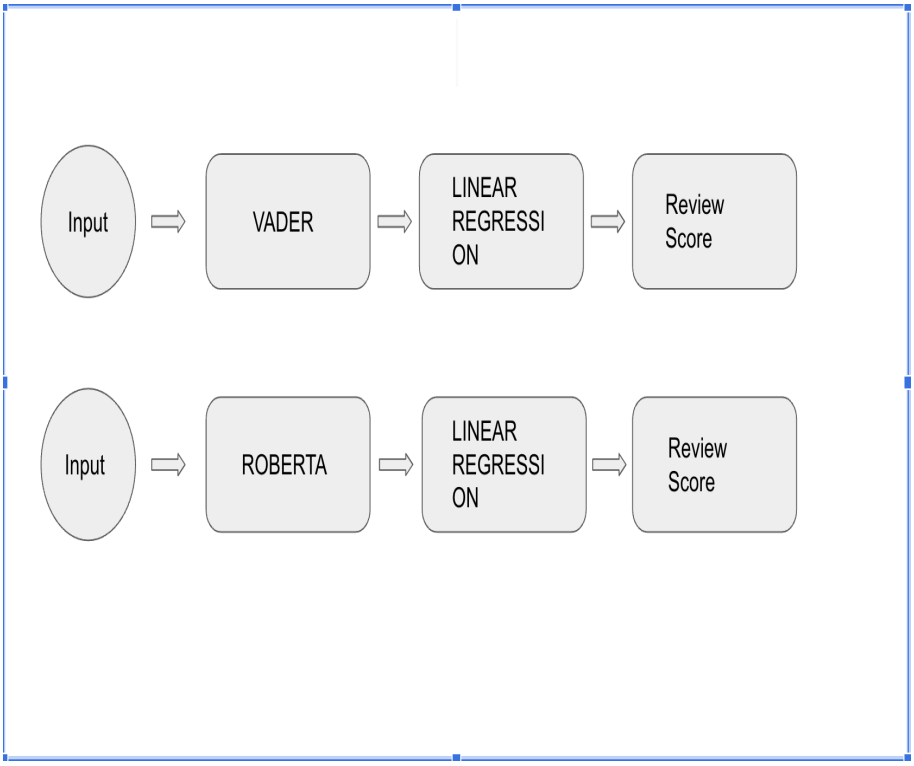


Figure 6: Vader and RoBERTa model pipelines

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).

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 7.

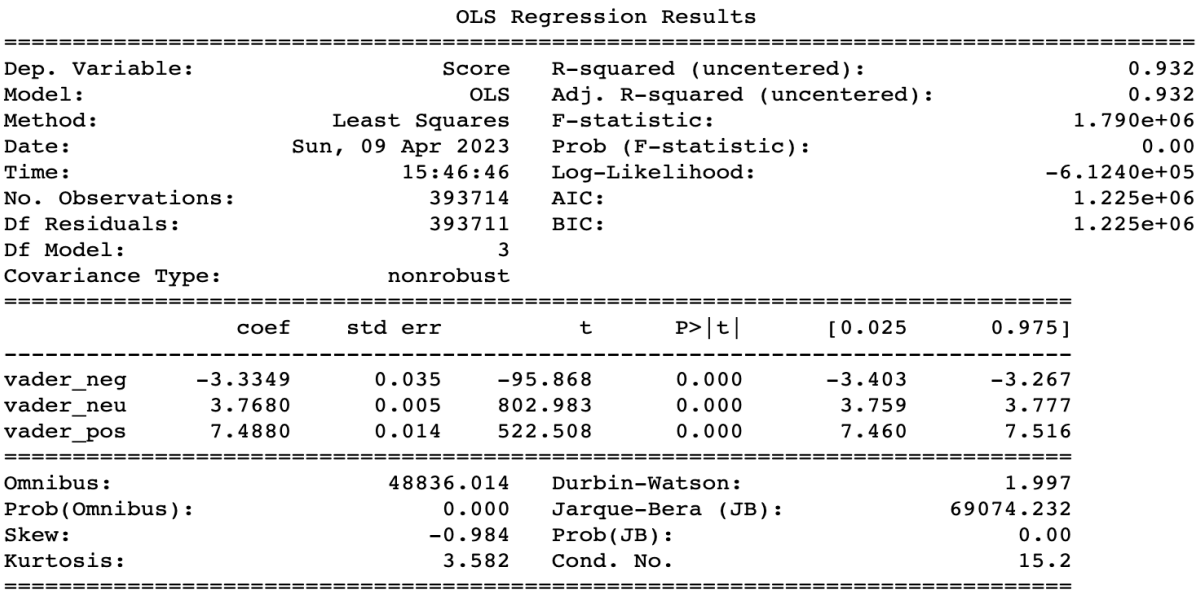
We can see the OLS regression summary of the RoBERTa - Linear regression in figure 8.

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.



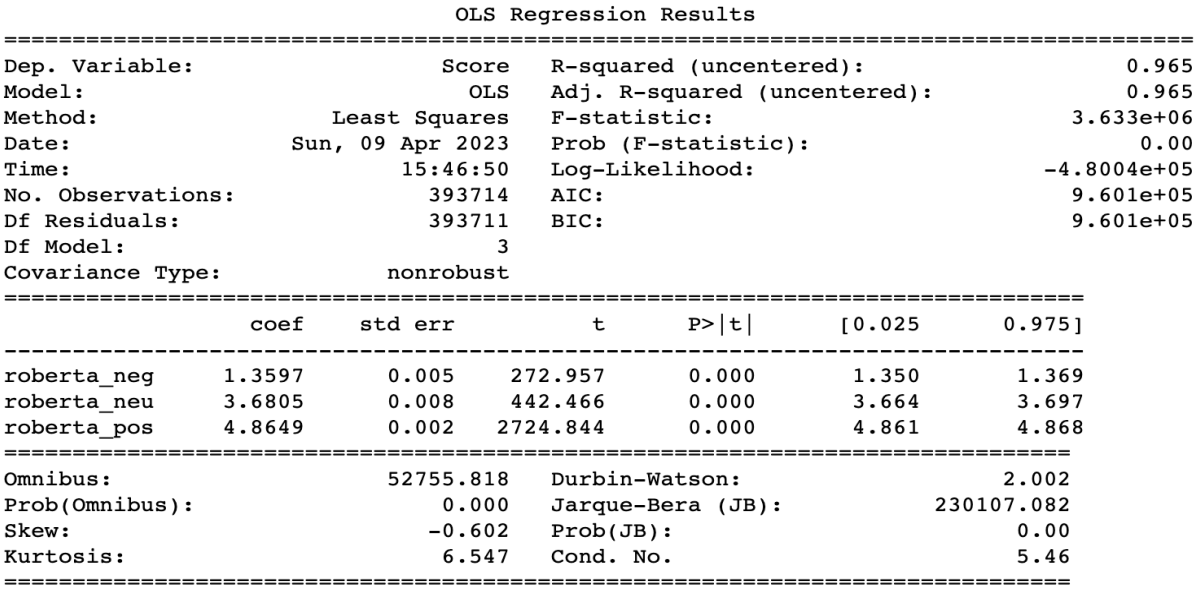
OLS Regression Results

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 7: Vader - Regression Summary



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 8: RoBERTa - Regression Summary

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.

6.3 Prediction for cases of interest

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 give a neutral score and Text3 should give us a negative score using the models.

Here are the resulting scores for all three cases using the VADER model :
Text1 = 5
Text2 = 2.9094616370870767
Text3 = 1

Here are the resulting scores for all three cases using the RoBERTa model :
Text1 = 4.842823739247891
Text2 = 2.494654918891032
Text3 = 1.4079366255270431

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..

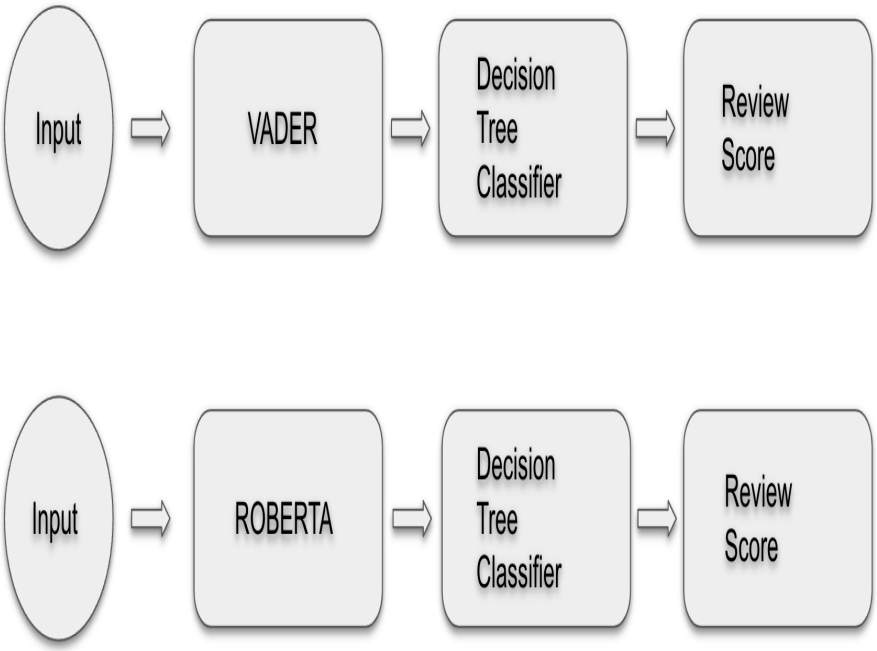


Figure 9: Vader and RoBERTa model pipelines for Decision Tree Classifier

As shown in the model pipeline in figure 9, 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.

7.2 Inference

Accuracy is the metric by which we can infer how well our classifier model has fit the data. Therefore we infer that RoBERTa model fits our data better than the VADER model. This is also consistent with the model complexity as Vader is a simple model compared to Roberta, so it is reasonable to conclude that Roberta model can capture more complex patterns avoiding the under-fitting problem.

7.3 Prediction for cases of interest

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 give a neutral score and Text3 should give us a negative score using the models.

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

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

8 Model : Random Forest Classifier

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

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..

As shown in the model pipeline in figure 10, 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

Accuracy is the metric by which we can infer how well our classifier model has fit the data. Therefore we infer that RoBERTa model fits our data better than the VADER model. This is also consistent with the model complexity as Vader is a simple model compared to Roberta, so it is reasonable to conclude that Roberta model can capture more complex patterns avoiding the under-fitting problem.

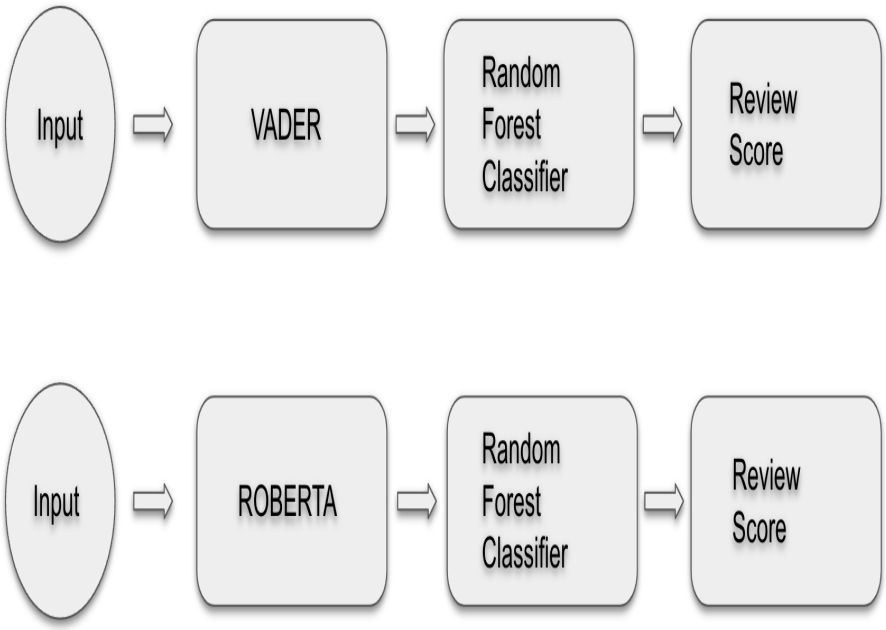


Figure 10: Vader and RoBERTa model pipelines for Random Forest Classifier

8.3 Prediction for cases of interest

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 give a neutral score and Text3 should give us a negative score using the models.

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

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

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.

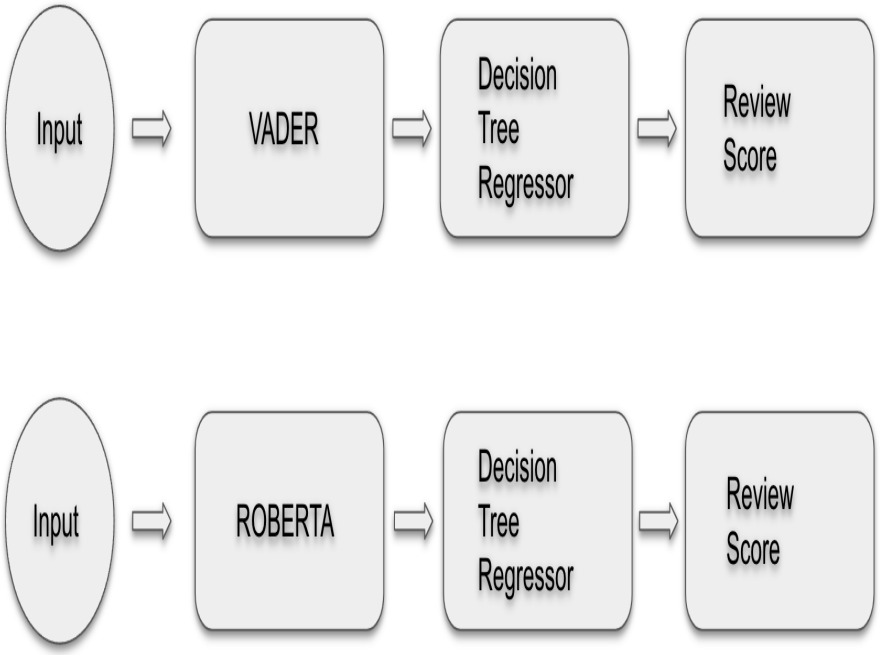


Figure 11: Vader and RoBERTa model pipelines for Decision Tree Regressor

As shown in the model pipeline in figure 11, we trained a Decision Tree Regressor model on the sentiment scores extracted from the VADER and RoBERTa models.

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.

9.3 Prediction for cases of interest

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 give a neutral score and Text3 should give us a negative score using the models.

Here are the resulting scores for all three cases using the VADER model :
Text1 = 3.0
Text2 = 5.0
Text3 = 1.0

Here are the resulting scores for all three cases using the RoBERTa model :
Text1 = 5.0
Text2 = 3.0
Text3 = 1.0

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.

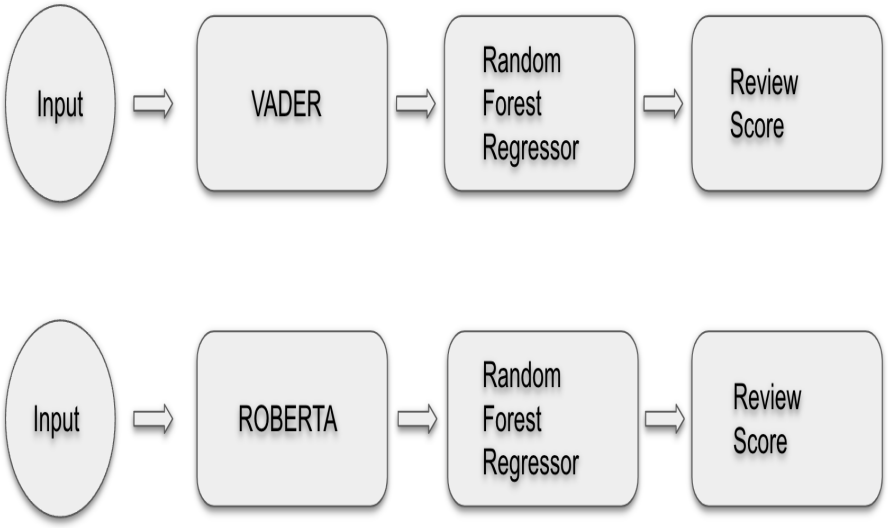


Figure 12: Vader and RoBERTa model pipelines for Random Forest Regressor

As shown in the model pipeline in figure 12, 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.

10.3 Prediction for cases of interest

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 give a neutral score and Text3 should give us a negative score using the models.

Here are the resulting scores for all three cases using the VADER model :
Text1 = 3.74
Text2 = 4.08
Text3 = 1.56

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