Sentiment Analysis for Amazon Reviews

Mrudula Padhebettu
mrudula2@illinois.edu

Introduction

Purchasing a product is an interaction between two entities, the consumers, and the business owners. The consumer reviews for products serve as feedback for businesses in terms of performance, product quality, and consumer service. Consumers use reviews to make decisions about what products to buy, on the other hand, businesses sell their products and receive feedback in terms of consumer reviews. Many consumers make decisions based on the reviews and opinions of other consumers and hence consumer experience has great influence today. In the last few years, consumer ways of expressing their opinions and feelings have changed according to the changes in social networks, virtual communities, and other social media communities. Discovering large amounts of data from unstructured data on the web has become an important challenge due to its importance in different areas of life. Sentiment analysis of product reviews has recently become very popular in text mining and computational linguistics research.

Problem Statement

In this paper, we will classify the positive and negative amazon reviews of the customer and build a supervised learning model to polarize large amounts of reviews. After extracting the features of the dataset, we will analyze the different supervised models we have built. The models are Logistic Regression, Random Forest Classifier, Naive Bayes, and Ridge Regression. We will then calculate and compare the accuracy of these models to get a better understanding of the polarized attitudes towards the products.

Methodology and Implementation

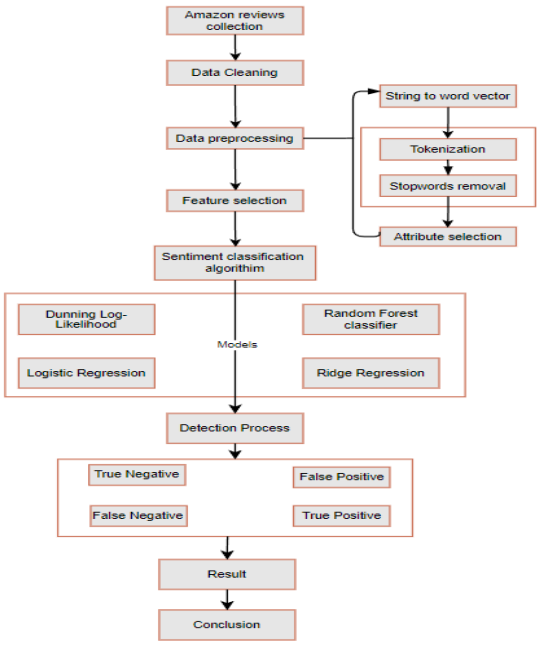

Fig. 1. Steps used in the supervised learning approach

The methodology is organized as shown in Fig. 1. Amazon Reviews Collection - The experiment is based on analyzing the standard dataset's sentiment value using machine learning algorithms. Data Cleaning - The dataset used in our experiment is obtained from Amazon product data and was divided into five scales rating: 1 star, 2 stars, 3 stars, 4 stars and 5 stars. To clean the data, we have deleted some blank rows that can cause confusion in the analysis process. Data Preprocessing - Data preprocessing is a significant step in the text mining process and plays an important part in several supervised learning techniques. Data preprocessing is broken into String to word vector, Tokenization, Stopwords removal and Attribute selection. The models worked on are Dunning Log-Likelihood, Logistic Regression, Random Forest classifier and Ridge Regression. These are explained in detail in the below section. Detection Processes - This step consists in predicting the models output on testing the datasets and then generating a confusion matrix. Comparison of Results – In the final step, we are comparing the different accuracy and precision provided by the Amazon reviews datasets using different classification algorithms and identified the algorithm that was the most significant in the detection of positive and negative reviews.

Data Collection

The dataset used for this research was obtained from Kaggle. The data collected from amazon is about Consumer Reviews of Amazon products. The dataset consists of 34660 reviews and 21 columns. The names of all the columns are listed in Fig 3. Each example includes the type, name of the product as well as the text review and the rating of the product. The other important details are captured in Fig 2 which shows us an actual Amazon customer review sample.

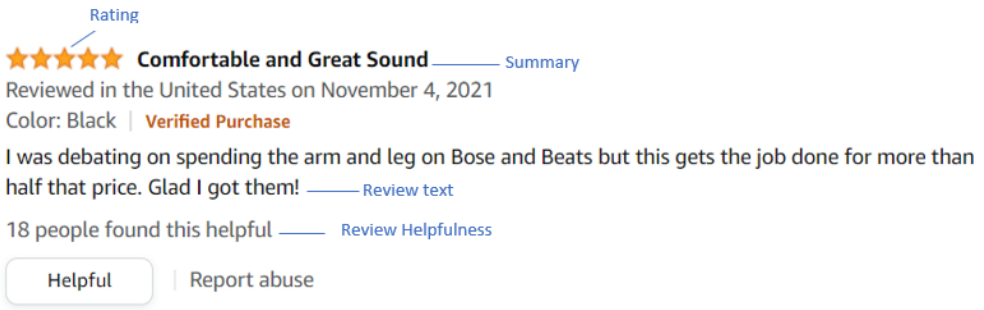

Fig. 2. Actual amazon customer review sample

An Amazon user review consists of four important aspects –
Summary: The title of the review
Review text: The actual content of the review
Rating: User rating of the product on a scale of 1 to 5
Helpfulness: The number of people who found the review useful
Other important columns are Asin which is Amazon Standard Identification Number, a ten-digit alphanumeric code that identifies products on Amazon and reviews.doRecommend which is a binary value that shows us if a review is positive or negative.

Next using heat map, we will analyze the columns with null values. Fig.4 shows a heat map with a graphical representation of data that translates data values into colors within a matrix. This type of data visualization summarizes a vast amount of data within a single snapshot

which helps to quickly communicate relationships between data values. A lot of the columns with null values like reviews.userCity, reviews.didPurchase, reviews.id and reviews.userProvi nce will not be used by us in the analysis. We have a few missing data in reviews.rating and reviews.numHelpful hence we will be dropping the null values from both these columns. The final number of rows in our dataset is 34131.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 34660 entries, 0 to 34659
Data columns (total 21 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   id                    34627 non-null  object
 1   name                  27868 non-null  object
 2   asins                 34625 non-null  object
 3   brand                 34627 non-null  object
 4   categories            34627 non-null  object
 5   keys                  34627 non-null  object
 6   manufacturer          34627 non-null  object
 7   reviews.date          34598 non-null  object
 8   reviews.dateAdded     24039 non-null  object
 9   reviews.dateSeen      34627 non-null  object
 10  reviews.didPurchase   1 non-null      object
 11  reviews.doRecommend   34066 non-null  object
 12  reviews.id            1 non-null      float64
 13  reviews.numHelpful    34131 non-null  float64
 14  reviews.rating        34627 non-null  float64
 15  reviews.sourceURLs    34627 non-null  object
 16  reviews.text          34626 non-null  object
 17  reviews.title         34622 non-null  object
 18  reviews.userCity      0 non-null      float64
 19  reviews.userProvince  0 non-null      float64
 20  reviews.username      34625 non-null  object
dtypes: float64(5), object(16)
memory usage: 5.6+ MB
```

Fig. 3. List of all the columns in the dataset

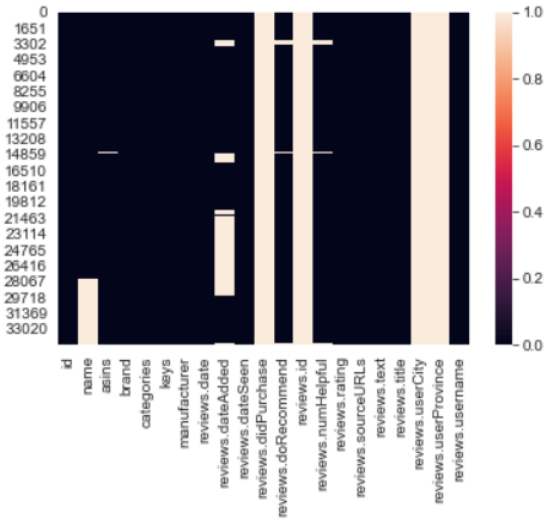

Fig. 4. Heat map showing the null values

Data Preprocessing

Next, we are analyzing the distribution of the ratings. Figure 5 shows us the rating distribution between rating 1 to 5. These five classes are imbalanced as class 1 and class 2 have small amount of data while class 5 has more reviews. Below is an example of one of the samples from our dataset.

Review text: The visual quality on the Fire 8 HD is amazing. It runs very fast and is easy to use, very durable.

Rate: '5'

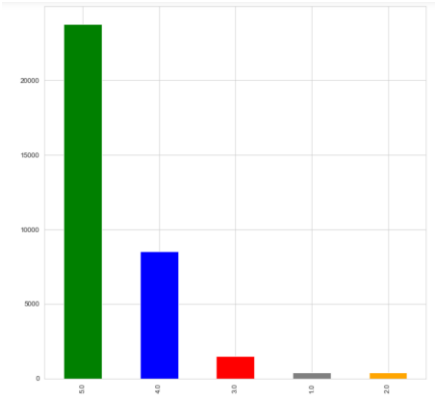

Fig 5. Rating Distribution of Amazon Reviews

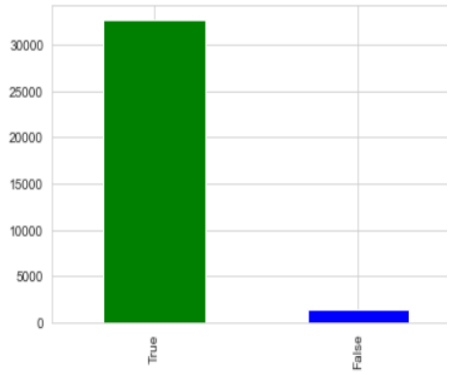

Fig 6. Recommendations for a product

Correlation

In this section, we will look at the data from the null hypothesis angle and see how much we learn from the data. We look at the rating attribute and the hopefulness number in the Amazon data set for our sentimental analysis. The below figure shows the helpfulness of the reviews vs the rating. It clearly shows a large bias towards a higher rating.

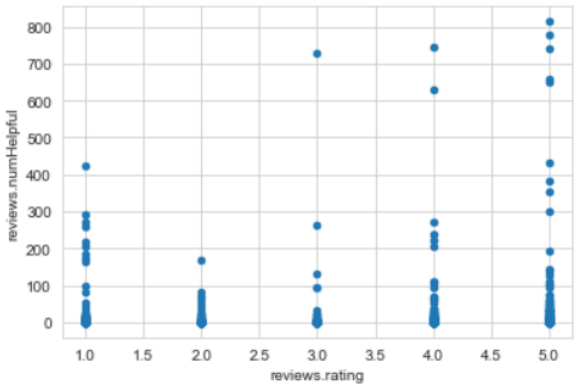

Fig 7. Rating vs HelpfulNumbers

Pearson's correlation coefficient is a statistical test that measures the relationship between two continuous variables. It is known as the best method of measuring the association between variables of interest because it is based on the method of covariance. It also gives information about the magnitude of the correlation and the direction of the relationship.

Formula

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}}$$

r = correlation coefficient
x_i = values of the x-variable in a sample
$\bar{x}$ = mean of the values of the x-variable
y_i = values of the y-variable in a sample
$\bar{y}$ = mean of the values of the y-variable

```
pearsonr(cleanreviews['reviews.rating'], cleanreviews['reviews.numHelpful'])
```

```
(-0.04239535375797332, 4.666158383728314e-15)
```

Fig 8. Pearson's correlation Results

We know that the Pearson correlation coefficient can be used to summarize the strength of the linear relationship between two variables. After running our tests, we found that the p-value is < 0.05, hence the correlation coefficient is statistically significant.

Pearson's r value is almost equal to 0 hence there is no linear relationship between reviews rating and reviews.numHelpful.

Word Cloud

Word cloud is a visually representation of text and can be used in various ways. Generally, this can be made with pure text summarization. Word clouds are mainly used in text analytics and are mainly made with a body of text. Word clouds help to know that how much similar an information is for specific research. They give us a summary of isolated words without knowing their linguistic meaning or relations. They are statistically used and provide no or limited interaction capabilities. Fig. 9 shows the list of words from features. To extract this, we have used Countvectorizer to convert text to numerical data. Countvectorizer is used to transform a given text into a vector based on the frequency (count) of each word that occurs in the entire text. This is helpful when we have multiple texts, and we wish to convert each word in each text into vectors. We have then printed feature names selected (terms selected) from the raw documents and visually represented in the form of word cloud.

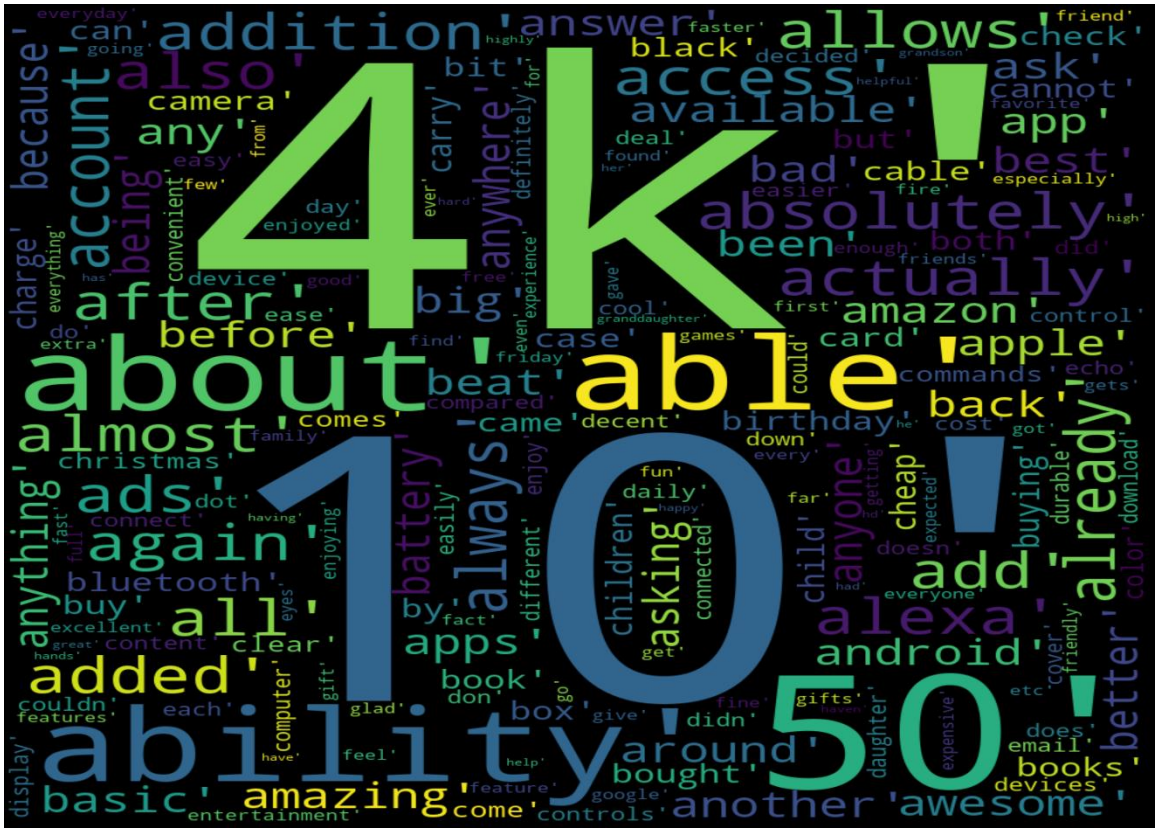

Fig.9 Word Cloud

Graphs and Plots

From further analysis below, we can see that the first 18 ASINs show that consumers recommend the product and have good ratings between 4.0 to 5.0. The remaining ASINs have fluctuating results due to lower sample size. We can also say that there are many ASINs with low occurrence that have high variances. Similarly in our correlation analysis between ASINs and reviews. rating, we see that there is almost no correlation.

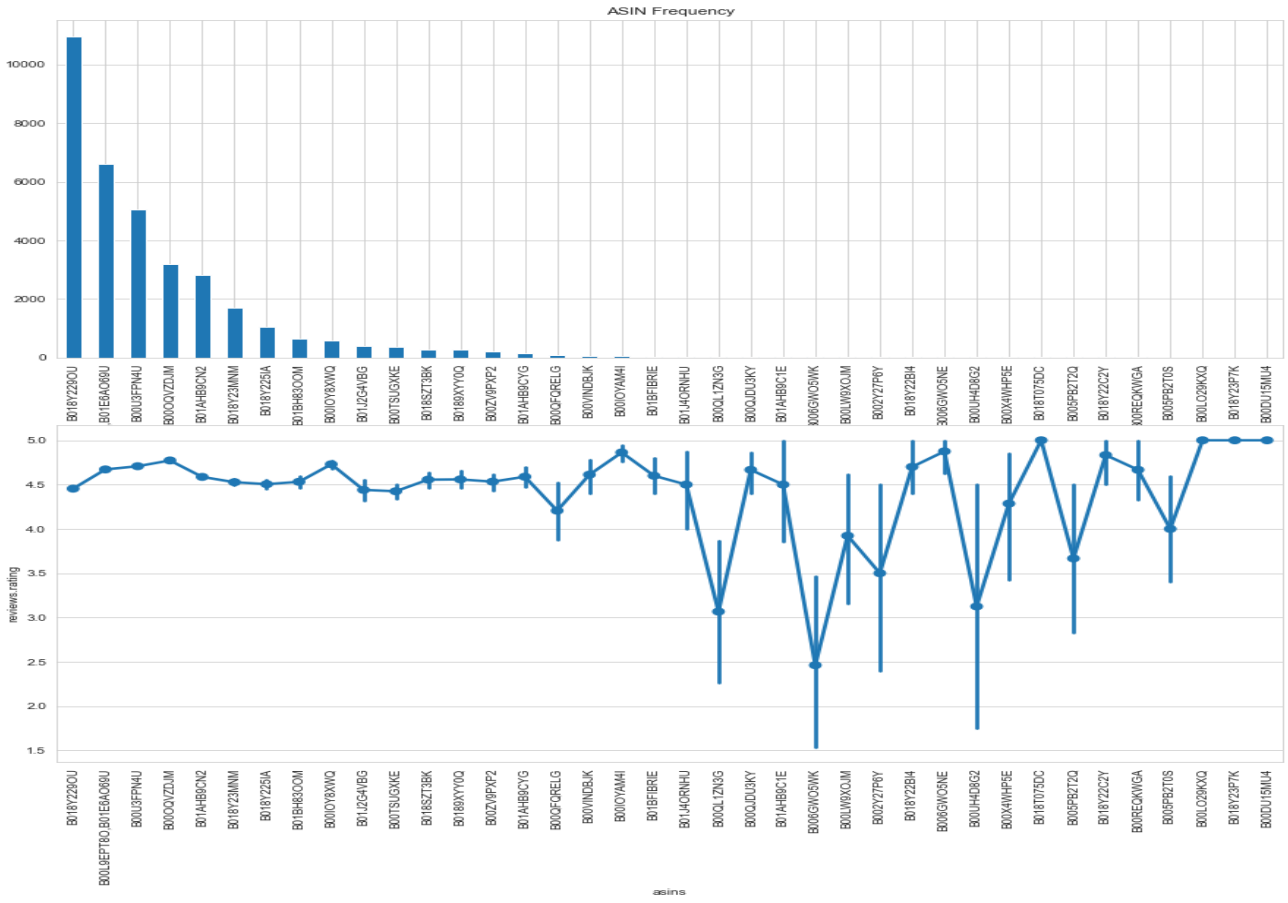

Fig.10 ASINS and Ratings

	asins	reviews.rating
B018Y229OU	10966	4.454222
B00L9EPT8O,B01E6AO69U	6608	4.670702
B00U3FPN4U	5056	4.707278
B00OQVZDJM	3176	4.772355
B01AHB9CN2	2814	4.586709
B018Y23MNM	1685	4.527003
B018Y225IA	1038	4.504817
B01BH83OOM	636	4.531447
B00IOY8XWQ	580	4.729310
B00TSUGXKE	372	4.424731
B018SZT3BK	269	4.553903
B0189XYY0Q	256	4.558594
B00ZV9PXP2	212	4.533019
B01AHB9CYG	158	4.588608
B00VINDBJK	67	4.611940
B00QFQRELG	66	4.121212
B00IOYAM4I	51	4.862745
B01BFIBRIE	30	4.600000

Fig.11 ASINS and Avg Ratings

	asins	reviews.rating
asins	1.000000	0.145934
reviews.rating	0.145934	1.000000

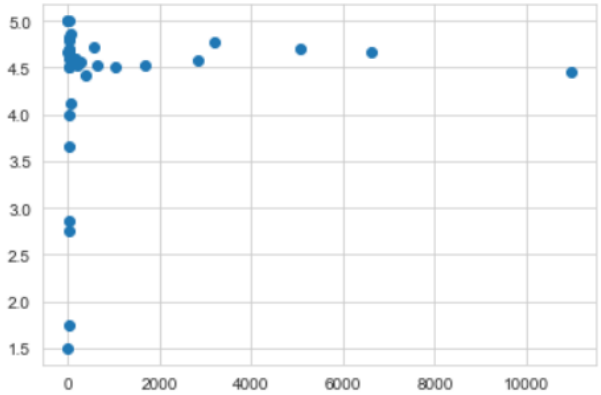

Fig.12 Ratings Scatter Plot

Dunning Log-Likelihood

The statistical model used to analyze the distinctiveness of words is called Dunning Log-Likelihood, and it calculates how distinct a word is in its in one corpus compared to another corpus. It is also a very effective way to analyze the word usage as it returns words that are not only distinctive by sheer amount but show a clear significantly distinctive. This means that words with relatively low frequencies can still receive high dunning scores when compared to the rest of the words in their respective corpus.

$$\text{Dunning's log} - \text{likelihood}\ (G^2) = O_i * \ln\left(\frac{O_i}{E_i}\right)$$

This formula that incorporates both absolute magnitude (O is the observed frequency) and a ra tio (O/E is the observed frequency divided by the frequency you would expect). Below figure Fig. 8 represents the list of positive and negative words. For example, no one mentions “issue ” or “cheap” in a positive review. I have also extracted a list of extreme features and represented it visually in the form of word cloud. We can see that, words such as amazon, mu sic, get, don and purchase are more frequently.

```
dunnings[-20:]
```

myself	4.737546
who	4.764590
thought	4.976360
loved	5.085439
lighting	5.210931
item	5.455174
recommend	5.890610
doesn	6.082314
highly	6.106454
actually	6.190255
tv	6.229765
light	6.332072
book	6.617679
feel	7.200803
perfect	8.190672
she	9.652200
read	10.361338
kindle	14.289723
paperwhite	15.432093
reader	15.539586

dtype: float64

```
dunnings[0:20]
```

issue	-13.891628
however	-10.550994
tablet	-10.094089
times	-9.337084
not	-8.907913
that	-7.447662
only	-7.231530
them	-6.307064
work	-6.129944
using	-5.964486
he	-5.607341
grandson	-5.452842
or	-5.308994
cheap	-4.921725
screen	-4.912297
helpful	-4.862142
ipad	-4.638568
again	-4.431907
will	-4.287019
easily	-4.248007

dtype: float64

Fig.13 Dunning’s Results List

Fig.14 Extreme Features Word Cloud for our data

Logistic Regression

Logistic regression is a supervised machine learning classifier that extracts real-valued features from the input, multiplies each by a weight, sums them, and passes the sum through a sigmoid function to generate a probability. A threshold is used to make a decision. Logistic regression can be used with two classes (e.g., positive, and negative review) or with multiple classes (multinomial logistic regression for example for n-ary text classification, part-of-speech labeling, etc.). Logistic regression is also one of the most useful analytic tools, because of its ability to transparently study the importance of individual features.

A logistic regression produces a logistic curve, which is limited to values between 0 and 1. Logistic regression is like linear regression, but the curve is constructed using the natural logarithm of the "odds" of the target variable, rather than the probability. Moreover, the predictors do not have to be normally distributed or have equal variance in each group.

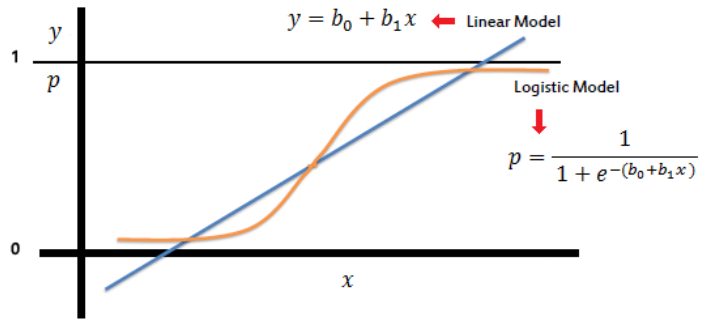

Fig.15 Logistic Regression

Evaluation Parameters

In this work, the metrics used to test the performance of machine learning classifier are accuracy, precision, recall, and F1-score. Precision measures the percentage of positive reviews that predict truly divided by the total number of reviews that are classified positively as defined by equation 1

PR = tp/ (tp + fp) ------- (1)

Recall on the other hand measures the percentage of the reviews that classify positively divided by the total number of reviews which are truly positive, as in equation 2

RC = tp/ (tp + fn) --------(2)

F1-score combines both precision and recall as in equation 3

F1-score = 2 ∗ (P R ∗ RC/ PR + RC) -------(3)

Lastly, accuracy is defined as the percentage of reviews that are classified correctly divided by the total number of reviews, equation 4.

Where tp, tn, fp, and fn are true positives, true negatives, false positives, and false negatives respectively.

ACC = (tp + tn)/ (tp + tn + fp + fn) ---------(4)

Random Forest Classifier

Random forest classifiers fall under the broad umbrella of ensemble-based learning methods. They are simple to implement, fast in operation, and have proven to be extremely successful in a variety of domains. The key principle underlying the random forest approach comprises the construction of many "simple" decision trees in the training stage and the majority vote (mode) across them in the classification stage. Among other benefits, this voting strategy has the effect of correcting for the undesirable property of decision trees to overfit training data. In the training stage, random forests apply the general technique known as bagging to individual trees in the ensemble.

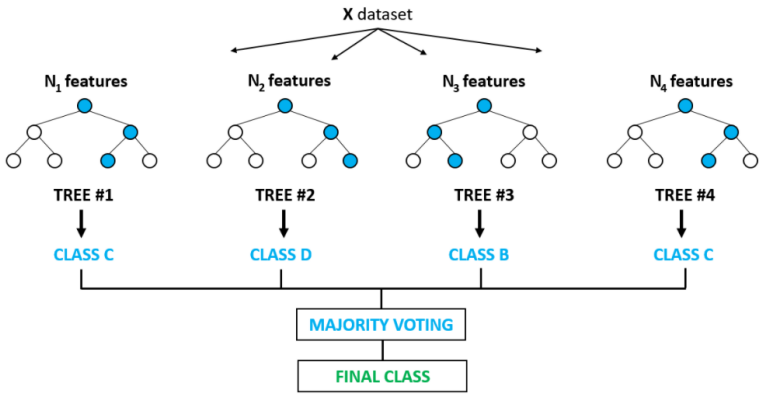

Fig.16 Random Forest classifier

Ridge Regression

Ridge regression is a model tuning method that is used to analyze any data that suffers from multicollinearity. This method performs L2 regularization. When the issue of multicollinearity occurs, least-squares are unbiased, and variances are large, this results in predicted values being far away from the actual values. The cost function for ridge regression:

$$\text{Min}(||Y - X(\text{theta})||^2 + \lambda||\text{theta}||^2)$$

Lambda is the penalty term. λ given here is denoted by an alpha parameter in the ridge function. So, by changing the values of alpha, we are controlling the penalty term. The higher the values of alpha, the bigger is the penalty and therefore the magnitude of coefficients is reduced.

- It shrinks the parameters. Therefore, it is used to prevent multicollinearity
- It reduces the model complexity by coefficient shrinkage

Results

The dataset was divided into a training set of size 28629 and a test set of size 5000. Four different modelling techniques were implemented including Dunning Log-Likelihood, Logistic Regression, Random Forest classifier and Ridge Regression. While Dunning Log-Likelihood is helpful in analyzing the most frequently used words in reviews, we are using the other models to predict the accuracy of the model. The accuracy for the models Random Forest classifier and Ridge Regression is 96.22% which is higher than the accuracy of the model Logistic Regression (96.14). We have also calculated the K Fold Cross Validation of Logistic

Regression to find out the mean and the Standard deviation to communicate how good our model is when it tries to predict new data.

Further, we have created a confusion matrix to describe the performance of a classification model. A confusion matrix, also known as an error matrix, is a performance measurement for assessing classification models. The results based on various metrics are:

True Positive: *4807* outcomes where the model correctly predicts the positive class.
True Negative: *Zero* outcome where the model correctly predicts the negative class.
False Positive: *189* outcomes where the model incorrectly predicts the positive class.
False Negative: *4* outcomes where the model incorrectly predicts the negative class.

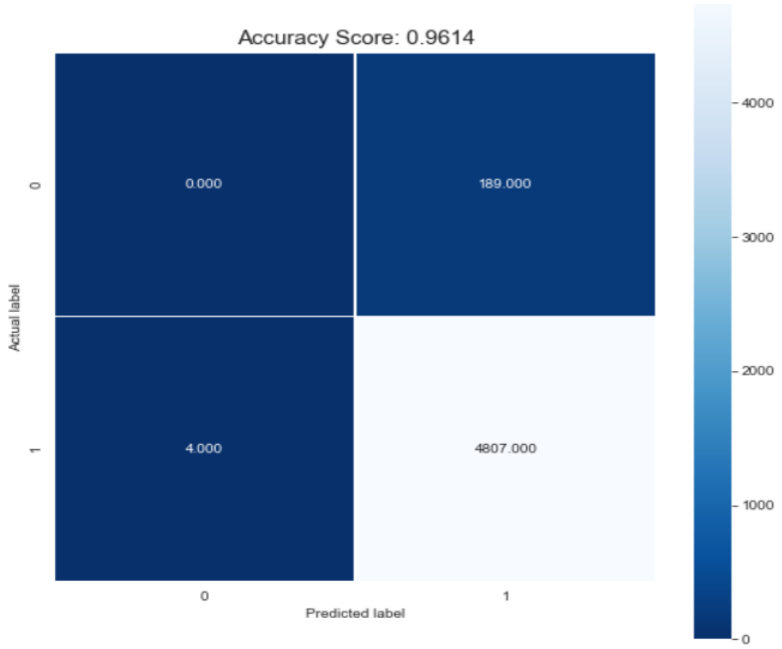

Fig.17 Confusion Matrix

Conclusion

In summary, four types of features were tried. For these four types of features, different algorithms were used including Dunning Log-Likelihood, Logistic Regression, Random Forest classifier and Ridge Regression. We can see that our accuracy is the best when we use Random Forest classifier and Ridge Regression. The accuracy is slightly high because of imbalanced classes. To identify the products that were not recommended and to fix the data imbalance issue we need to measure the F1 score. To measure this, I have cross validated the model to find an ideal c-parameter and calculated the F1 score. The F1 measure is 0.75 which is a much more reliable way to evaluate review data since the classes are imbalanced.

References

Statistics Solutions. (2021, June 9). *Pearson's Correlation Coefficient*. https://www.statisticssolutions.com/free-resources/directory-of-statistical-analyses/pearsons-correlation-coefficient/

Boost Labs. (2020, November 3). *What are Word Clouds? The Value of Simple Visualizations*. https://boostlabs.com/blog/what-are-word-clouds-value-simple-visualizations/

GeeksforGeeks. (2020, July 17). *Using CountVectorizer to Extracting Features from Text*. https://www.geeksforgeeks.org/using-countvectorizer-to-extracting-features-from-text/

V. A. (2014, May 9). *Identifying diction that characterizes an author or genre: why Dunning's may not be the best method.* The Stone and the Shell. https://tedunderwood.com/2011/11/09/identifying-the-terms-that-characterize-an-author-or-genre-why-dunnings-may-not-be-the-best-method/

Aljuhani, S. A., & Saleh, N. (2019). A Comparison of Sentiment Analysis Methods on Amazon Reviews of Mobile Phones. *International Journal of Advanced Computer Science and Applications*, *10*(6). https://doi.org/10.14569/ijacsa.2019.0100678

Tan, W., Wang, X., & Xu, X. (n.d.). *Sentiment Analysis for Amazon Reviews*.

Elmurngi, E. I., & Gherbi, A. (2018). Unfair Reviews Detection on Amazon Reviews using Sentiment Analysis with Supervised Learning Techniques. *Journal of Computer Science*, *14*(5), 714–726. https://doi.org/10.3844/jcssp.2018.714.726

Rashid, A., & Huang, C. Y. (2021). Sentiment Analysis on Consumer Reviews of Amazon Products. *International Journal of Computer Theory and Engineering*, *13*(2), 35–41. https://doi.org/10.7763/ijcte.2021.v13.1287

Amazon Reviews using Sentiment Analysis. (2019, June 10). Mickzhang. https://mickzhang.com/amazon-reviews-using-sentiment-analysis/

Richardson, W. W. L. (2019, November 27). *Random Forests*. Kevin. https://kevintshoemaker.github.io/NRES-746/RandomForests.html

Great Learning Team. (2021, December 7). *What is Ridge Regression?* GreatLearning Blog: Free Resources What Matters to Shape Your Career! https://www.mygreatlearning.com/blog/what-is-ridge-regression/

Shin, T. (2021, December 13). *How to Evaluate Your Machine Learning Models with Python Code!* Medium. https://towardsdatascience.com/how-to-evaluate-your-machine-learning-models-with-python-code-5f8d2d8d945b

Shin, T. (2021b, December 13). *How to Evaluate Your Machine Learning Models with Python Code!* Medium. https://towardsdatascience.com/how-to-evaluate-your-machine-learning-models-with-python-code-5f8d2d8d945b