

INSTITUTE FOR ADVANCED COMPUTING AND SOFTWARE DEVELOPMENT AKRUDI, PUNE

DOCUMENTATION ON

**“Amazon Product Reviews Sentiment Analysis”**

PG-DBDA Feb-2020

**Submited by:**

Group No. :09

Sayali Anil Mahajan(1327)

Snehal Sanjay Warade(1358)

Mr. Prashant Karhale

Mr.Akshay Tilekar

**Centre Coordinator**

**Project Guide**

**INDEX**

**Table of Contents**

1. Abstract…………………………………………………………………………5
2. Introduction……………………………………………………………………..6

2.1 What is Sentiment Analysis? ………………………………………….6

2.2 Purpose…………………………………………………………………6

2.3 Scope of the Project……………………………………………………6

2.3.1 Initial functional requirement………………………………….6

2.3.2 Initial nonfunctional requirement will be……………………...6

1. Review of Literature…………………………………………………………...8
2. Objectives of the Review……………………………………………………….9
3. System Design…………………………………………………………………10

5.1 Hardware Requirements……………………………………………...10

5.2 Software Requirements………………………………………………10

* 1. Libraries……………………………………………………………...10

1. Methodology For Implementation(Formula/Algorithm)………………………11

6.1 Data Collection………………………………………………………..11

6.1.1 Types Of data…………………………………………………..11

6.1.2 Methods of collecting data……………………………………..11

6.2 Data Exploration and Data Visualization……………………………...12

6.2.1 Data Exploration……………………………………………….12

6.2.2 Data Visualization……………………………………………...12

6.3 Data Cleaning………………………………………………………….12

6.4 Data Preprocessing…………………………………………………….13

6.5 Naïve Bayes……………………………………………………………14

6.6 Logistic Regression……………………………………………………14

6.7 Decision Tree………………………………………………………….15

6.8 Support Vector………………………………………………………...15

6.9 Random Forest………………………………………………………...16

1. Implementation Details…………………………………………………………..18
2. Results and Sample Output………………………………………………………33
3. Conclusion………………………………………………………………………..35
4. Future Scope and References…………………………………………………….36

**List of Figures**

Fig 1. Sentiment Analysis Model…………………………………………………………..9

Fig 2. Data Visualization…………………………………………………………………...12

Fig 3. Data Cleaning………………………………………………………………………..13

Fig 4. Data Frame Work (Preprocessing)…………………………………………………..13

Fig 5. Logistic Regression………………………………………..................................15

Fig 7. Support Vector Machine………………………………………………………….16

Fig 8. Work Flow of Random Forest Classifier………………………………………...17

Fig 9. Example Using Random Forest………………………………………………….17

**List of Tables**

Table 1 Terms and Definitions………………………………………………………7

**Chapter 1**

**ABSTRACT**

Every one of us have a different perspective and view towards online marketing. The review and its analysis speaks some thoughts. sentimental analysis is another term in reading the data efficiently from the customers point of view which is cost effective too. from user rating, suggestions a large amount of data is generated over time. By Gathering necessary information about products and their reviews we can classify them on basis of review level categorization. The model can focus over the overall semantic(positive or negative) depending on the reviews through consumers. Study involving analyzation of this semantic data will prove highly beneficial to marketers in order to grasp the preferences and suggestions and improve their services. The prediction model can further execute and undergo through certain machine learning algorithms, regression ,visualized views and methodologies in predicting and analyzing amazon ecommerce products reviews.

**Chapter 2**

**Introduction**

**2.1What is Sentiment Analysis?**

Sentiment Analysis is contextual mining of text which identifies and extracts subjective information in source material, and helping a business to understand the social sentiment of their brand, product or service while monitoring online conversations. By using sentiment analysis,we extract the features of over product.

**2.2**.**Purpose**

The goal of this project is to focus over the overall semantic(positive or negative) depending on the reviews through consumers. Study involving analyzation of this semantic data will prove highly beneficial to marketers in order to grasp the preferences and suggestions and improve their services. The prediction model can further execute and undergo through certain machine learning algorithms,visualized views and methodologies in predicting and analyzing amazon ecommerce products reviews.

**2.3. Scope of the project**

**2.3.1 Initial functional requirement will be:-**

* Selecting the algorithm meeting requirement.
* Choosing the optimum algorithm form set of algorithm.
* Testing it on the datasets.
* After getting the result if the result is low change the hyperparameters.
* Out of all result get best of all.

**2.3.2 Initial nonfunctional requirement will be:-**

* Getting the large datasets which can provide developer enough product reviews to train the model.
* Maintain the minimum variance& bias so the model is successfully work.
* Avoid the underfitting and overfitting.

|  |  |  |  |
| --- | --- | --- | --- |
| **Terms** | **Definitions** |  |  |
| Dataset | Data for training and testing for the model |  |  |
| Variance | Difference between the training and testing accuracy |  |  |
| Bias | Both learning and training accuracy is low |  |  |
| Overfitting | Model is very complex |  |  |
|  |  |
| Underfitting | Model is bias |  |  |
| Developer | Who is developing the model |  |  |
| Review | A written recommendation about the appropriateness of an Product for |  |  |
|  | selling and buying may include suggestions for improvement. |  |  |
| Reviewer | A person that examines an Product and has the ability to recommend |  |  |
|  | approval Product for buying or to request that changes be made in the |  |  |
|  | Product. |  |  |
| Software | A document that completely describes all of the functions of a proposed |  |  |
| Requirement | system and the constraints under which it must operate. For example, |  |  |
| Specification | this document |  |  |
| User | Reviewer |  |  |

**Table 1: Terms and Definitions.**

**Chapter 3**

**Review of Literature**

The most fundamental problem in sentiment analysis is the sentiment polarity categorization, by considering a dataset containing over 5.1 million product reviews from Amazon.com with the products belonging to four categories.A max-entropy POS tagger is used in order to classify the words of the sentence, an additional python program to speed up the process. The negation words like no, not, and more are included in the adverbs whereas Negation of Adjective and Negation of Verb are specially used to identify the phrases.The following are the various classification models which are selected for categorization: Naïve Bayesian, Random Forest, Logistic Regression and Support Vector Machine. For feature selection, Pang and Lee suggested to remove objective sentences by extracting subjective ones. They proposed a text-categorization technique that is able to identify subjective content using minimum cut. Gann et al. selected 6,799 tokens based on Twitter data, where each token is assigned a sentiment score, namely TSI(Total Sentiment Index), featuring itself as a positive token or a negative token. Specifically, a TSI for a certain token is computed as:

**Chapter 4**

**Objectives of the project**

1. First collect the data from kaggle
2. Analyze and Categorize that data
3. Analyze sentiment on dataset from review level
4. Categorization or Classification of opinion sentiment into-
   * + - * Positive
         * Neutral
         * Negative

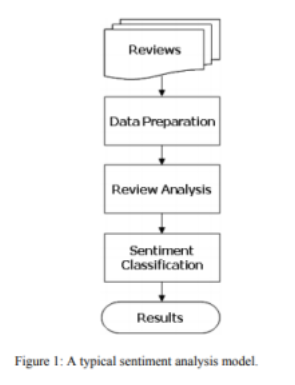


Fig 1: Sentiment Analysis Model

**Chapter 5**

**System Design**

**Hardware Requirements:-**

* + Core i5/i7 processor
  + At least 8 GB RAM
  + At least 60 GB of Usable Hard Disk Space

**Software Requirements:-**

* Python 3.x
* Anaconda Distribution
* NLTK Toolkit
* UNIX/LINUX Operating System.

**Libraries:-**

* Pandas
* Numpy
* matplotlib.pyplot

**Chapter 6**

**Methodology for Implementation**

**(Formulation/Algorithm)**

6.1 Data Collection:-

Basically, Data Collection is the process of gathering and measuring the information which we have collected through various ways. Data collection enables a person or organization to answer relevant questions, evaluate outcomes and make predictions about future probabilities and trends. So first we collect data of product reviews from kaggle. Each reviews include the following information such as id, name, asins, brand, category etc

* + 1. Types of Data

1. Quantitative Data:-

Quantitative Data is any data that is in numerical form – for e.g., percentage

1. Qualitative Data:-

Qualitative Data is a descriptive data – e.g., quality, color

* + 1. Methods of collecting data:-

1. Interviews
2. Observations
3. Documents and Records
4. Surveys
5. Group Discussion etc…
   1. Data Exploration and Data Visualization:-

6.2.1 Data Exploration

Data Exploration is the initial step in data analysis, where user explore a large data set in an unstructured way to uncover initial patterns, characteristics.

Data exploration can use a combination of manual methods and automated tools such as data visualization, charts etc…

6.2.2 Data Visualization:-

Data Visualizations is the process of translating large data sets and metrics into charts, graphs and other visuals.

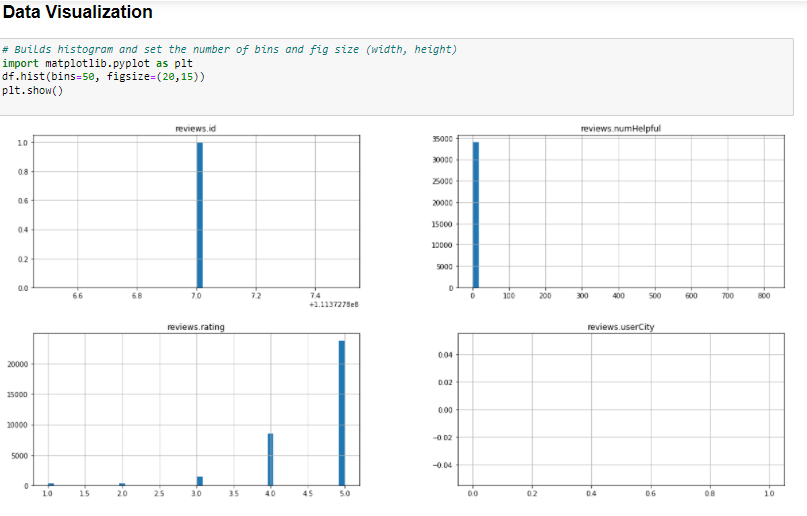


Fig 2:- Data Visualization

* 1. Data Cleaning:-

Data cleaning is the process of detecting and correcting duplicate or inaccurate records from a record set, table or database. The main aim of Data Cleaning is to remove that errors and duplicate records.

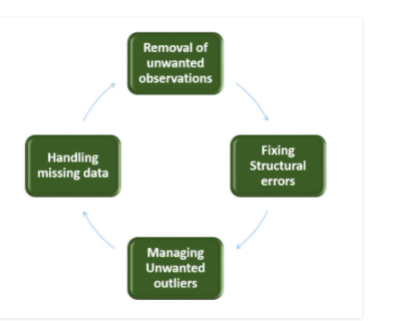


Fig 3: Data Cleaning

* 1. Data Pre-Processing:-

Data Pre-Processing is the process of preparing the raw data and making it suitable for our model

Why do we need Data Preprocessing?

A real-world data generally contains noises,missing values, and maybe in an unusable format which can not be directly used for models. So, data preprocessing is required tasks for cleaning data and making it suitable for our model.

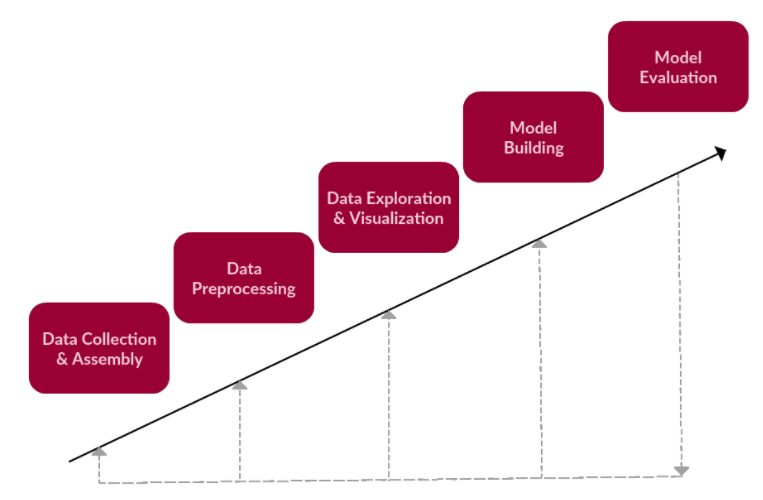


Fig 4: Data Frame Work

* 1. Naïve Bayes:-

Naïve Bayes algorithm is a supervised learning algorithm, which is based on Bayes theorem and used for solving classification problems. It is mainly used in text classification that includes a high-dimensional training dataset.

Naïve Bayes Classifier is one of the simple and most effective classification algorithms which helps in building the fast machine learning models that can make quick prediction. It is a probability classifier, which means it predicts on the basis of the probability of an object. Some popular examples of Naïve Bayes Algorithm are Sentimental analysis etc..

The Naïve bayes algorithm is comprised of two words Naïve and Bayes, which can be described as:

* Naïve:-

It is called Naïve because it assumes that the occurrence of a certain feature is independent of the occurrence of other feature.

* Bayes:-

It is called Bayes because it depends on the principle of Bayes’ Theorem

* What is Bayes theorem?

Bayes’ theorem is also knows as Bayes’ Rule or Bayes’ law, which is used to determine the probability of a hypothesis with prior knowledge. It depends on the conditional probability.

* 1. Logistic Regression:-

Logistic regression is a supervised learning classification algorithm used to predict the probability of a target variable. It is one of the simplest ML algorithms that can be used for various classification problems. Logistic regression predicts the probability of an outcome that can only have two values (i.e. a dichotomy). The prediction is based on the use of one or several predictors (numerical and categorical). A linear regression is not appropriate for predicting the value of a binary variable for two reasons:

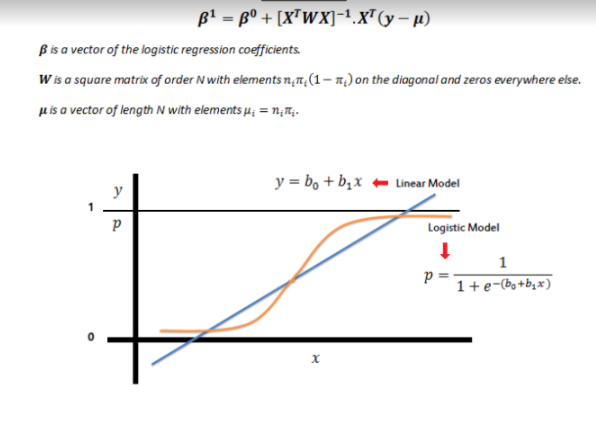


Fig 5: Logistic Regression

6.7 Decision Tree:-

Decision Trees are a type of Supervised Machine Learning (that is you explain what the input is and what the corresponding output is in the training data) where the data is continuously split according to a certain parameter. The tree can be explained by two entities, namely decision nodes and leaves. The leaves are the decisions or the final outcomes. And the decision nodes are where the data is split.

6.8 Support Vector Machine(SVM):-

Support Vector Machines are supervised learning models with associated learning algorithms that analyze data for classification and regression analysis.

Types of Support Vector Machine:-

* Linear SVM:-

Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and classifier is used called as Linear SVM classifier.

* Non-linear SVM:-

When we cannot separate data with a straight line we use Non-Linear SVM. In this, we have kernel functions. They transform non-linear spaces into linear spaces.

So, in our Project we use Linear SVC, it is a support vector classifier is used to fit to the data you provide, returning a “best fit” hyperplane that divides, or categorizes, your data.

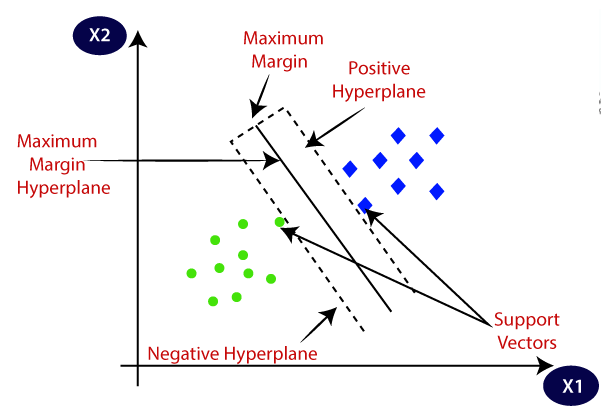


Fig 6: Support Vector Machine

* 1. Random Forest Classifier:-

Random forest is an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision tree at training time. The random forest classifier divides this dataset into subsets. These subsets are given to every decision tree in the random forest system. Each decision tree produces its specific output. So basically, decision trees are the building blocks of a random forest algorithm. A decision tree consists of three components: decision nodes, leaf nodes, and a root node. The nodes in the decision tree represent attributes that are used for predicting the outcome.

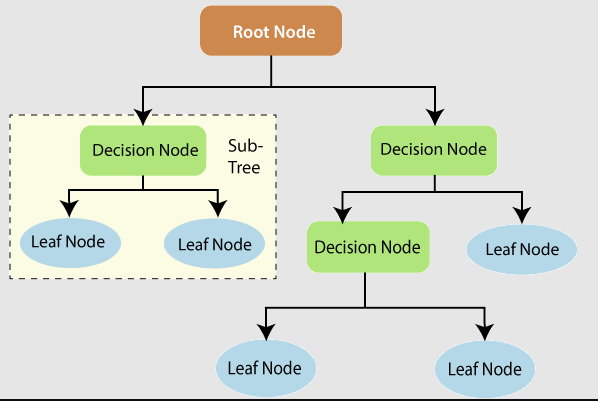


Fig 7: Work Flow Of Random Forest Classifier

For example, the prediction for trees 1 and 2 is apple.

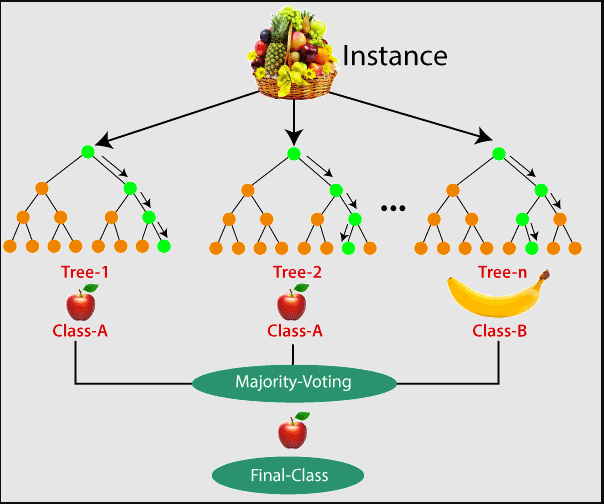


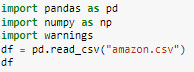
Fig 8: Example using Random Forest

**Chapter 7**

Implementation Details

7.1 Unpacking of Data:-

The huge datasets of reviews obtained from kaggle comes in a .csv file.A small python code has been implemented in order to read the datasets from those files.



7.2 Check data columns:-

Check our data consist which data columns, so it consist following columns.

‘id', 'name', 'asins', 'brand', 'categories', 'keys', 'manufacturer',

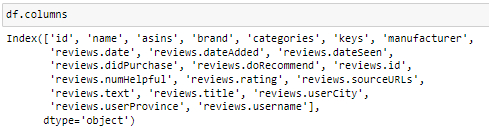
'reviews.date', 'reviews.dateAdded', 'reviews.dateSeen',

'reviews.didPurchase', 'reviews.doRecommend', 'reviews.id',

'reviews.numHelpful', 'reviews.rating', 'reviews.sourceURLs',

'reviews.text', 'reviews.title', 'reviews.userCity',

'reviews.userProvince', 'reviews.username'],



7.3 Describe the Dataset:-

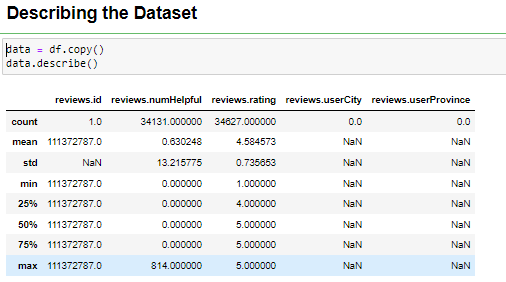
Based on the descriptive statistics above, we see the following:

* Average review score of 4.58, with low standard deviation

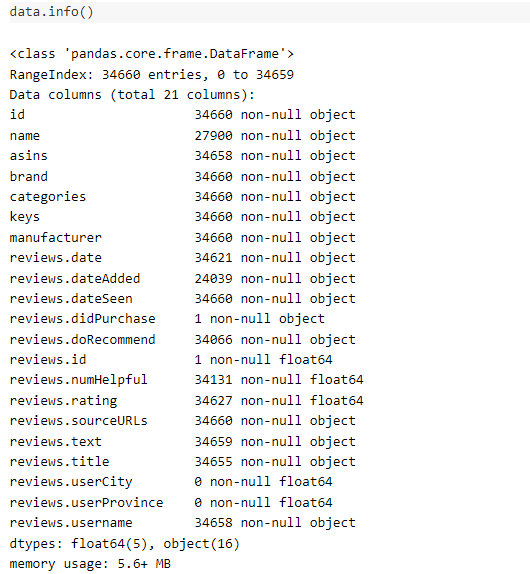
Most review are positive from 2nd quartile onwards

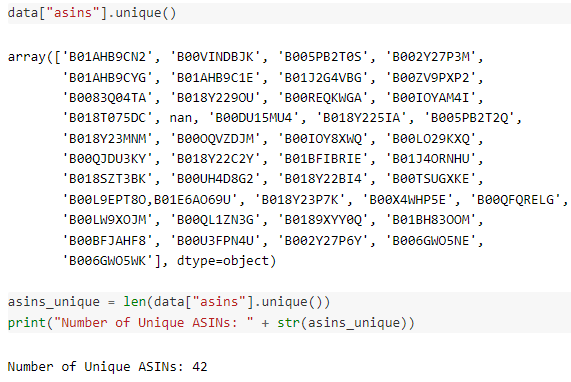
* The most helpful review was helpful to 814 people

This could be a detailed, rich review that will be worth looking at



* 1. Based on the information:-
* Drop reviews.userCity, reviews.userProvince, reviews.id, and reviews.didPurchase since these values are floats (for exploratory analysis only)
* Not every category have maximum number of values in comparison to total number of values
* reviews.text category has minimum missing data (34659/34660) -> Good news!
* We need to clean up the name column by referencing asins (unique products) since we have 7000 missing values



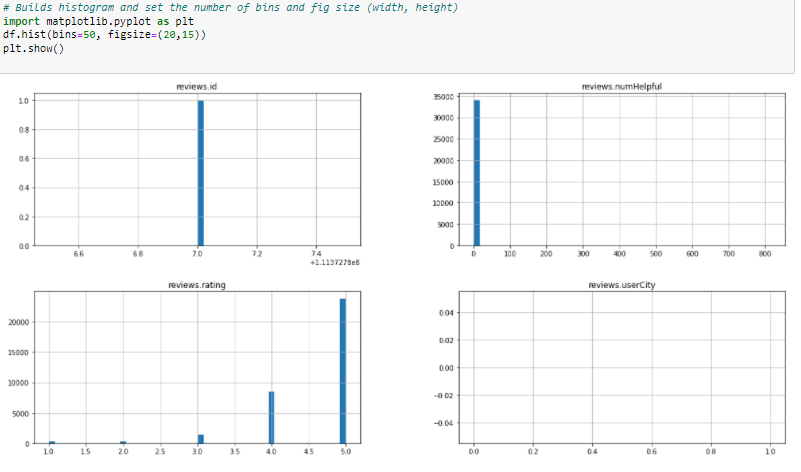


7.3 Visualize data:-

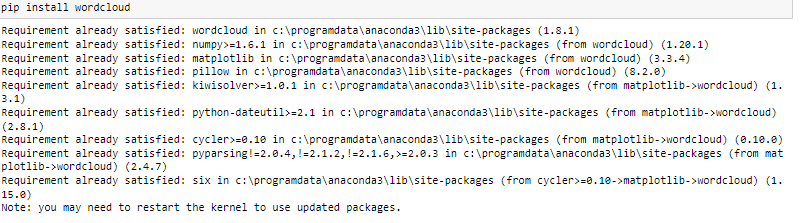
In this step, we do some visualization part on our data with the help of histogram.

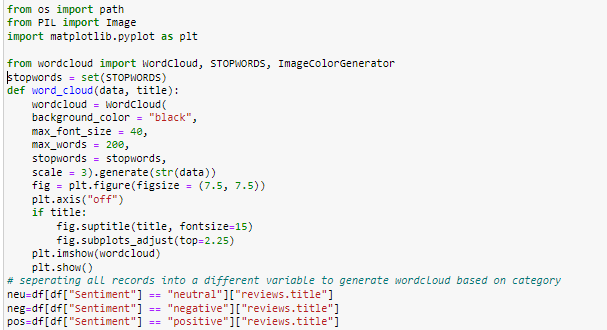
Based on the distributions:

* reviews.numHelpful: Outliers in this case are valuable, so we may want to weight reviews that had more than 50+ people who find them helpful
* reviews.rating: Majority of examples were rated highly (looking at rating distribution). There is twice amount of 5 star ratings than the others ratings combined

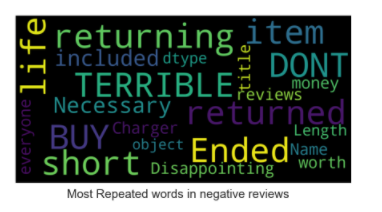


Worldcloud:-







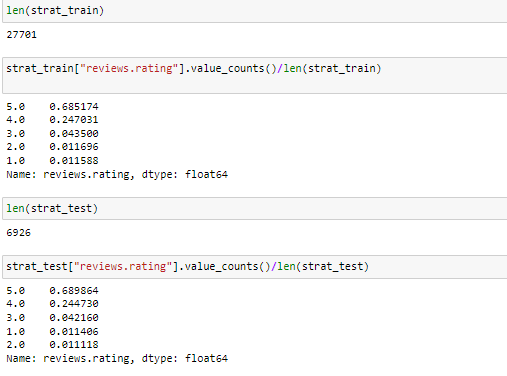


7.4 Split into Train/Test:-

* Before we explore the dataset we’re going to split it into training set and test sets
* Our goal is to eventually train a sentiment analysis classifier
* Since the majority of reviews are positive (5 stars), we will need to do a stratified split on the reviews score to ensure that we don’t train the classifier on imbalanced data
* To use sklearn’s *Stratified ShuffleSplit* class, we’re going to remove all samples that have NAN in review score, then covert all review scores to *integer* datatype.



Check to see if train/test sets were stratified proportionately in comparison to raw data.

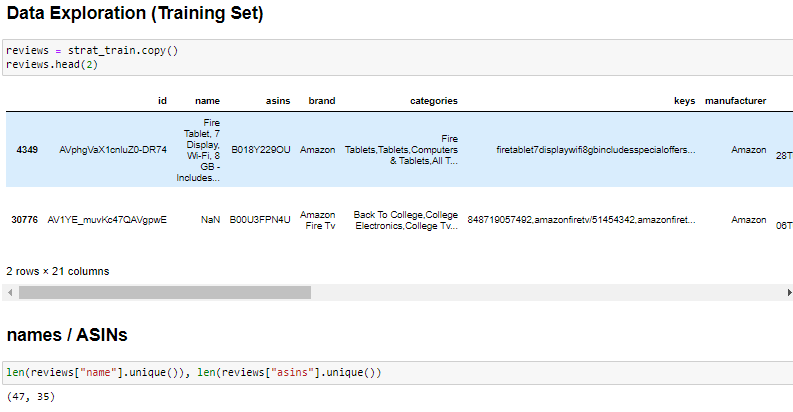


* 1. Data Exploration (Training Set):-

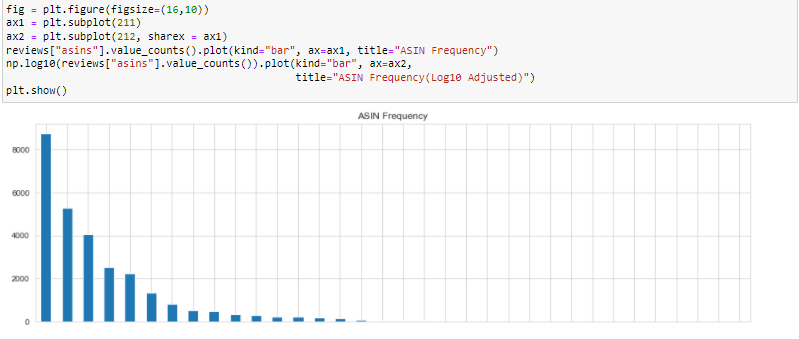
Data exploration is an approach similar to initial data analysis, whereby a data analyst uses visual exploration to understand what is in a dataset and the characteristics of the data, rather than through traditional data management system.

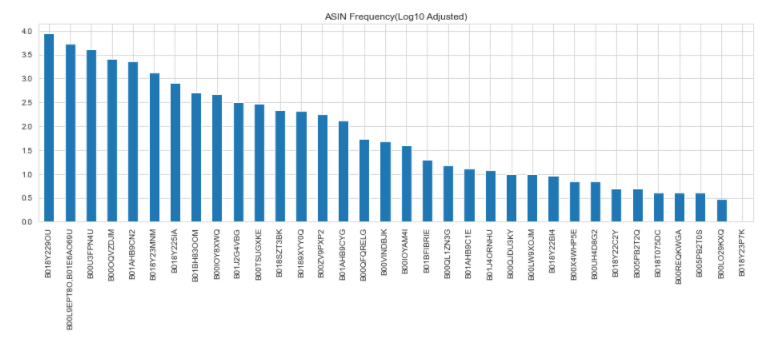
Data exploration is the first step of data analysis used to explore and visualize data to uncover insights from the start or identify areas or patterns to dig into more.

We will use regular expressions to clean out any unfavorable characters in our dataset, and then preview what the data looks like after the cleaning.



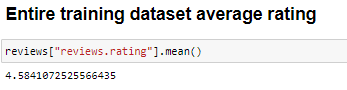
Confirmed our hypothesis that each ASIN can have multiple names. Therefore we should only really concern ourselves with which ASINs do well, not the product names.

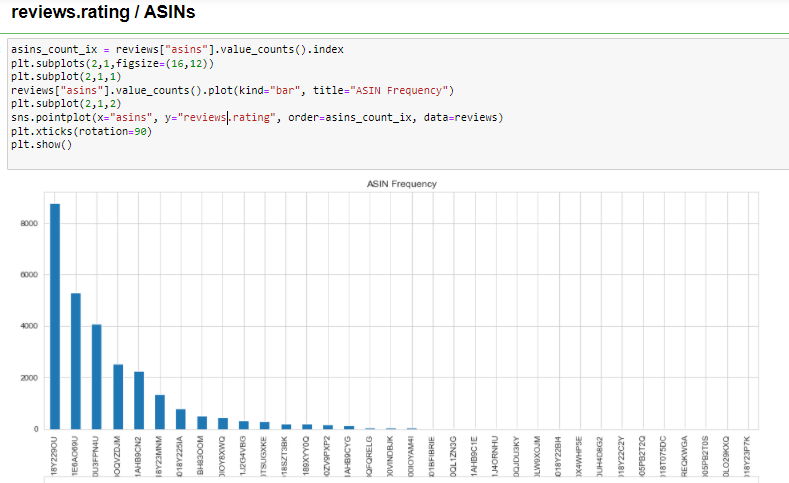


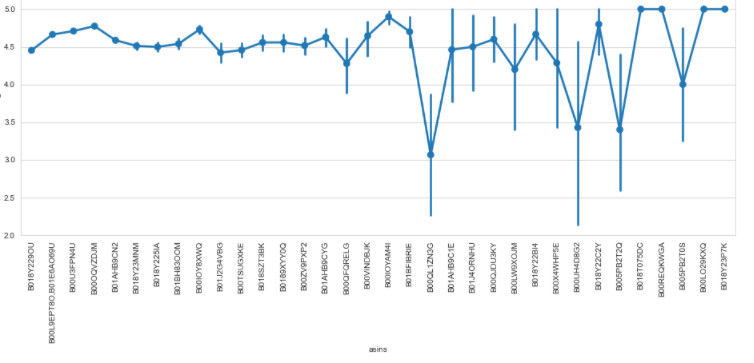


* Based on the bar graph for ASINs, we see that certain products have significantly more reviews than other products, which may indicate a higher sale in those specific products
* We also see that the ASINs have a “right tailed” distribution which can also suggest that certain products have higher sales which can correlate to the higher ASINs frequencies in the reviews

**This answers the first question that certain ASINs (products) have better sales, while other ASINs have lower sale, and in turn dictates which products should be kept or dropped.**

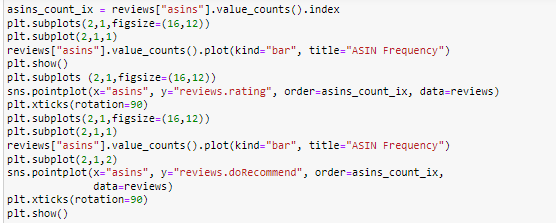
****

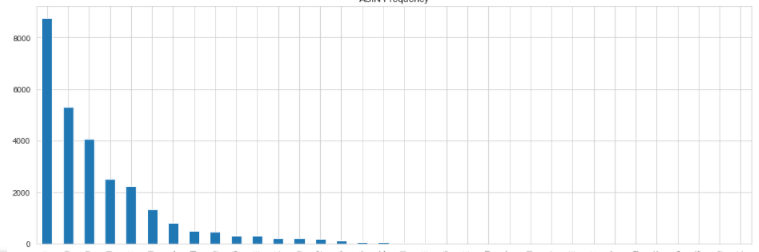




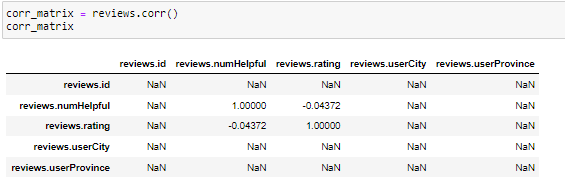
* The most frequently reviewed products have their average review ratings in the 4.5 – 4.8 range, with little variance
* Although there is a slight inverse relationship between the ASINs frequency level and average review ratings for the first 4 ASINs, this relationship is not significant since the average review for the first 4 ASINs are rated between 4.5 – 4.8, which is considered good overall reviews

Reviews.doRecommend / Asins:-





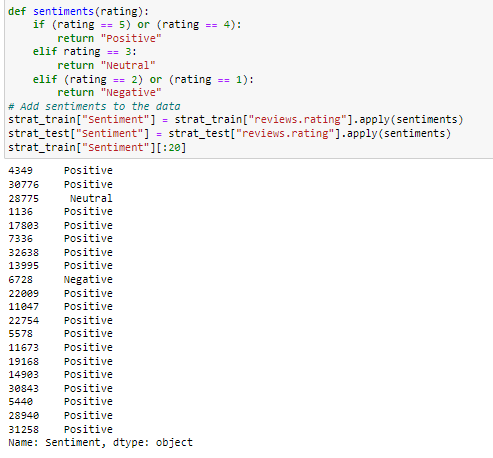
* 1. Correlations:-

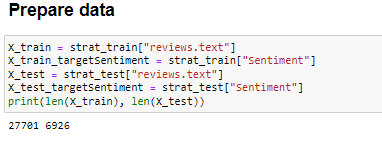


* 1. Sentiment Analysis:-

Using the features in place, we will build a classifier that can determine a review’s sentiment.

* + 1. Set Target Variables(Sentiments):-





* + 1. Extract Features:-

Here we will turn content into numerical feature vectors using the **Bag of Words**strategy:

* Assign fixed integer id to each word occurrence (integer indices to word occurrence dictionary)
* X[i,j] where i is the integer indices, j is the word occurrence, and X is an array of words (our training set)

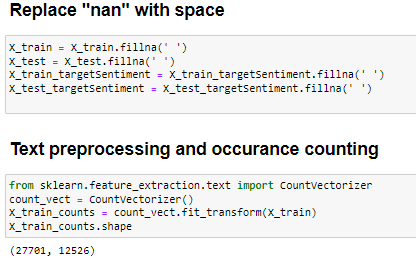
In order to implement the **Bag of Words** strategy, we will use SciKit-Learn’s **CountVectorizer**to performs the following:

* Text preprocessing:

Tokenization (breaking sentences into words)

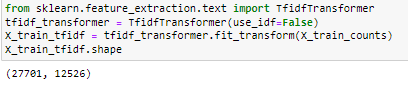
Stopwords (filtering “the”, “are”, etc)

* Occurrence counting (builds a dictionary of features from integer indices with word occurrences)
* Feature Vector (converts the dictionary of text documents into a feature vector)



Here we have 27,701 training samples and 12,526 distinct words in our training sample. Also, with longer documents, we typically see higher average count values on words that carry very little meaning, this will overshadow shorter documents that have lower average counts with same frequencies, as a result, we will use **TfidfTransformer** to reduce this redundancy:

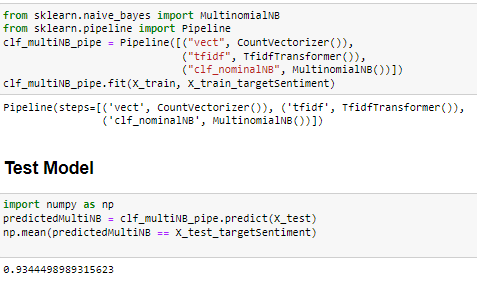
* Term Frequencies (Tf) divides number of occurrences for each word by total number of words
* Term Frequencies times Inverse Document Frequency (**Tfidf**) downscales the weights of each word (assigns less value to unimportant stop words ie. “the”, “are”, etc)



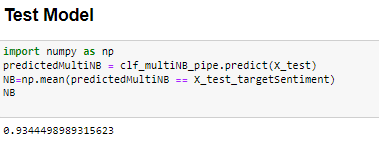
* 1. Building a pipeline from the extracted features:-

We will use **Multinominal Naive Bayes** as our Classifier

* Multinominal Niave Bayes is most suitable for word counts where data are typically represented as word vector counts (number of times outcome number X[i,j] is observed over the n trials), while also ignoring non-occurrences of a feature i
* Naive Bayes is a simplified version of Bayes Theorem, where all features are assumed conditioned independent to each other (the classifiers), P(x|y) where x is the feature and y is the classifier

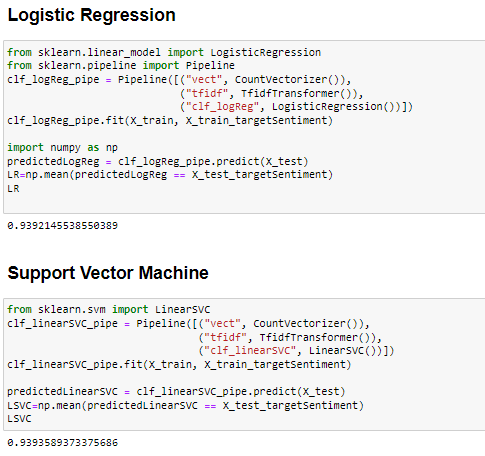


* 1. Test Model:-

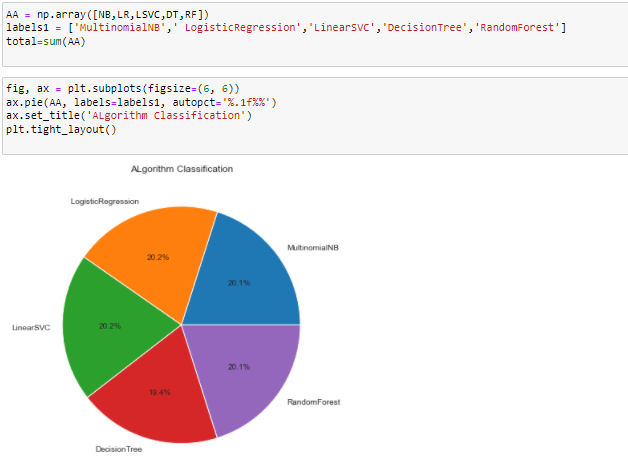


Here we see that our Multinominal Naive Bayes Classifier has a 93.45% accuracy level based on the features.

* 1. Testing Other Models:-
* Test other models
* Fine tune the best models to avoid over-fitting



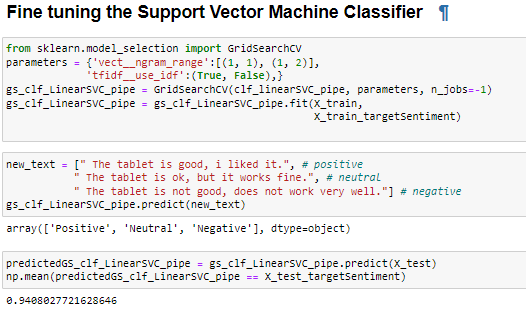




Looks like all the models performed very well (>90%), and we will use the **Support Vector Machine Classifier** since it has the highest accuracy level at **93.94%.** Now we will fine tune the Support Vector Machine model (Linear\_SVC) to avoid any potential over-fitting.

* 1. Fine Tuning The Support Vector Machine Classifier:-
* Here we will run a **Grid Search** of the best parameters on a grid of possible values, instead of tweaking the parameters of various components of the chain (ie. use\_idf in tfidftransformer)
* We will also run the grid search with LinearSVC classifier pipeline, parameters and cpu core maximization
* Then we will fit the grid search to our training data set
* Next we will use our final classifier (after fine-tuning) to test some arbitrary reviews
* Finally we will test the accuracy of our final classifier (after fine-tuning)

Note that **Support Vector Machines** is very suitable for classification by measuring extreme values between classes, to differentiate the worst case scenarios so that it can classify between Positive, Neutral and Negative correctly.

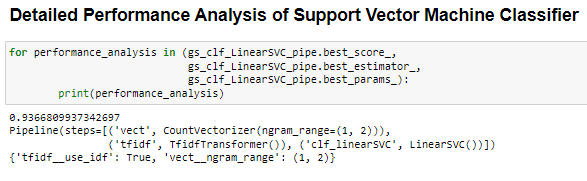


Results:-

* After testing some arbitrary reviews, it seems that our features is performing correctly with Positive, Neutral, Negative results
* We also see that after running the grid search, our Support Vector Machine Classifier has improved to **94.08%** accuracy level
  1. Detailed Performance Analysis of Support Vector Machine Classifier:-

For detailed analysis, we will:

* Analyze the best mean score of the grid search (classifier, parameters, CPU core)
* Analyze the best estimator
* Analyze the best parameter



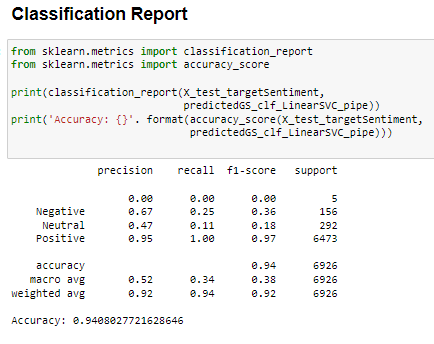
**Chapter 8**

**Results and Sample Output**

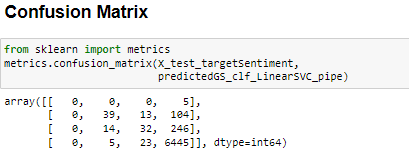
The ultimate outcome of this Training of Public reviews dataset is that, the machine is capable of judging whether an entered sentence bears positive response or negative response.

**Precision** (also called positive predictive value) is the fraction of relevant instances among the retrieved instances, while **Recall** (also known as sensitivity) is the fraction of relevant instances that have been retrieved over the total amount of relevant instances. Both precision and recall are therefore based on an understanding and measure of relevance.

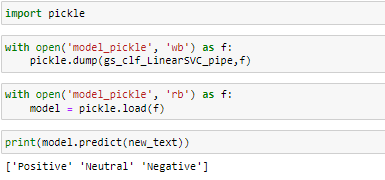
**F1 score** (also F-score or F-measure) is a measure of a test's accuracy. It considers both the precision p and the recall r of the test to compute the score: p is the number of correct positive results divided by the number of all positive results returned by the classifier, and r is the number of correct positive results divided by the number of all relevant samples (all samples that should have been identified as positive). The F1 score is the harmonic average of the precision and recall, where an F1 score reaches its best value at 1 (perfect precision and recall) and worst at 0.



Confusion Matrix:-



**Export Model:-**

****

**Chapter 9**

**Conclusion**

Sentiment analysis deals with the classification of texts based on the sentiments they

contain. This article focuses on a typical sentiment analysis model consisting of three

core steps, namely data preparation, review analysis and sentiment classification, and

describes representative techniques involved in those steps.

Sentiment analysis is an emerging research area in text mining and computational

linguistics, and has attracted considerable research attention in the past few years.

Future research shall explore sophisticated methods for opinion and product feature

extraction, as well as new classification models that can address the ordered labels

property in rating inference. Applications that utilize results from sentiment analysis

is also expected to emerge in the near future.

**Chapter 10**

**Future Scope**

The future of sentiment analysis is going to dig deeper, far past the surface of the number of likes, comments and shares, and aim to reach, and truly understand, the significance of social media interactions and what they tell us about the consumers behind the screens. This forecast also predicts broder applications for sentiment analysis.

**References**

1. <https://www.kaggle.com/datafiniti/consumer-reviews-of-amazon-products>
2. <https://github.com/avinash-vk/Sentiment-analysis-on-amazon-reviews/blob/main/datacleaning.ipynb>
3. <https://github.com/alaBay94/Sentiment-analysis-amazon-Products-Reviews/blob/master/SentimentAnalysis%20Word2Vec%20word%20embeding.ipynb>
4. <https://github.com/alaBay94/Sentiment-analysis-amazon-Products-Reviews/blob/master/SentimentAnalysis%20Word2Vec%20word%20embeding.ipynb>
5. <https://towardsdatascience.com/sentiment-analysis-concept-analysis-and-applications-6c94d6f58c17>