Artificial Intelligence Project

Sentiment Analysis for Amazon Reviews

Submitted by

Titiksha Sahai 05901032017

Payal Mohanty 04901032017

Ashwini Hassa Purty 05501032017

Under the Supervision of

Mr. Rishabh Kaushal

Assistant Professor

Department of Information Technology

Department of Information Technology

Indira Gandhi Delhi Technical University for Women

Kashmere Gate, Delhi – 110006

1. INTRODUCTION

Sentiment analysis is the technique of predicting emotions associated with the data through Natural Language Processing (NLP). It classifies the text into three basic emotions, namely, positive, negative & neutral. With the advancement in digital platforms like social media, e-commerce website, etc. Millions of people are able to share their views and opinions on the internet conveniently. This raw data generated by the customer on the internet is the source of valuable information which can be analyzed and used for various purposes like developing more insightful marketing strategies, understanding brand perceptions, finding leaders and influencers, providing better customer services etc.[1]

The objective of our analysis is to build supervised learning algorithms to classify the various product reviews shared on Amazon.com by costumers into positive or negative. Although, this analysis can also be applied on various other platforms to find polarity or sentiments of customers. This helps businesses to find consumer opinions and emotions about their products and services. Potential customers also want to know the opinions and emotions of existing users before they use a service or purchase a product. Last but not least, researchers [2] uses these information to do an in-depth analysis of market trends and consumer opinions, which could potentially lead to a better prediction of the stock market. We will use various supervised learning algorithms namely, Multinomial Naive Bayes, decision tree, logistic regression, SVM, Decision tree and random forest and compare their accuracy. We will fine tune the most accurate model to optimize its performance.

2. RELATED WORK:

There has been ample research on sentiment analysis of product reviews. Some of the research includes:

Xu Yun [3] el al from Stanford University. They applied existing supervised learning algorithms such as perceptron algorithm, naive bayes and supporting vector machine to predict a review's rating on Yelp's rating dataset. They used hold out cross validation using 70% data as the training data and 30% data as the testing data. The author used different classifiers to determine the precision and recall values.

In paper [4], Maria Soledad Elli and Yi-Fan extracted sentiment from the reviews and analyze the result to build up a business model. They claimed that this tool gave them pretty high accuracy. They mainly used Multinomial Naive Bayesian (MNB) and support vector machine as the main classifiers.

Callen Rain [5] proposed extending the current work in the field of natural language processing. Naive Bayesian and decision list classifiers were used to classify a given review as positive or negative.

Our work differs from the prior work as we used traditional supervised learning algorithms including Multinomial Naive Bayes, decision tree, logistic regression, SVM, Decision tree and random forest to understand the working of these models in sentiment analysis field with deep learning approach. We will compare their accuracy and fine tune the algorithm in order to avoid bias sentiments.

3. PROPOSED METHODOLOGY:

3.1 DATASET DESCRIPTION

We collected this data from kaggle.com. “Kaggle is a subsidiary of Google LLC that provides crowd-sourced platform to attract, nurture, train and challenge data scientists from all around the world to solve data science, machine learning and predictive analytics problems. It allows users to find and publish data sets, explore and build models in a web-based data-science environment, work with other data scientists and machine learning engineers, and enter competitions to solve data science challenges” [6]

The dataset that we used is present as 1429_1.csv file provided by Datafiniti's Product Database. It contains 34,660 consumer reviews for Amazon products (like the Kindle, tablets, clothes, Fire TV Stick etc.). It also contains 21 attributes providing basic product information (like brand, price, category etc), ratings, reviews, and more for each product.

Table 3.1.1 Details of data attributes:

ATTRIBUTE NAME	ATTRIBUTE TYPE	DESCRIPTION
Id	Non null object	It is unique and associated with each entry of dataset.
Name	Non null object	It describes the name of the product.
Asins	Non null object	Asin is acronym for Amazon Standard Identification Number. It is a unique ten character alphanumeric identifier assigned by Amazon.com for product-identification within their product catalog.
Brand	Non null object	Name of the brand which manufactured the product.

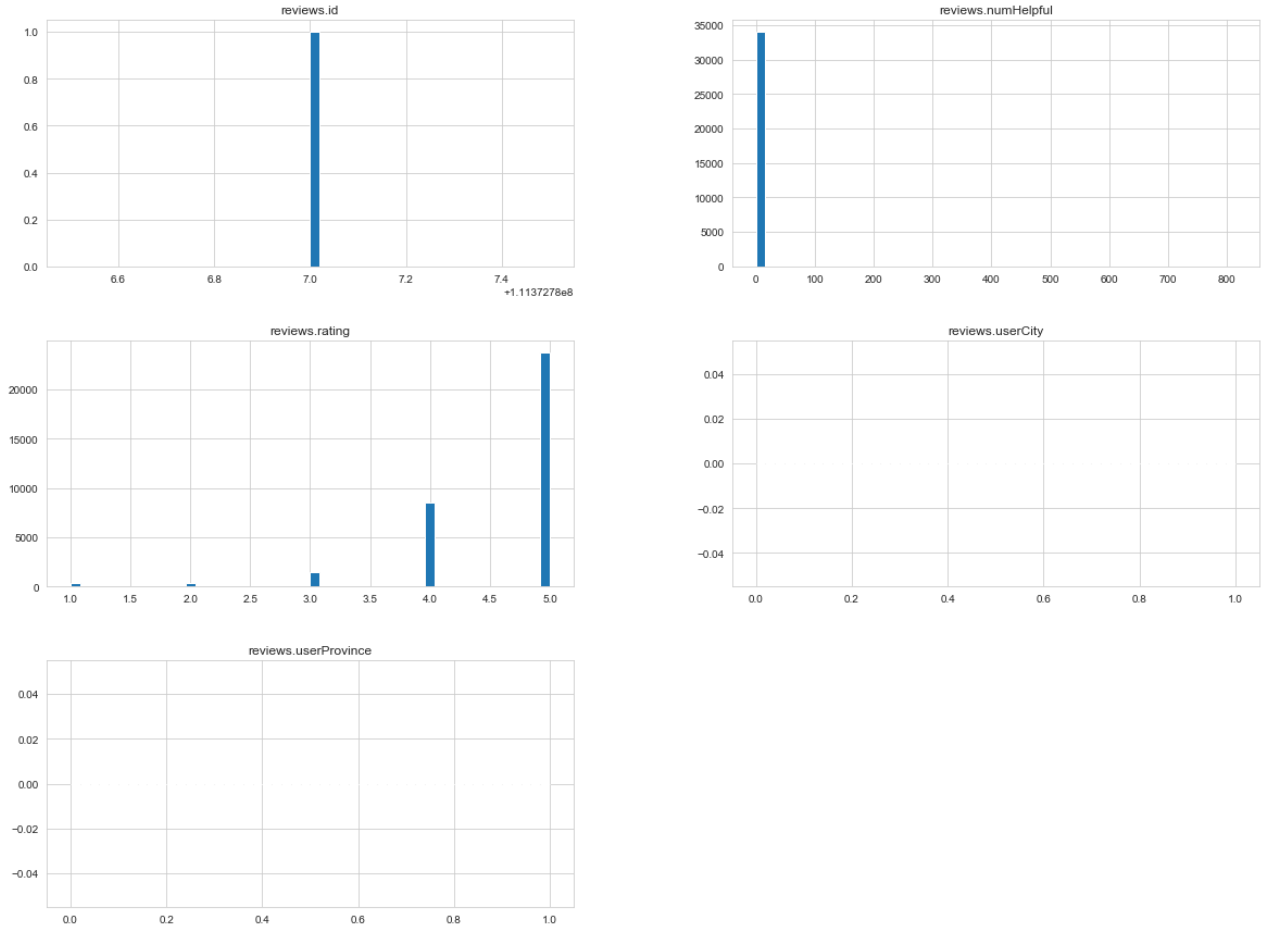
Categories	Non null object	The retail category in which the product belongs.
Keys	Non null object	It uniquely identifies the record.
Manufacturer	Non null object	It provides the name of the manufacturer of the product.
reviews.date	Non null object	It contains information regarding the review timing
reviews.dateAdded	Non null object	It contains the date at which the review was added
reviews.dateSeen	Non null object	It contains the date at which the review was seen.
reviews.didPurchase	Non null object	It verifies if the reviews is added by a customer.
reviews.doRecommend	Non null object	It is Boolean value describing if the customer recommend the product or not.
reviews.id	Non null float64	It is the unique object associated with each review.
reviews.rating	Non null float64	It contains rating of a product associated with each review (1-5)
reviews.sourceURLs	Non null object	It contains the URL source of each review in the dataset
reviews.text	Non null object	text in each product review by customer
reviews.title	Non null object	title of each review by customer

reviews.userCity	Non null float64	City name of the customer who writes the review
reviews.userProvince	Non null float64	Province name of the customer who writes the review
reviews.username	Non null object	Username of the customer who add the review

3.2 DATA EXPLORATION

3.2.1 DESTRIBUTION OF RAW DATA

Figure 3.2.1: Distribution of Numerical Values



The fig 3.2.1 shows the distribution of numerical values in our raw dataset. Few useful inferences can be deduced from this distribution. Firstly, reviews.rating shows that majority of the products in our dataset are rated highly. Secondly, outliers in reviews.numHelpful are valuable and we will focus on the reviews that are found useful by at least fifty people.

3.2.2 DATA PREPROCESSING

We checked for categories that do not have comparable number of values with respect to total number of values. We removed certain columns namely reviews.id, reviews.userCity, reviews.didPurchase and reviews.userProvince since these columns contain float values which are used for exploratory analysis only. We checked for missing values in the dataset and reviews.text category had minimum missing data (34659/34660) which is optimal for our analysis .We also found that the name column had 7000 missing values. Hence, we had to clean it by referencing asins (unique product ID).

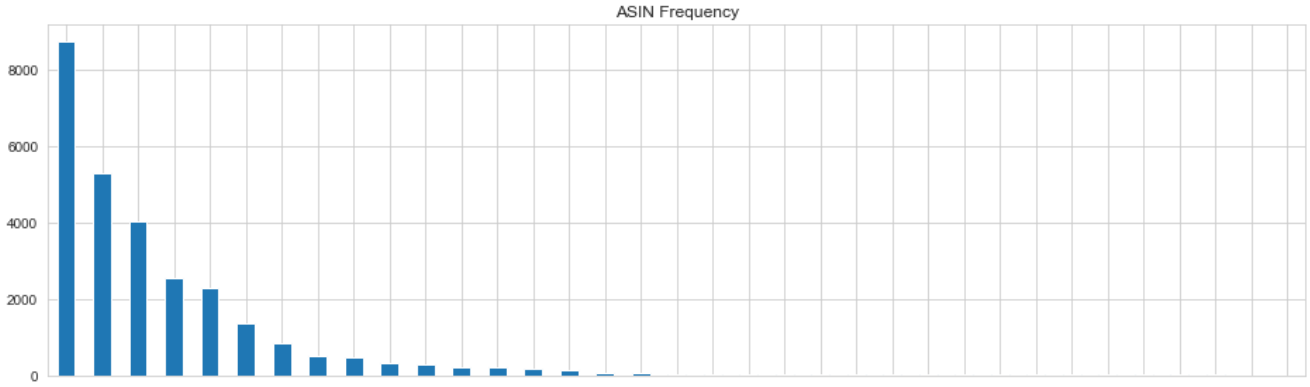
Based on the training data we found that only 35 products are present in the dataset .One corresponding to each ASIN. But 47 product names were found. This implies that comparatively more unique names with a slight variation in title (like 64 GB vs. 64 GB) or a lot of missing names (NAN) are present. Hence, we will be working with ASINs of the products for our analysis. Also we found that a single ASIN had two product names in our dataset due to different vendor listings. Hence, it's imperative to work with ASINs for sentiment analysis of the product reviews.

4. VISUALIZATIONS

4.1 ASIN FREQUENCY:

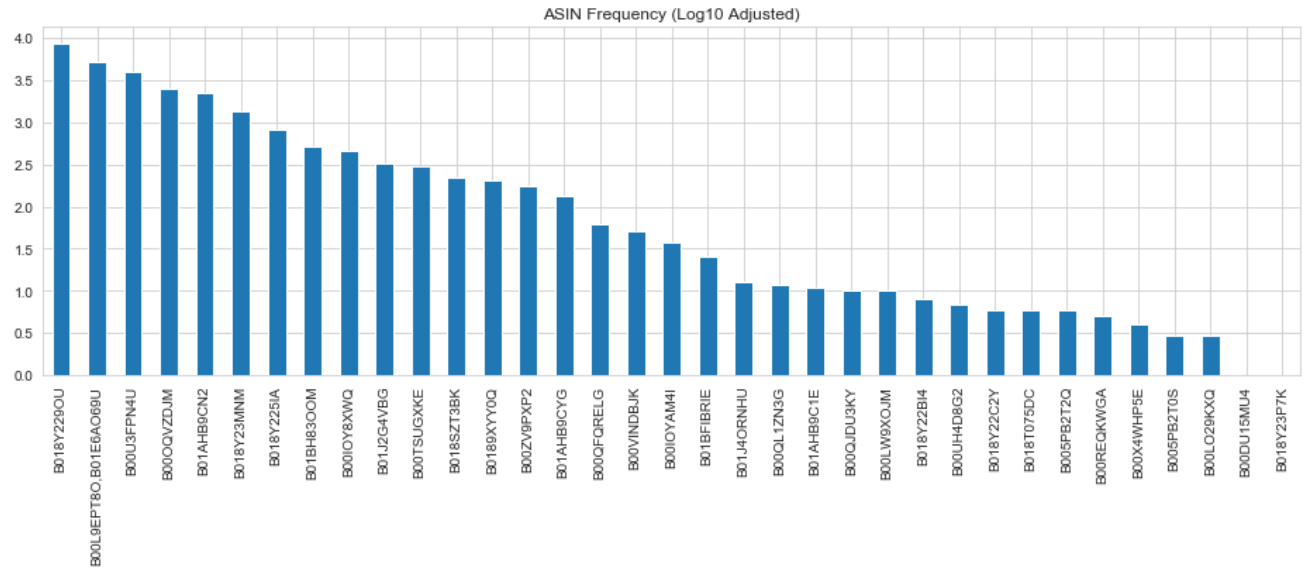
In the following bar graphs we have plotted the ASINs of the products on the x axis and frequency of these ASINs in the reviews on the y axis. It clearly shows that certain products have significantly more reviews than other products. This may indicate that those specific products have comparatively higher sales. We can also observe a “right tailed” distribution which suggests that there exists a correlation between higher ASINs frequency in the reviews and higher sales of the product.

Figure 4.1 (a): ASIN Frequency



We also normalized the data by using logarithmic function on the ASINs data to get a better representation of the distribution on the bar graph represented in fig 4.1(b). We again observed a similar “right tailed” distribution.

Figure 4.1 (b): ASIN Frequency after normalizing the data



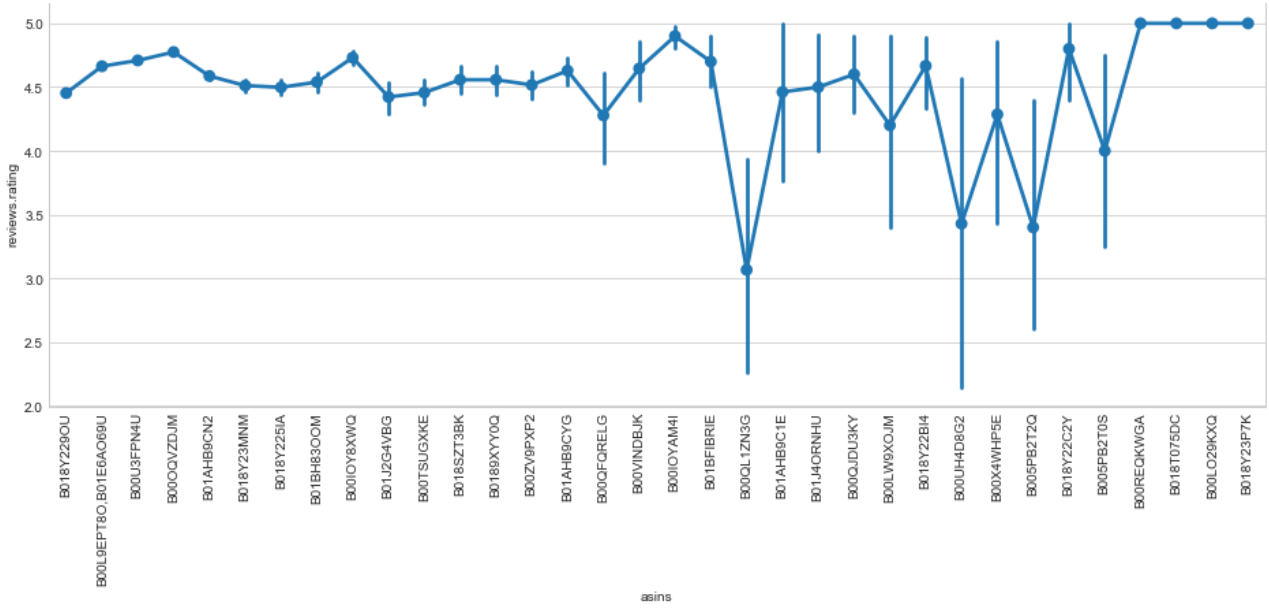
The above visualization helps us to answer our first research question that products can be kept or dropped from the stock based on higher or lower sales of that ASINs (product).

4.2. Reviews.rating/ASINs

The point plot in fig 4.2 represents the relationship between ASINs (plotted on the x-axis) and reviews.rating (plotted on the y-axis). We can observe that the most frequently reviewed ASINs (as deduced from fig 4.1) have an average review rating between 4.5-4.8 with not a very significant variance. Although a slight inverse relationship for the first four ASINs is observed, it's not of much significance due to an overall good rating range. For ASINs with lower frequencies we can observe a relatively higher variance as shown by the vertical lines in the point plot. Hence, the average review rating of these lower frequency ASINs are not significant for our analysis. Although we can suggest that this variance can indicate lower quality products.

Furthermore, towards the right end of the graph, we can observe that the last four ASINs have no variance and they stand at a perfect review rating of 5.0. But we will not consider these in our analysis due to their lower frequencies (as deduced from fig 4.1).

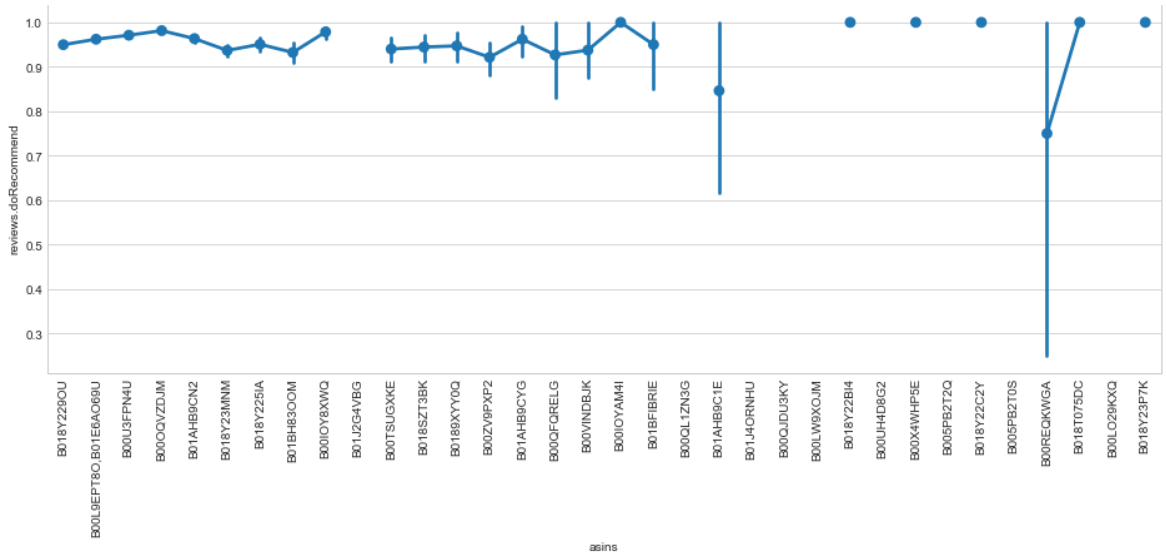
Fig 4.2.2 Relationship between ASINs and review ratings



4.3 Relationship between ASINs and Review.doRecommend

The point plot in the fig 4.3 shows the relationship between ASINs (plotted on the x-axis) and review.doRecommend (plotted on y-axis). We can observe from the initial 19 ASINs that the consumer recommend the product which supports our observation that the initial 19 ASINs have good review ratings ranging between 4.0 to 5.0 (refer section 4.2).Furthermore, the remaining ASINs are observed to have fluctuating results which is due to lower sample size. Hence, we will not consider them for our analysis.

Fig 4.2.3 Relationship between ASINs and Review.doRecommend

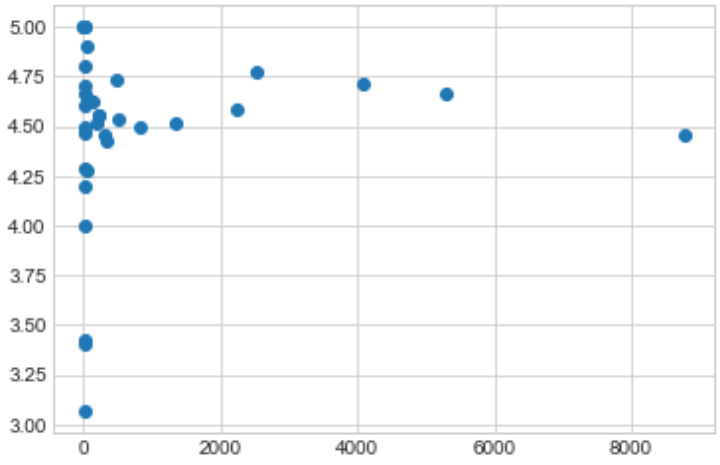


4.4 CORRELATION BETWEEN ASINs AND REVIEW RATINGS

The scatter plot in the fig 4.4 shows the correlation between ASINs (plotted on x axis) and Review.rating (plotted on y axis).

We observed in section 4.2 that our dataset contains few ASINs with low frequencies that have high variance and we concluded that we will not use these ASINs for our analysis given the low sample size. The same inference can be drawn from the scatter plot in fig 4.4 since there exist almost no correlation which is consistent with our findings.

Fig 4.2.4 Scatter plot of ASINs vs. Review.rating



5. ALGORITHMS

5.1. BAG OF WORDS

To convert the contents of product review into numerical feature vectors, we used the Bag of Words strategy which we implemented using SciKitLearn's CountVectorizer.
The words present in the training set are represented as an array X [i, j] where i is the integer indices and j is the word occurrence.

5.2 MULTINOMINAL NAIVE BAYES

Multinomial Naive Bayes classification algorithm is most suitable classifier for word counts where data are typically represented as word vector counts. It estimates the conditional probability of a particular word given in a class as the relative frequency of term in documents belonging to that class. The variation takes into account the number of occurrences of the term in training documents from the class, including multiple occurrences. This algorithm assumes that x_i's are conditionally independent given y, which is Naïve Bayes assumption.

$$\boldsymbol{p(x, \ldots\ x_k|y) = \prod_{i=1}^{k} p(xi_i|y)}$$

For representing review texts, it takes an array of non-negative integers, and models p (x_i |y) with multinomial distribution.

5.3 LOGISTIC REGRESSION:

Logistic regression is a classification method which predicts the binary output value (0 or 1). It uses coefficient values or weights to combine the input values linearly to predict the output value.
Below is an example logistic regression equation

y = e^ (b0 + b1*x) / (1 + e^ (b0 + b1*x))

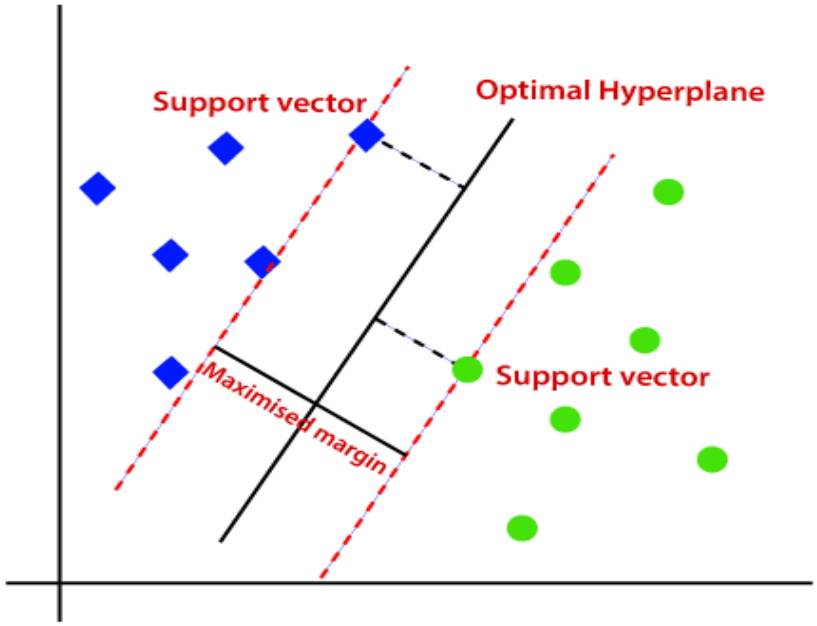
Where y is the predicted output, b0 is the bias or intercept term and b1 is the coefficient for the single input value (x). Each column in the input data has an associated b coefficient (a constant real value) that must be learned from our training data.

The actual representation of the model that we will store in memory or in a file are the coefficients in the equation.

5.4 LINEAR SUPPORT VECTOR MACHINE:

Linear SVM is a supervised learning algorithm which is used for both classification as well as regression problem. It selects the extreme points in the dataset, known as support vectors, to create hyperplanes. The distance between hyperplane and vectors is called margin. It finds the optimal decision boundary by maximizing the margin. It can be understood by the figure 5.4 below.

Figure 5.4: Linear Support Vector Machine Classifier.
(*Source: javatpoint.com\machine-learning-support-vector-machine-algorithm)*



5.5 DECISION TREE:

Decision tree is used for classification and prediction of sentiments in our project. It's a type of supervised learning. It is represented by trees for solving problems in which the leaf nodes represent the class label and the internal nodes of the tree represent the attribute of the class. Identification of the attribute for the root node can be done using the following two measures:

INFORMATION GAIN: It's the measure of the change in entropy during partitioning of the training instances into similar subsets.

$$\textbf{Gain (I, T) = Entropy (I) - } \sum_{v \in Values\,(T)} \left|\frac{I_v}{I}\right| . Entropy\,(I_v)$$

GINI INDEX: Measures the frequency of incorrectly identified randomly chosen element.

$$\textbf{GiniIndex = 1 - } \sum_k t_k^2$$

Random Forest:

Random forest is a supervised learning algorithm which is used for both classification as well as regression. It contains a large number of decision trees on various subsets of the given dataset. It takes the prediction of all the decision trees and based on the majority votes of prediction, it predicts the final output. This leads to higher accuracy and eliminates the problem of overfitting.

$$K_k^{cc}(x,z) = \sum_{k_{i,\ldots,k_d}, \sum_{j=1}^{d} k_j = k} \frac{k!}{k_1! \ldots k_d!} \left(\frac{1}{d}\right)^k \prod_{j=1}^{d} \mathbf{1}_{\left[2^{k_j} x_j\right] = \left[2^{k_j} z_j\right]}$$

$$\textbf{For all x,z} \in [0,1]^{\mathbf{d}}$$

6. RESULT

We divided the entire dataset of 34,660 costumer reviews into 27,701 training samples and 6,926 testing samples. By using bag of words strategy to our dataset we found 12,526 distinct words in our training sample which we trained using Multinomial Naive Bayes, Logistic Regression classifier, Linear Support vector machine classifier, Decision tree classifier and random forest classifier to predict polarity of the review. The accuracy of each of these classifier found using the testing sample is represented in the table below.

MODEL	TEST ACCURACY
Multinomial Naive Bayes	93.44%
Logistic Regression	93.70%
Linear Support Vector Machine	93.94%
Decision Tree	90.18%
Random Forest Classifier	93.45%

We found that Linear Support Vector Classifier generated the best predictions. Hence, we fine tune this classifier to avoid any potential over-fitting. This in turn improved the accuracy of our model to 94.08%. Hence, we performed a detailed performance analysis on it. The results are summarized in the table below.

	Precision	Recall	F1 Score	Support
Negative	0.67	0.25	0.36	156
Neutral	0.47	0.11	0.18	292
Positive	0.95	1.00	0.97	6473
Average/total	0.92	0.94	0.92	6926

Accuracy: 0.9408027721628646

7. CONCLUSION AND FUTURE WORK:

Our analysis supports our initial data exploration analysis that the dataset is much skewed to the positive reviews. Negative and neutral reviews are not significant for our analysis, due to large standard deviation with very small frequency, which is evident by the lower precision, recall and F1 scores in the classification results.

But despite the skewed dataset, we were still able to build a Sentiment Analysis machine learning system with 94.08% accuracy level to determine the polarity of the reviews. We tested it by inputting arbitrary text and it showed great results, since the machine learning system was able to learn from all the positive, neutral and negative reviews. We also fine-tuned the algorithm to avoid bias sentiments.

Hence, as we will continue to input new dataset in the future containing balanced reviews and our model will re-adjust to a more balanced classifier and hence, the accuracy level will also increase.

For future perspective, we would like to expand the functionalities of our model by applying unsupervised learning algorithms as well as switching from unigram to bigram model which will increase the linkage between data and provide more accurate sentiment analysis results. Moreover, we would like to make a web application for users to input keywords and get analyzed results.

.

REFRENCES:

- [1] Sultana, Najma & Kumar, Pintu & Patra, Monika & Chandra, Sourabh & Alam, Sk. (2019). *SENTIMENT ANALYSIS FOR PRODUCT REVIEW*. International Journal of Soft Computing. 09. 7. 10.21917/ijsc.2019.0266.
- [2] K. Dave, S. Lawrence, and D. M. Pennock. *Mining the peanut gallery: Opinion extraction and semantic classification of product reviews*. In Proceedings of the 12th international conference on World Wide Web, pages 519–528. ACM, 2003
- [3] Y. Xu, X. Wu, and Q. Wang. *Sentiment analysis of yelps ratings based on text reviews*, 2015
- [4] M. S. Elli and Y.-F. Wang*. Amazon reviews, business analytics with sentiment analysis*.
- [5] *C. Rain. Sentiment analysis in amazon reviews using probabilistic machine learning*. Swarthmore College, 2013.
- [6] https://www.kaggle.com/getting-started/44916