Sentiment Analysis on Consumer Reviews of Amazon Products

Bala Mendu, Sharanya Chiluka
School of Computer Science and Engineering
Oakland University

Abstract - **In the last few years sentiment analysis has made much progress. Sentiment analysis has been used in several applications to identify the opinions of people, products, brands, services, etc., which can, for example, improve a company's business. Sentimental Analysis is mainly meant for classifying the text based on its polarity. Opinion Mining is one of the major categories in sentimental analysis. Opinion of any user in buying a product or rating a movie contributes highly to the product or movie For example: If an online product is given various levels of star rating, then the other customers who think of buying the product might have an overview on the rating and then decide whether to buy the product or not.For selecting a product, a customer needs to go through thousands of reviews to understand a product. But in this prospering day of machine learning, going through thousands of reviews would be much easier if a model is used to polarize those reviews and learn from it. We used supervised learning methods on a large scale amazon dataset to polarize it and get satisfactory accuracy.**

Keywords - **Sentiment Analysis, Feature extraction, Naive Bayes Classifier, Natural Language Toolkit (NLTK).**

I. INTRODUCTION

Sentimental analysis is one of the important fields in text mining. The varying thoughts of different users in any of the particular categories are gathered as a single dataset and are being considered for the analysis. Generally, the analysis could be done with the help of any of the machine learning techniques. It helps in effective classification of the text.

In the emerging trend of online shopping, online movie booking, the opinion of the existing users is very much necessary for the users to go for a better choice. Hence the need for sentimental analysis is keeping on increasing. This analysis helps the users to make decisions in any of the tough and confusing situations based on the reviews [1].The overall sentiment defines various emotions of the users such as (sad, angry, happy) will help the users to make further decisions. The texts are generally classified into two main categories namely positive and negative. And finally the category which is predominant is declared to be the category of the text or the sentence.

In our model, We used a dataset which consists of various Amazon product reviews like the Kindle, Fire TV Stick, and more provided by Datafiniti's Product Database and predicted for each review given for those products as positive or negative review. We went through step by step analysis of the project by analysing the Data initially then Data cleaning, preprocessing the text reviews using various techniques.We used combination of two kinds of approaches to extract features: one is using NLTK Naive Bayes and the other one using Count vectorization & Tf-Idf approach for getting higher accuracy.After Feature extraction various classification methods are applied and compared.

II. LITERATURE REVIEW

Several algorithms have been applied in the field of sentimental analysis over the past few years. In [4], the author focuses on the sentimental analysis of book reviews using both supervised and unsupervised approaches.Applied the popularly used techniques namely NB, SVM and Semantic Orientation based SO-PMI-IR on two datasets from GoodReads and Amazon for sentiment classification. Finally, unsupervised algorithms gave better results when the dataset contains long phrases whereas supervised algorithms give higher accuracy on the dataset containing short one-lined reviews.

In [5], the author focuses on the clustering of the positive and the negative feedbacks using K-means clustering algorithm and determines the cluster labels.

In [6], the author focuses on the generation of reviews which are trustworthy which will guide the customer in accurate decision making. Compare the original and reverse reviews and applied Dual Sentiment Analysis Algorithm for training and prediction. Applied the Classification algorithms Naive Bayes and SVM for comparing the results generated by the algorithms to improve the accuracy. Finally, the result is that SVM is

comparatively better (91%) than compared to Naive Bayes (66%).
In [7], the author focuses on classification based on the prior and posterior probability by considering word occurrence mainly.Calculated accuracy using Naive Bayes Algorithm and the result is that the accuracy of Naïve Bayes algorithm is 89%

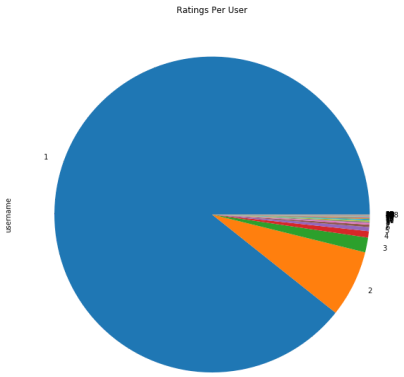
III. METHODOLOGY

A. Collection of Data

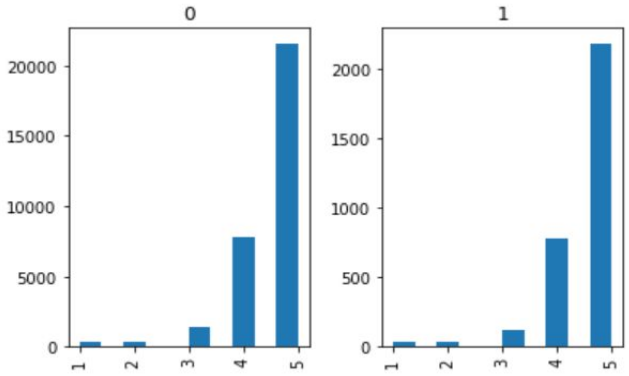
This is a list of over 34,660 consumer reviews for Amazon products like the Kindle, Fire TV Stick, and more provided by Datafiniti's Product Database. The dataset contains 21 columns with basic product information, rating, review text for each product. Username, rating, review text and title are the main features of the dataset.

B. Dataset Analysis

There are a total of 26,789 total users. Out of them, 146 users are bulk users who have given ratings more than 10 times. Users who have given single ratings are more than bulk users.



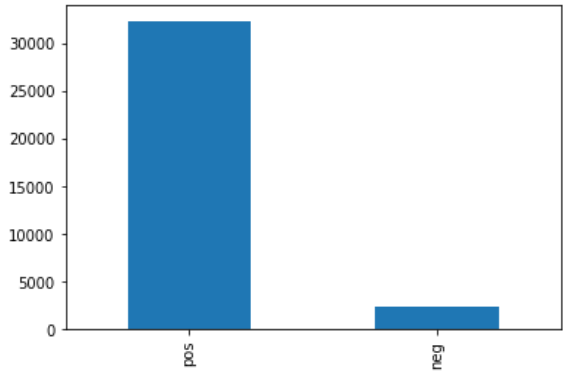
The distribution of ratings given by bulk and normal users are the same. We don't think that bulk users are spam.



When we observed rating distribution, more than half of the users gave good ratings either 4 or 5.

C. Data Cleaning

Filtered rows which don't have user ratings are nothing but null values present in the dataset. Classified the review text as positive and negative. If the review rating is greater than or equal to 4, classify the review text as positive and if the review rating is less than 4, classify the review text as negative. Plotted bar graph with positive and negative reviews ratio.



D. Text Preprocessing

Whenever we have textual data, we need to apply several preprocessing steps to the data to transform words into numerical features that work with machine learning algorithms. In Preprocessing, unnecessary text and other symbols which are not at all required for classification purpose are being removed.

There are several preprocessing techniques in Machine learning. All we need to apply is also based on the project/application that we build. In this project, we have applied the following steps:Sentence lowering, Removing white spaces,Removing

Punctuation marks.

Sentence Lowering: Converting a word to lowercase (NLP -> nlp).Words like Book and book mean the same but when not converted to the lower case those two are represented as two different words in the vector space model (resulting in more dimensions).Sentence lowering in NLTK is usually done by using keyword lower ( ).

Removing White Spaces: To remove leading and ending spaces, you can use the strip() function.Leading and ending spaces are nothing but the spaces in front and at the end of the word.

Lemmatization: Lemmatization is the algorithmic process of determining the lemma of a word based on its intended meaning. Unlike stemming, Lemmatization depends on correctly identifying the intended part of speech and meaning of a word in a sentence, as well as within the larger context surrounding that sentence, such as neighboring sentences or even an entire document.

Word	***Lemmatization***	***Stemming***
was	be	wa
studies	study	studi
studying	study	study

Table 1: Depicting the outputs of Lemmatization and Stemming for original words

Lemmatization is preferred over Stemming because lemmatization does a morphological analysis of the words.We have used WordNetLemmatizer in order to lemmatize the reviews.

E. Feature Extraction

We have done two different types of feature extraction.One is used for Naive Bayes Classifier i.e., after processing the review text, A word_features function is used to extract the features. This is the main feature extraction step, here we are iterating our preprocessed text and passing each text containing review words and its category to the find_features function. In this function we are checking if the a review words are present in the complete word_features list, if yes, then we are marking them as 'true' and remaining as 'false' word_features as 'false'.Transforming all the review text records in specific format, where every list item has been replaced with a dictionary(i.e., key ,value pair).Where key are the words and value will be either True or False.

The other type of feature extraction is done using Count Vectorization and TF-IDF which is fed into different sets of algorithms.

Count Vectorization: In Count vectorization We will be creating vectors that have a dimensionality equal to the size of our vocabulary, and if the text data features that vocab word, we will put a one in that dimension. Every time we encounter that word again, we will increase the count, leaving 0s everywhere we did not find the word even once.The result of this will be very large vectors, if we use them on real text data, however, we will get very accurate counts of the word content of our text data. Unfortunately, this won't provide use with any semantic or relational information.

Basically, Count Vectorization involves counting the number of occurrences each word appears in a document (i.e distinct text such as an article, book, even a paragraph!). Python's Sci-kit learn library has a tool called CountVectorizer to accomplish this.Example sentence: "The weather was wonderful today and I went outside to enjoy the beautiful and sunny weather." You can tell from the output below that the words "the", "weather", "and "and" appeared twice while other words appeared once. That is what Count Vectorization accomplishes.The features used in Counter Vectorization function are:

ngram_range: This deals with contiguous sequences of words. Let's say you want to deal with features such as "peanut butter and jelly", "intercontinental flight", or "rush hour traffic", you can get a lot more meaning than if you put all of these words as independent features. You could lose the meaning if you don't use this parameter.

Stop words: Stop words are common words (i.e. English stop words such as a, of, and, the) that can be removed in order to focus on more relevant words in your analysis.

Min_df: When building the vocabulary ignore terms that have a document frequency strictly lower than the given threshold.

Term Frequency and Inverse Document Frequency (TF-IDF) : Whether a word appears frequently in some documents but less frequently in others can be really useful! This is where Term Frequency-Inverse Document Frequency (TF-IDF) comes in. The term frequency refers to how much a term (i.e. a word) appears in a document. Inverse document frequency

refers to how common or rare a term appears in a document.Inverse document frequency takes the logarithmic function of the size of the set of documents, D, and in how many documents a word appears even if it appears more than once within a document. This is then multiplied by the term frequency to get a score.

If the TF-IDF score is pretty high, it means the words is pretty rare and is good at discriminating between documents.When words do have high TF*IDF weight in content, content will always be amongst the top search results, so anyone can:

1. Stop worrying about using the stop-words,
2. Successfully find words with higher search volumes and lower competition.

Here,

CountVectorizer - Transforms text into a sparse matrix of n-gram counts.

TfidfTransformer - Performs the TF-IDF transformation from a provided matrix of counts.

F. Data Classification

After the Feature Extraction step, the dataset is classified into training and testing.The first method of feature extraction is used and fed into Naive Bayes Classifier.

Naive Bayes : The Naive Bayes classifier is a likelihood classifier, in light of Bayes' hypothesis. Bayes' hypothesis determines scientifically the connection between likelihood of two occasions An and B, P(A) and P(B) and contingent likelihood of occasion A molded by B and occasion B adapted by A, P (A | B) and P(B | A). Consequently Bayes' equation is :

$$P(A/B)= (P(B/A) * P(A)) / P(B)$$

This hypothesis empowers us to decide a contingent likelihood having the likelihood of opposite occasions and autonomous probabilities of occasions. In this way, we can gauge the likelihood of an occasion taking into account the case of its event. Along these lines, we can evaluate the likelihood of an occasion in view of the case of its event. For this situation, we assess likelihood that a record is sure or negative, in a specific setting, or the probability that an occasion to happen in the event that it was foreordained to be certain or negative [2].NLTK Naive Bayes is used for the classification.

For the next classifiers Count vectorization and TF-IDF are used as feature extractors.

Multinomial Naive Bayes: This is the event model typically used for document classification with discrete features (eg. Word counts for text classification). In text learning, the frequency of occurrence of each word in the document is taken into account to predict the class or label. Implements the algorithm for multinomially distributed data.

Bernoulli Naive Bayes: It assumes that all our features are binary variables such that they take only two values (i.e 0s and 1s). Means 0s can represent “word does not occur in the document” and 1s as "word occurs in the document" .

Linear Regression: It is one of the most simple and commonly used Machine Learning algorithms for two-class classification. It is easy to implement and can be used as the baseline for any binary classification problem.

IV. RESULTS

There were several Machine learning algorithms used in our experiment like Naive Bayesian, Multinomial Naive Bayes, Bernoulli Naive Bayes and Linear Regression. We have used the Feature extractor function for the NLTK Naive Bayes Classifier for which the accuracy obtained was 58.9 % . Then we went with another classifier with different classification algorithms.

The performance metrics were used there to measure the classification performance.Accuracy measure is the most common for this purpose. The accuracy of a classifier on a given test dataset is the percentage of those dataset which are correctly classified by the classifier [3]. And for the text mining approach the accuracy measure is not enough to give a proper decision so we also took some other metrics to evaluate classifier performance. Three important measures are commonly used precision, recall, F-measure. Before discussing with different measures there are some terms we need to get comfortable with-

- TP (True Positive) represents numbers of data correctly classified
- FP (False Positive) represents numbers of correct data misclassified
- FN (False Negative) represents numbers of incorrect data classified as correct
- TN (True Negative) is the numbers of incorrect data classified

Precision: Precision measures the exactness of a classifier, how

many of the return documents are correct. A higher precision means less false positives, while a lower precision means more false positive. Precision (P) is the ratio of numbers of instances correctly classified from total. It can be defined as-

$$P = \frac{TP}{TP+FP}$$

Recall: Recall calculates the sensitivity of a classifier; how many positive data it returns. Higher recall means less false negatives. Recall is the ratio of the number of instances accurately classified to the total number of predicted instances. This can be shown as-

$$R = \frac{TP}{TP+FN}$$

F-Measure: Combining precision and recall produces single metrics known as F-measure, and that is the weighted harmonic mean of precision and recall. It can be defined as -

$$F = \frac{2P.R}{P+R}$$

Accuracy: Accuracy predicts how often the classifier makes the correct prediction. Accuracy is the ratio between the number of correct predictions and the total number of predictions.

Label	Classifier	Precision	Recall	F1 Score
Positive	Multinomial naive Bayes	0.00	0.00	0.00
	Bernoulli Naive Bayes	0.33	0.17	0.23
	Logistic regression	0.56	0.33	0.41
Negative	Multinomial naive Bayes	0.93	1.00	0.97
	Bernoulli Naive Bayes	0.94	0.97	0.96
	Logistic regression	0.95	0.98	0.97

Table 2: Experimental result for classifiers

Classifier	Accuracy
NLTK Naive Bayes Classifier	54.8 %
Multinomial Naive Bayes Classifier	93 %
Bernoulli Naive Bayes Classifier	91.7 %
Logistic Regression	93.86 %

Table 3 : Accuracy Results of Classifiers

We can clearly see that the accuracy obtained from Logistic regression is higher than the other classifiers.We have even tested a few handwritten samples using the logistic regression model.

test_sample_review(log_reg, "product is not affordable")
test_sample_review(log_reg, "The product was good and easy to use")
test_sample_review(log_reg, "the whole experience was horrible and product is worst")

Output:
Sample estimated as NEG: negative prob 0.554567, positive prob 0.445433
Sample estimated as POS: negative prob 0.000000, positive prob 1.000000
Sample estimated as NEG: negative prob 0.995168, positive prob 0.004832.

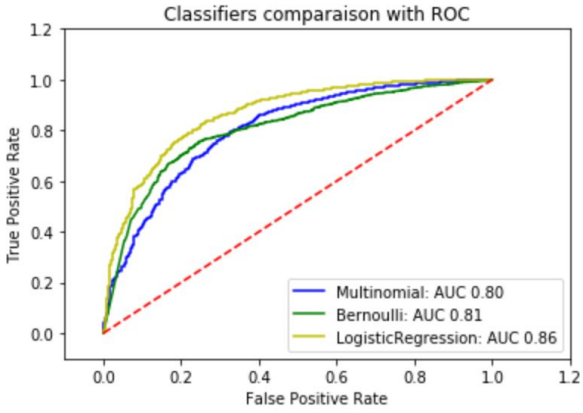
Which also shows the probability of the sentence being negative and positive.

V. COMPARATIVE ANALYSIS

In Machine Learning, performance measurement is an essential task. So when it comes to a classification problem, we can count on an AUC - ROC Curve. When we need to check or visualize the performance of the multi - class classification problem, we use AUC (Area Under The Curve) ROC (Receiver Operating Characteristics) curve. It is one of the most important evaluation metrics for checking any classification model's performance. It is also written as AUROC (Area Under the Receiver Operating Characteristics).

AUC - ROC curve is a performance measurement for classification problems at various thresholds settings. ROC is a

probability curve and AUC represents degree or measure of separability. It tells how much a model is capable of distinguishing between classes. Higher the AUC, better the model is at predicting 0s as 0s and 1s as 1s. By analogy, Higher the AUC, better the model is at distinguishing between patients with disease and no disease.An ROC curve plots the true positive rate on the y-axis versus the false positive rate on the x-axis.



An excellent model has AUC near to the 1 which means it has good measure of separability. A poor model has AUC near to 0 which means it has the worst measure of separability. In fact it means it is reciprocating the result. It is predicting 0s as 1s and 1s as 0s. And when AUC is 0.5, it means the model has no class separation capacity whatsoever.

From the above graph we can clearly make a conclusion that Logistic Regression has higher AUC which means that it performs better than any other classifier used.

VI. CONCLUSION AND FUTURE WORKS

In recent years with the vastly increasing amount of customer data around the digital world, text analysis is gaining more adoption.The user can easily compare the different sentiment reviews of the different products. We proposed our model which is a supervised learning method and used 2 kinds of feature extractor approach. Feature extraction of bag of n grams with TF-IDF works better than NLTK Naives bayes feature extraction. The accuracy percentage obtained is also higher than 90% which features are a good model.As the part of Future Work, we want to improvise or apply new feature extraction techniques which are more powerful and extracts more accurate features.

VII. REFERENCES

[1] Zhen Hai , Gao Cong, Kuiyu Chang , Peng Cheng , Chunyan Miao, “Analyzing Sentiments in One Go: A Supervised Joint Topic Modeling Approach”, IEEE Transactions on Knowledge and Data Engineering (Volume: 29 , Issue: 6 , June 1 2017).

[2] Smeureanu, Ion, and Cristian Bucur. "Applying Supervised Opinion Mining Techniques on Online User reviews." InformaticaEconomica 16.2 (2012): 81-91.

[3] Tanjim Ul Haque, Nudrat Nawal Saber, Faisal Muhammad Shah, “ Sentiment Analysis on Large Scale Amazon Product Reviews“, IEEE International Conference on Innovative Research and Development 11-12 May 2018,Bangkok,Thailand.

[4]Vipin Deep Kaur, “Sentimental Analysis of Book Reviews using Unsupervised Semantic Orientation and Supervised Machine Learning Approaches”Second International Conference on Green Computing and Internet of Things (ICGCIoT)

[5] Atharva Patil,Nishita S. Upadhyay, Karan Bheda, Rupali Sawant, “Restaurant’s Feedback Analysis System using Sentimental Analysis and Data Mining Techniques”, Proceeding of 2018 IEEE International Conference on Current Trends toward Converging Technologies, Coimbatore, India

[6] Stephina Rodney D’souza,Kavita Sonawane “Sentiment Analysis Based on Multiple Reviews by using Machine learning approaches”,IEEE Xplore Part Number: CFP19K25-ART; ISBN: 978-1-5386-7808-4

[7] Surya Prabha PM,Subbulakshmi B,”Sentimental Analysis using Naive Bayes Classifier”, 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN)