SUMMER INTERNSHIP PROGRAM 2019

20th May,2019 - 5th July,2019

Sentiment Analysis of Amazon Reviews

Final Report

MALAVIYA NATIONAL INSTITUTE OF TECHNOLOGY

ROBOTICS AND MACHINE ANALYTICS RAMAN LAB

Submitted by:

Ujjawal Gupta
National Institute of
Technology, Hamirpur
gujjawal29@gmail.com

Shreya Jakhar
MODY University,
Laxmangarh
shreyajakhar22@gmail.com

Under the supervision of
Prof.(Dr.) Rajesh Kumar
Department of Electrical Engineering
Malaviya National Institute of Technology
Jaipur, Rajasthan, India

Contents

1 Introduction

Sentiment analysis of product reviews, an application problem, has recently become very popular in text mining and computational linguistics research. Here, we want to study the correlation between the Amazon product reviews and the rating of the products given by the customers. We use both traditional machine learning algorithms including Multinomial Naive Bayes, Bernoulli Naive Bayes, Random Forest, Logistic Regression and Bagging. By comparing these results, we could get a better understanding of the these algorithms. They could also act as a supplement to other fraud scoring detection methods.

2 Problem Development

Recent years have seen an increasing amount of research efforts expanded in understanding sentiment in textual resources.One of the subtopics of this research is called sentiment analysis or opinion mining, which is, given a bunch of text, we can computationally study peoples opinions, appraisals, attitudes, and emotions toward entities, individuals, issues, events, topics and their attributes. Applications of this technique are diverse. The objective of this paper is to classify the positive and negative reviews of the customers over different products and build a supervised learning model to polarize large amounts of reviews. Our dataset consists of customer's reviews and ratings. We extracted the features of our dataset and built several supervised model based on that.These models include algorithms such as Multinomial Naive Bayes, Bernoulli Naive Bayes, Random Forest, Logistic Regression and Bagging. We compared the accuracy of these models and got a better understanding of the polarized attitudes towards the products.

3 Methodology

3.1 Multinomial Naive Bayes

With a multinomial event model, samples (feature vectors) represent the frequencies with which certain events have been generated by a multinomial $\mathrm{p_1}$, ...,$\mathrm{p_n}$ where $\mathrm{p_i}$ is the probability that event i occurs (or K such multinomials in the multiclass case). A feature vector x = $\mathrm{x_1}$, ...,$\mathrm{x_n}$ is then a histogram, with $\mathrm{x_i}$ counting the number of times event i was observed in a particular instance. This is the event model typically used for document classification, with events representing the occurrence of a word in a single document (see bag of words assumption). The likelihood of observing a histogram x is given by:

$$p(\mathbf{x} \mid C_k) = \frac{(\sum_i x_i)!}{\prod_i x_i!} \prod_i {p_{ki}}^{x_i} \tag{1}$$

3.2 Bernoulli Naive Bayes

In the multivariate Bernoulli event model, features are independent booleans (binary variables) describing inputs. Like the multinomial model, this model is popular for document classification tasks, where binary term occurrence features are used rather than term frequencies. If $\mathrm{x_i}$ is a boolean expressing the occurrence or absence of the $\mathrm{i^{th}}$ term from the vocabulary, then the likelihood of a document given a class $\mathrm{C_k}$ is given by:

$$p(\mathbf{x} \mid C_k) = \prod_{i=1}^{n} p_{ki}^{x_i} (1 - p_{ki})^{(1-x_i)} \tag{2}$$

Where p_{ki} is the probability of class C_k generating the term x_i.This event model is especially popular for classifying short texts. It has the benefit of explicitly modelling the absence of terms.$_i$

3.3 Random Forest

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees.Random decision forests correct for decision trees habit of overfitting to their training set.[1]

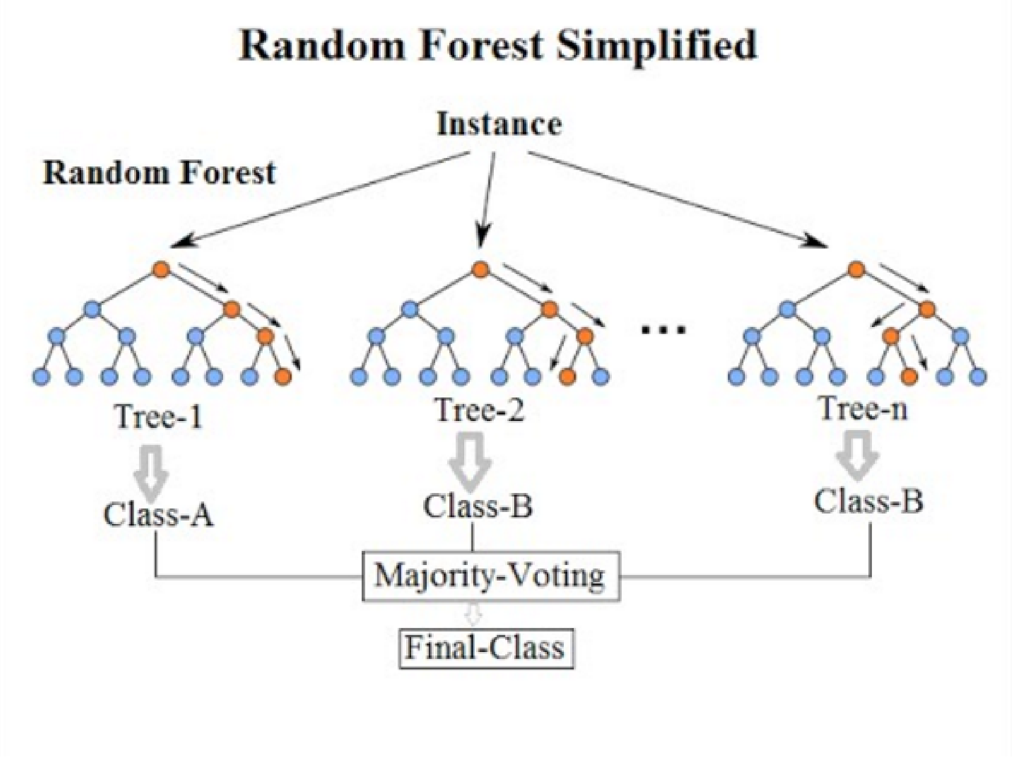


Figure 1: Random Forest[2]

3.4 Logistic Regression

Logistic regression is named for the function used at the core of the method, the logistic function. The logistic function, also called the sigmoid function was developed by statisticians to describe properties of population growth in ecology, rising quickly and maxing out at the carrying capacity of the environment. It's an S-shaped curve that can take any real-valued number and map it into a value between 0 and 1, but never exactly at those limits. Logistic regression uses an equation as the representation, very much like linear regression. Input values (x) are combined linearly using weights or coefficient values (referred to as the Greek capital letter Beta) to predict an output value (y). A key difference from linear regression is that the output value being modeled is a binary values (0 or 1) rather than a numeric value.[3]

3.5 Bagging

Bootstrap aggregating, also called bagging, is a machine learning ensemble meta-algorithm designed to improve the stability and accuracy of machine learning algorithms used in statistical classification and regression. It also reduces variance and helps to avoid overfitting. Although it is usually applied to decision tree methods, it can be used with any type of method.Bagging is a special case of the model averaging approach. Given a standard training set D of size n, bagging generates m new training sets D_i, each of size n', by sampling from D uniformly and with replacement. By sampling with replacement, some observations may be repeated in each D_i.If n'=n, then for large n the set D_i is expected to have the fraction (1 - 1/e) (≈63.2%) of the unique examples of D, the rest being duplicates.This kind of sample is known as a bootstrap sample. Then, m models are fitted using the above m bootstrap samples and combined by averaging the output (for regression) or voting (for classification).[4]

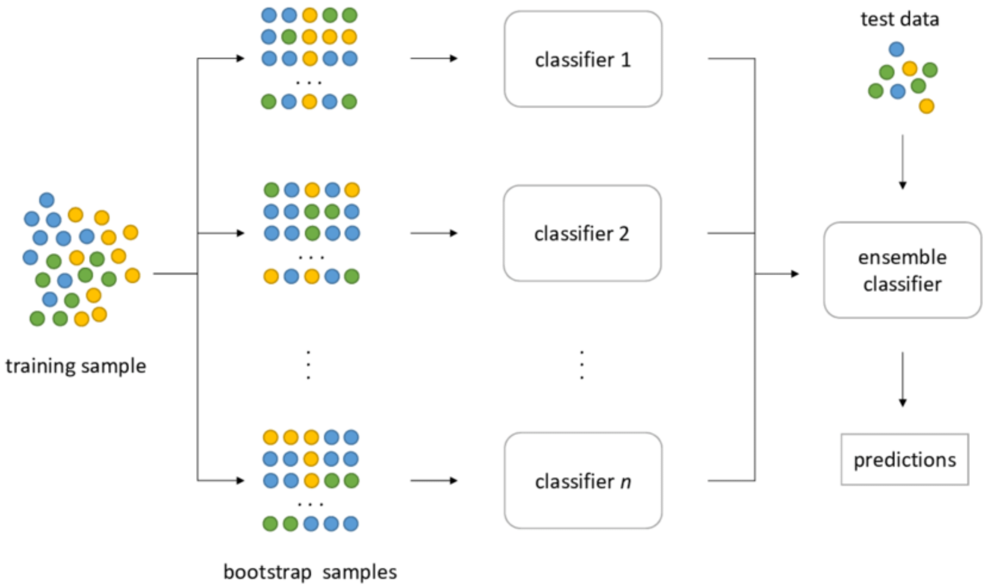


Figure 2: Bagging[5]

4 Simulations and Results

The entire dataset of 2,96,337 reviews was divided into a training set of size 217812 (80%) and a test set of size 54454 (20%).With 779569-d input features representing review text, we implemented Multinomial Naive Bayes, Bernoulli Naive Bayes, Random Forest, Logistic Regression and Bagging. Among all these classifiers Logistic Regression seems to perform best with 95.66% accuracy.

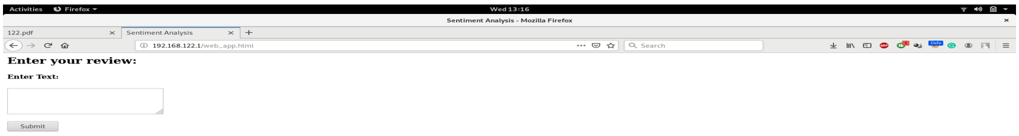
Table 1: Accuracy Table

Accuracy Score		
Models	Training Acc.	Test Acc.
Multinomial NB	93.66%	93.47%
Bernoulli NB	93.18%	92.99%
Random Forest	92.9%	92.9%
Logistic Regression	99.52%	95.8%
Bagging	95.6%	95.63%

On the basis of results obtained we used Logistic Regression as our final model.We used Logistic Regression in our web interface because of its high accuracy.

4.1 Our Web Interface

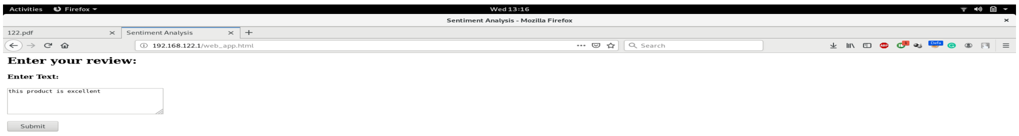
In our project we felt the need of one interface to make our project more productive and efficient.So we built one interface using Python CGI and HTML.In this interface we can enter the review in the form of statements be it short or long.This interface is used by the person who is managing the site.So he can check whether the entered reviews are positive or negative.When the review is entered on the interface it will recognize the positive and negative words used in the statement and will generate the result based upon that.The result that is obtained will show the probability of positivity and negativity of that review.



Enter your review:

Enter Text:

Figure 3: Interface

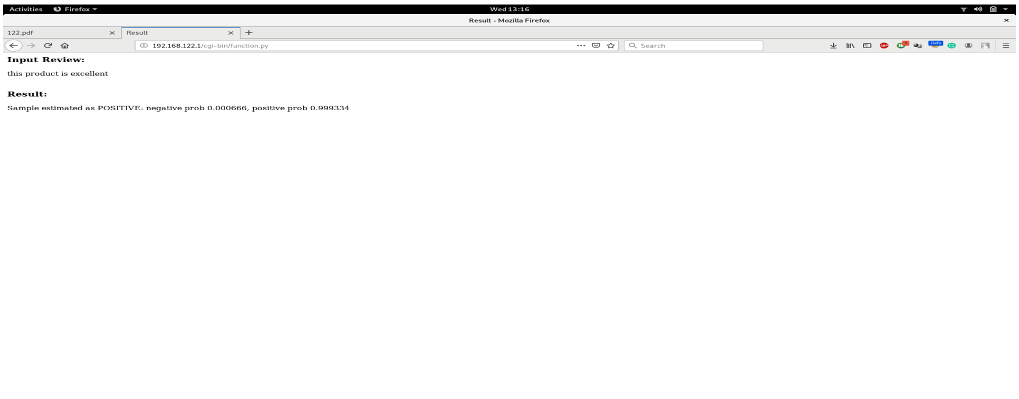


Enter your review:

Enter Text:

this product is excellent

Figure 4: First review



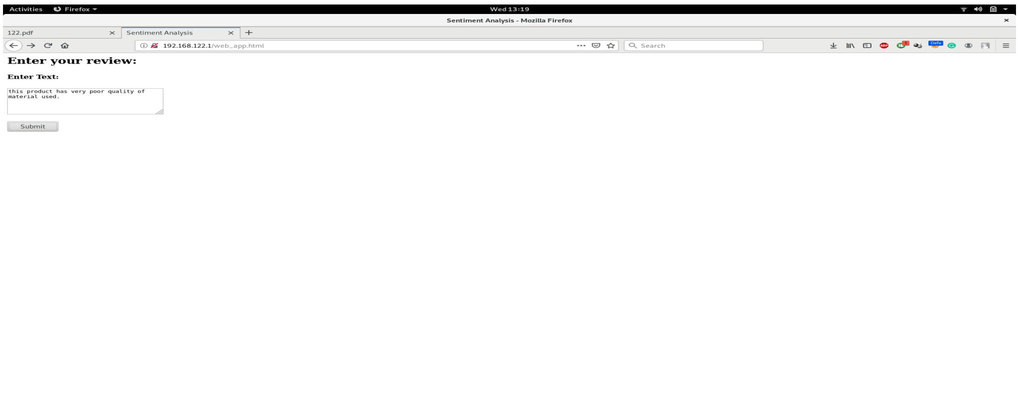
Input Review:

this product is excellent

Result:

Sample estimated as POSITIVE: negative prob 0.000666, positive prob 0.999334

Figure 5: Result


Activities
Firefox
Wed 13:19
Sentiment Analysis - Mozilla Firefox
122.pdf
Sentiment Analysis
192.168.122.1/web_app.html
Search

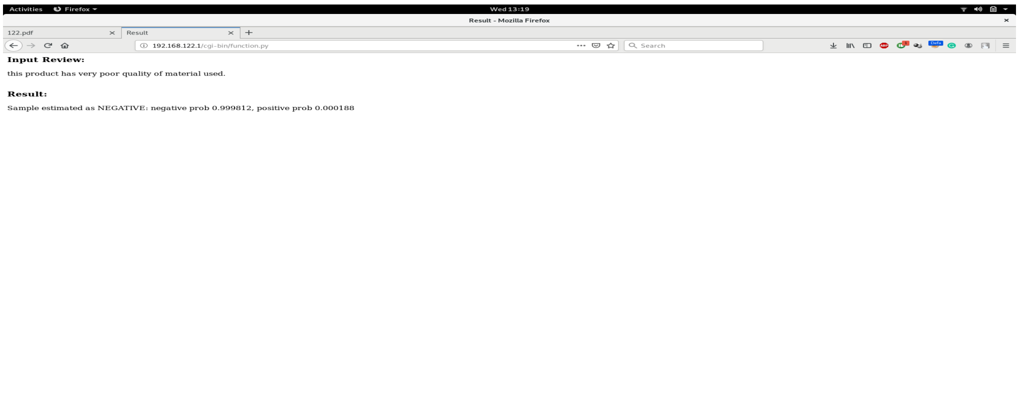
Enter your review:

Enter Text:

this product has very poor quality of material used.

Submit

Figure 6: Second review



Input Review:

this product has very poor quality of material used.

Result:

Sample estimated as NEGATIVE: negative prob 0.999812, positive prob 0.000188

Figure 7: Result

In our web interface we don't have any character limit as we can insert as many characters we want.Our interface is successful in predicting the results.
In our interface when we use twisted reviews the result that we obtain are less accurate than those with less twisted reviews.

5 Conclusions and Future Work

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.

As we know in this competition era many companies in order to uphold the positions in the market are making False reviews on competitor's companies products. So in order to come over this situation we need to develop one strong false review detection system which will detect the false review made on the products and will ensure transparency among the users.

Bibliography

[1] Van den Poel D. Prinzie, A. Random forests for multiclass classification: Random multinomial logit. 2008.

[2] Will Koehrsen. Random forest simple explanation. 2017.

[3] Michael I Ng, Andrew Y.; Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. 2002.

[4] Sotiris Kotsiantis. Bagging and boosting variants for handling classifications problems: a survey. 2014.

[5] Roberto Tagliaferr Paola Galdi. Data mining: Accuracy and error measures for classification and prediction. 2018.