iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 8 **Issue:** IX **Month of publication:** September 2020

DOI: https://doi.org/10.22214/ijraset.2020.31548

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue IX Sep 2020- Available at www.ijraset.com

Sentiment Analysis on Amazon Fine Food Reviews by using Linear Machine Learning Models

Maitrik Das[1], Sayantan Roy[2], Raj Saha[3]
[1, 2, 3]*TCS, India*

Abstract: ***The famous ecommerce site Amazon has the mammoth source of fine food information along with their reviews and ratings given by the users encompassed with their personal information for the span of at least ten years up to 2012. This user given reviews elucidates the tricky role of the product assessment as concerned with the current market scenario. We have performed the sentiment analysis on review-texts and ratings which has helped to accentuate the needy purpose of the market research for such products. Sentiment Analysis is like a machine learning function having loomed over plethora of expressions in the form of reviews from the end users in order to derive the polarity either as positive or as negative. We have also augmented our analysis with applying machine learning algorithm on the structured and preprocessed data along with the sentiment analysis prior to Natural Language Processing. These overall studies will sincerely provide some of the significant insights of the business and also will give the positive goal about the future product review analysis.***

I. INTRODUCTION

Nowadays, in the digital world users are very aware and particular about expressing their opinions on certain products once they have used, they love to write it as reviews in the ecommerce websites in the most lucid way. This has helped the concerned company identifying the issues for those products which they can able to fix faster as well and moreover, manufactures also become aware about the performance of those products in the market. Sentiment Analysis plays an important role in terms of detecting the contextual polarity of the reviews, in other way it is also known as opinion mining as it determines the attitude of the users towards the products from the huge volume of opinions i.e. reviews. Here, we have adopted certain preprocessing techniques as a part of the NLP such as tokenization, stemming or Lemmatization followed by the removal of stop-words, HTML tags with the help of different Python libraries. Then we have availed some of the vectorizer techniques such as Bag of Words, Tf-Idf and W2V which has converted texts into vectors so that the Machine Learning algorithms can be applied onto these. After converting into vectorized data, we have applied classification algorithms such Logistic Regression and SVM to prognosticate the polarity of the reviews. We also have ratings of these products, if the ratings are 1 or 2 the corresponding products are said to fall under the negative polarity and if the ratings are 4 or 5 it belongs to the positive polarity. For those products having rated as 3, We have kept those as neutral.

II. METHODOLOGY

A. Data Collection

We have collected this data from the official website of Amazon. In order to extract the data from the site a python code is developed with the help of different python libraries, an amazon specific crawler is implemented which is created on Hyper Text Mark-up Language (HTML). It consists of various element called tags. The main content of the webpage is written between <body> <head>…. </head></body> tags. Beautiful Soup is a Python library, which is predominantly used here for pulling data out of HTML and XML files. It actually parses the document and extracts the exact text mentioned within the target tags.

B. Data Pre-processing

In Machine Learning, Data pre-processing is one of the most integral steps before applying algorithms. Because Machine Learning algorithms do not work with raw data, so text data needs to be cleaned and converted into numerical vectors. This process is called text-processing.

These are below basic steps that we are going to show you in this paper.

1) Understanding the data: First of all, you need to see what the data is all about and what parameters (Stopwords, Punctuation, html tag…. etc) are in the data.
2) Data Cleaning: In this step, we are going to discuss about how to remove parameters.
3) Techniques for encoding text data: There are lots of techniques for encoding text data. But below are the techniques I have mostly used while solving real-world problems.

a) Bag of Words
b) Bi-gram, n-gram
c) TF-IDF
d) Avg-Word2Vec

C. Data Preparation

At first, after loading the data we need to prepare it in such a way so that it could fit best into a model. In our dataset, there is a column named 'Score', containing values such as 1,2,3,4,5 which we consider as our target column. Our main object is to predict whether a review is positive or negative. Here, if we consider 1,2 as negative reviews and 4,5 as positive reviews then logically 3 does not add any value to our objective. So, we have discarded the rows containing value 3. After that, we have converted the score values into class label as 0 for negative review and as 1 for positive review. Thereafter, we remove duplicates and unwanted records based on data and also domain knowledge because it is one of the most 'state of the art' part in data science.

D. Text Pre-processing

Before starting text-processing we have explained about some topics such as Stopwords and Stemming which is necessary for text-processing.

1) *Stop words:* These are some unimportant words even if you remove them from sentences, semantic meaning of the text doesn't change. Example: 'This restaurant is good'(Here 'This', 'is' are Stopwords)
2) *Stemming:* It is a technique which can convert words to their base word or stem word (i.e. tasty, tastefully is converted into base word tasti ….). In our case we have used Snowball Stemmer.

There is also another technique called Lemmatization but here in our case, we have used only stemming process
Below are the steps that we have done for pre-processing:

a) Remove html tags.
b) Remove any punctuations and special characters
c) Convert the word to lowercase
d) Remove Stopwords
e) Finally we use Snowball Stemmer for stemming the words.

E. Techniques for Text Encoding

We have used below encoding techniques for featurizing our data.

1) Bag of Words (BOW)

In BOW, we construct a dictionary that contains a set of all unique words from our review text dataset. Here, the frequency of every word is counted. If there are d unique words in our dataset, then for every review the vector will be the length of size d. The vector will be very sparse in this case.

2) Bi-gram, n-gram

Bi-gram is basically a pair of two consecutive words used for creating dictionary, tri-gram is basically three consecutive words.
Scikit-learn CountVectorizer has parameter ngram_range, if it is assigned as (1,2) then it is called Bi-gram.

3) TF-IDF

Term Frequency -Inverse Document Frequency (TF-IDF) gives less importance to most frequent words and gives more importance to less frequent words.
Term Frequency is number of times a particular word(W) occurs in a review divided by total number of words (Wr) in review. The term frequency value ranges from 0 to 1.
Inverse Document Frequency is calculated as log (Total Number of Docs(N) / Number of Docs which contains particular word(n)). Here Docs referred as Reviews.
TF-IDF =TF * IDF =(W/Wr) *LOG(N/n)
Word2Vec:

It actually takes the semantic meaning of the words and their relationships between other words. It learns all the internal relationships between the words. It represents the word in dense vector form.
Here we import gensim library which has Word2Vec takes the parameters like min_count=5 means if a word repeats less than 5 times then it will ignore that word, size=50 gives a vector of length of size 50, and workers are cores to run this.

4) Average Word2Vec

To compute Average Word2Vec, below are the steps to follow.

a) Compute Word2Vec for each of the words
b) Add the vectors of each words of the sentence
c) Then divide the vector with the number of words in the sentence

It's a simple average of the Word2Vec of all the words.

III. RESULTS

Once we get the structured data after pre-processing, we have used Logistic Regression Classifier on text data in order to get the polarity of those reviews along with using unigram to n gram as a feature vector. we have also used Tf-Idf and Bag of words as other feature vectors along with this classifier.
A support vector machine is another supervised machine learning classification model, similar to logistic regression but a bit of advanced is also used as another algorithm for depicting the polarity of the texts.
Below table shows that the accuracy of applying Logistic Regression and SVM on test dataset with various featurization techniques.

Model	Featurization Technique	Test Accuracy
Logistic Regression	Bag of Words	90.03%
	Tf-Idf with one-gram	87.23%
	Tf-Idf with two-gram	89.15%
	Average W2V	91.42%
Linear SVM	Bag of Words	90.89%
	Tf-Idf with one-gram	88.02%
	Tf-Idf with two-gram	89.45%
	Average W2V	93.11%

A. Sentiment score Prediction

These prediction models have mainly focused on sentiment polarity like positive or negative instead of scores.
Predictive techniques like Logistic Regression and SVM are used to test the data. The below table is the representation of scores of some test data.

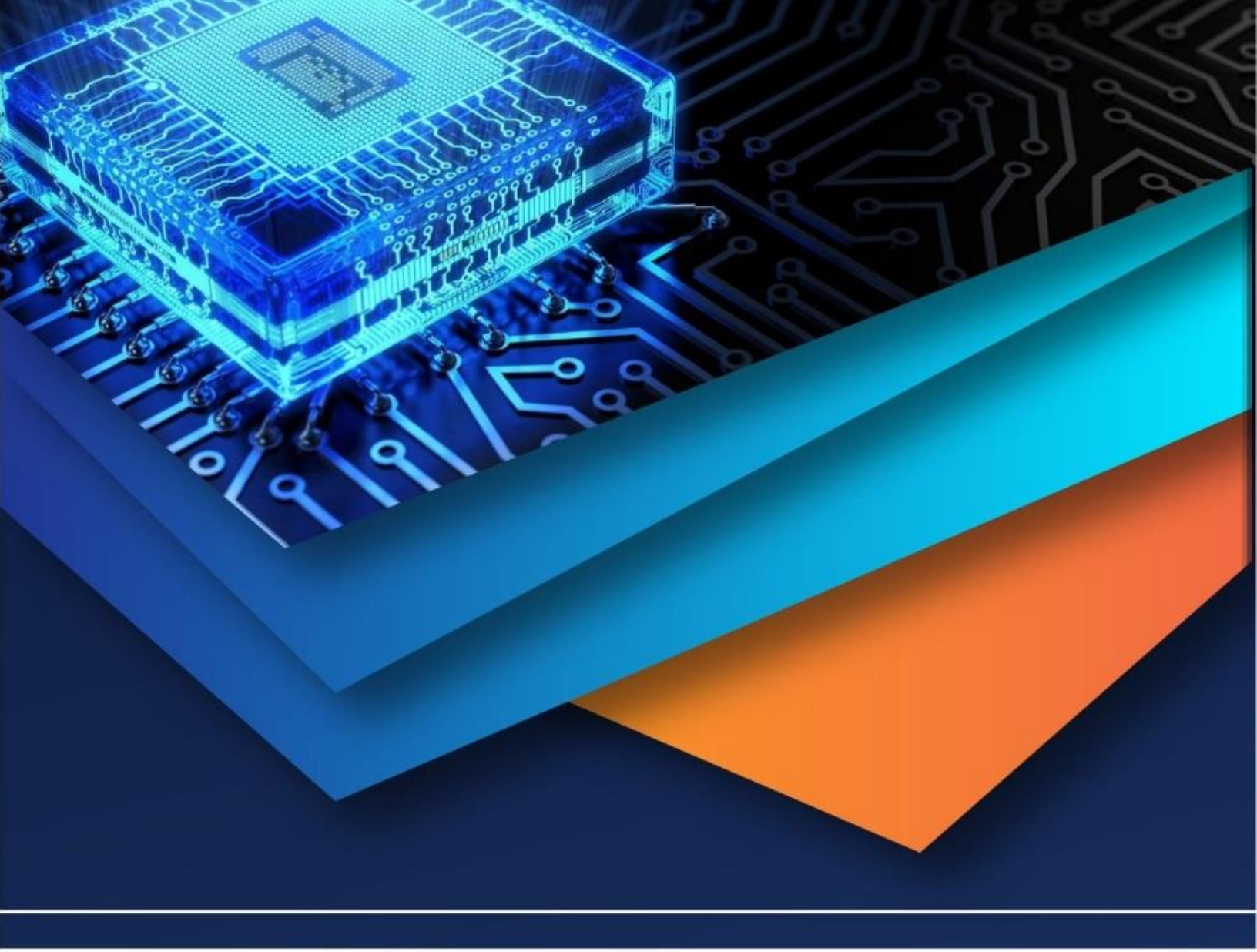
Score	**Summary**
5	Good Quality Dog Food
1	Not as Advertised
4	"Delight" says it all
2	Cough Medicine
5	Great taffy
4	Nice Taffy
5	Great! Just as good as the expensive brands!
5	Wonderful, tasty taffy
5	Yay Barley
5	Healthy Dog Food
5	The Best Hot Sauce in the World

IV. CONCLUSION

In this paper, we present different linear machine learning algorithmic approaches along with different featurization techniques for sentiment classification on Amazon Fine Food Reviews dataset. We have reached 91.42% and 93.11% test accuracies in binary class classification task while applying featurizing techniques like average W2V along with Logistic Regression and SVM on the test set respectively. In our experiment, we find that Average W2V on reviews proves more useful and Average W2V technique surpasses approaches without Average W2V that we implement for all models. Future work might focus on trying out on deep learning models like RNN model, like the bidirectional RNN and tuning other parameters like hidden layer size and number of steps.

REFERENCES

[1] Deepu S, Pethuru Raj, and S.Rajaraajeswari. "A Framework for Text Analytics using the Bag of Words (BoW) Model for Prediction" International Journal of Advanced Networking & Applications (IJANA).
[2] Kim, S. M., & Hovy, E. (2004, August). Determining the sentiment of opinions. In Proceedings of the international conference on Computational Linguistics (p.1367). Association for Computational Linguistics.
[3] Liu B (2014) ,‖The science of detecting fake reviews‖, http://content26.com/blog/bing-liu-the-science-of-detecting-fakereviews/
[4] Anjali Ganesh Jivani, ―A Comparative Study of Stemming Algorithms‖, International. Journal. Computer. Technology
[5] Drozd, N., 2016. Text Classification with Word2Vec – DS lore. Retrieved from http://nadbordrozd.github.io/blog/2016/05/20/text-classification-withword2vec/



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY