****

**SCHOOL OF ARCHITECTURE, COMPUTING AND ENGINEERING**

**Implementing Sentiment Analysis on Amazon Online Food Reviews**

Student Name: MIKKILI RAJKUMAR

Student Number: 1957373

Supervisor: MARY AUGUSTINE

Module Code: CN7000

MSc in Computer Science

# Declaration

This research project is my original work and has not been submitted to any other university for academic award.

Sign : MIKKILI RAJKUMAR Date : 08/08/2021

**Name: MIKKILI RAJKUMAR**

**Registration Number: 1957373**

This project has been submitted with my approval as the appointed University supervisor.

Sign: **MARY AUGUSTINE**  Date: **27/06/21**

# Abstract

With the exponential growth of user-created content like feedback, opinions and reviews on the products and services of the companies like Amazon and Flipkart, traditional systems have failed to process it. Sentiment analysis, the technique of analyzing if a text/review is positive, neutral or negative, is the new yet important trend in the extraction of user opinions. It helps to determine the people opinion about a certain product and its reception in the market, by just looking at the data. This project researches the different methods of sentiment analysis applied to the huge datasets of Amazon Fine Food reviews. There are three methods namely, Linear SVC, Logistic Regression and Naïve Bayes, which give an accuracy of more than 80% and run using matplot which is in the Apache spark Library for Machine Learning in Python. Upon deep testing on these three methods, it is realized that the Linear SVC performs better among the other techniques.

Table of Contents

[Declaration ii](#_Toc81758565)

[Abstract iii](#_Toc81758566)

[List of Figures: vi](#_Toc81758567)

[Chapter 1: Introduction 1](#_Toc81758568)

[1.1 Overview 1](#_Toc81758569)

[1.2 Problem Statement 2](#_Toc81758570)

[1.3. Aim and Objectives 3](#_Toc81758571)

[1.4. Statement of Purpose 3](#_Toc81758572)

[1.5. Research Question 3](#_Toc81758573)

[1.6. Overview of Methodology 4](#_Toc81758574)

[1.7. Rationale and Significance 5](#_Toc81758575)

[1.9. Researcher Assumptions 6](#_Toc81758576)

[1.10. Definition of Key Terminology 6](#_Toc81758577)

[1.11. Project Plan 6](#_Toc81758578)

[Chapter 2: Literature Review 8](#_Toc81758579)

[2.1. Overview 8](#_Toc81758580)

[2.2. Background 10](#_Toc81758581)

[2.3. Research in review helpfulness evaluation 12](#_Toc81758582)

[2.4 Research in text classification 14](#_Toc81758583)

[2.5 Related Work 17](#_Toc81758584)

[Chapter 3: Research Methodology 22](#_Toc81758585)

[3.1. Research Approach 22](#_Toc81758586)

[3.2. What is NLTK? 22](#_Toc81758587)

[3.3. Data pre-processing 23](#_Toc81758588)

[3.4. What are stop words? 24](#_Toc81758589)

[3.5. Count Vectorizer in Python 25](#_Toc81758590)

[3.6. Processing tool from keras 26](#_Toc81758591)

[3.7. What is a Support Vector machine? 30](#_Toc81758592)

[3.8. Random Forest Classifier 31](#_Toc81758593)

[3.9. Decision Tree Algorithm 31](#_Toc81758594)

[Chapter 4: Results 33](#_Toc81758595)

[4.1. About Python 33](#_Toc81758596)

[4.2. Results 35](#_Toc81758597)

[Chapter 5: Discussion 48](#_Toc81758598)

[Chapter 6: Conclusion and Recommendations 51](#_Toc81758599)

[6.1. Conclusion 51](#_Toc81758600)

[6.2. Recommendations 52](#_Toc81758601)

[Reference: 55](#_Toc81758602)

[Appendix 63](#_Toc81758603)

# List of Figures:

[Figure 1: Memory Status of the GPU used in the project 37](#_Toc81758668)

[Figure 2: Confusion matrix for Navie Bayes algorithm 40](#_Toc81758669)

[Figure 3: Confusion matrix for Support Vector Classifier. 41](#_Toc81758670)

[Figure 4: Confusion matrix for Decision Tree algorithm. 41](#_Toc81758671)

[Figure 5: Confusion matrix for Random Forest algorithm. 42](#_Toc81758672)

[Figure 6: Confusion matrix for Cat Boost algorithm. 43](#_Toc81758673)

[Figure 7: Data frame for all the models Accuracy Score. 44](#_Toc81758674)

[Figure 8: Bar graph which compare Accuracies of models. 44](#_Toc81758675)

[Figure 9: Data frame for all the models Precision Score. 45](#_Toc81758676)

[Figure 10: Bar Graph to compare F1 Score. 45](#_Toc81758677)

[Figure 11: Data frame for all the models F1 Score. 46](#_Toc81758678)

[Figure 12: Bar Graph to compare F1 Score. 46](#_Toc81758679)

# Chapter 1: Introduction

## 1.1. Overview

In marketing analytics and business, online customer ratings and reviews for the products and services play an important role. By processing the rating and review data, conclusions about a product/service quality and its standards can be drawn. Also, one can know how well the product did in the market or know the public talk of the product. These key insights help in creating new approaches and processes for improving the quality of products. Customers too can easily decide on purchasing a certain product or not, by the sentiment analysis made on the huge customer feedback data. Be it for marketing purposes, growing the business, improving the standards or knowing customer satisfaction, it is evident that analyzing feedback data is important. The traditional analysis methods like dictionary methods or rule-based analysis methods are quite able to get fair results(Dumbleton, 2015).However it have their caviars like requiring more manual work for creating features, not being able to consider word order, unable to process huge chunks of data properly, or unable to get automated (z\_ai, 2019).

Sentiment Analysis is a Natural Learning Processing (NLP) technique that determines whether the user opinion about a specific topic, is positive, neutral or negative. There are multiple ways like Linear SVC, Logistic Regression and Naive Bayes, through which a good sentiment analysis system can be implemented into the websites and applications. These methods use a huge amount of text data labelled as positive, neutral, or negative, to learn and predict the opinion of the new text based on the patterns developed for positive feedback and negative feedback using the training dataset labels. The data is collected from Amazon Fine food reviews which contain features of the review like unique product\_id, unique user-id, profile name, total users who denoted whether itis useful or not, upvotes and downvotes, rating, date and time, a summary and the text of the review.

After data collection, the data is vectorized after pre-processing it. With the help of various techniques and methods on Apache spark’s MLlib, different analytic methods are explored to achieve high efficiency and performance. Different Machine Learning models are created using three methods mentioned above using Apache Spark Library and trained. Then the models can predict and give out the score for different texts. All three methods are deeply analyzed for their performance after building an ML model for each and predicting the sentiment of the text(Reddy, 2018).

## 1.2. Problem Statement

For sentiment analysis to work properly, it must tackle certain problems that are usually seen are listed below:

* The primary issue with sentiment extraction is categorizing the given text at the whole article level, sentence level, or feature/attribute level, based on its polarity. At present, there is no high accurate automated domain-independent sentiment classification tool that does not require manually validated and labelled datasets(Rafea and Morsy, 2015).
* Texts which are not authentic and contain outliers or sarcasm, definitely impact the overall sentiment of the product. The methods should be able to deal with punctuation, helpfulness, negation, sarcasm/irony and subjectiveness of the text.
* Finally, the context-sensitivity, domain-dependency, feature-dependency, topic-sentiment interaction of the words can altogether change the sentiment of the text(Gokce, 2020).These are some of the issues with sentiment extraction that need to be addressed.

## 1.3. Aim and Objectives

**Aim**

To conduct a study on the online food reviews through sentiment analysis

**Objectives**

* To gather numerous data for the literature review part.
* To know more about sentiment analysis.
* To choose the machine learning algorithm for researching to get the desired output.

## 1.4. Statement of Purpose

The statement of purpose this project holds is the betterment of sentiment extraction systems so that people who buy online can easily make their purchase decisions and businesses that sell the products can improve the products. This betterment is achieved by implementing the most accurate method after comparing the methods which have accuracy above 80%.Writing down an efficient sentiment extraction algorithm can be helpful as it can be used for monitoring the social media or a brand, analyzing the customer service, analyzing the feedback of the customer and researching the market(Roldos, 2020)

## 1.5. Research Question

Before choosing the method for sentiment extraction, a few questions that are given below must be answered:

* Are the traditional sentiment extraction techniques capable of processing huge data and get automated?
* How can be Machine Learning used for sentiment analysis?
* What are the different types, levels, techniques and methods of sentiment analysis using ML?

## 1.6. Overview of Methodology

There will be 5 stages namely acquiring dataset, data preprocessing, extraction of features, implementing the machine learning sentiment analysis methods through Spark MLlib, and evaluation of models through training and testing.

According to (Mcauley and Leskovec, 2013), Amazon’s Fine Food dataset is used for training the models. The dataset is then preprocessed before training the models to provide the best input to the models. The preprocessing includes necessary steps for cleaning of data like removing null and duplicate values and eliminating irrelevant datalike non-alphabetic characters, digits, special characters, punctuation marks and words that are common and have little or no significance like prepositions and conjunctions. Then data visualization and exploration are done using matplotlib and seaborn libraries to create a correlation matrix among features. Tokenized and normalized text data is converted into feature vectors using the term Frequency–Inverse Document Frequency(TF-IDF) so that models can be trained efficiently. TF-IDF is used in feature extraction in which TF finds the frequency of words and IDF finds the frequency of common and rare words.

Methods with more than 80% accuracy like Logistic Regression (LR), Linear Support Vector Classifier (LinearSVC), and Naïve Bayes (NB) are selected in this project.

LinearSVC is the technique that gives the best hyperplane that classifies the data. It has adaptable execution of SVC with the linear kernel (Awan, Rahim, Salim, *et al.*, 2021). Naive Bayes (NB) is a Bayesian theorem that is based on the supervised machine learning technique. The naive Bayes model is built by fixing the distribution of every feature(Al-Saqqa, Al-Naymat and Awajan, 2018) and(Mazhar Javed *et al.*, 2019). Lastly, logistic regression understands feature vectors of the variables and learns the coefficient for input expressions and then extracts the text sentiment(Prabhat and Khullar, 2017) and(Ali *et al.*, 2019). These models are built using Apache spark Library in Machine Learning in Python.

## 1.7. Rationale and Significance

The huge volume of data present on the web can help customers and organizations in making their decisions properly. But analyzing and understanding the combined opinion of others is a challenging task. It is impossible to find the sentiment sources, draw out the sentiment and render them in a definitive format manually. So, it is required to automate this procedure and sentiment extraction using machine learning is the need of the hour in automating this process(TangHuifeng, TanSongbo and ChengXueqi, 2009).

Dave et al. first noticed the term ‘opinion mining’ in a paper that defined an opinion mining tool. The terms opinion mining and sentiment analysis are interchangeable(Dave, Lawrence and Pennock, 2003).

Recently, there is a significant amount of interest and growth in the domain of sentiment analysis. Apart from mere product review analysis, the usage of sentiment analysis can be extended to medicine, finance, politics, calamities, software engineering and cybercrimes. Sentiment Analysis has come a long way from interpreting online product feedback to exploring the stuff on social media like Twitter and Facebook.

**1.8. Role of the Researcher**

The role of the researcher is to do extensive research on sentiment analysis and the different techniques for extracting the sentiment from the text. Then the researcher should collect data from Amazon reviews and select the most accurate methods for implementing the sentiment analysis on them. Lastly, the researcher should conduct an extensive examination of the implemented methods and produce a result.

## 1.9. Researcher Assumptions

Assumptions for this project are:

1. The secondary data collected from the journals, research papers and articles for the research is true.
2. The data collected from Amazon Fine Food Reviews is authentic and accurate for training.

## 1.10. Definition of Key Terminology

**Opinion Mining:** Another term for sentiment extraction.

**Linear SVC:** Linear SVC (Support Vector Classifier) is an NLP technique in sentiment analysis that uses the best fithy per plane to divide the data and predict.

**Naive Bayes:** It is an NLP (Natural Language Processing) technique used in sentiment analysis that extends Bayes theorem and uses it to predict.

**Logistic regression:** It is a technique that uses a logistic function to solve the binary classification problem.

**Python:** An object-oriented programming language.

## 1.11. Project Plan

This project focuses on analyzing different sentiment extraction methods using machine learning and improving sentiment extraction with the help of the most performing and accurate method. Hence, the project starts with a certified customer rating the product and writing its review or sharing thoughts about it. Similarly, other users also rate and review the product.

During the initial phase, all such reviews of the product are collected from Amazon fine food reviews as plain text. After collecting datasets, they are preprocessed thoroughly. Then, features of the text are extracted and parsers for this text are also implemented during this stage.

Existing NLP techniques for sentiment analysis are studied by reviewing relevant literature. Techniques or methods with high accuracy are selected so that they can be analyzed for their performance. During the design phase, models for the studied and selected NLP techniques are designed. During the development phase, coding for designed models is implemented and models are developed. With the train data set, these models are trained in this phase only. Now, the models are ready to be tested. During the testing phase, models are tested with test data set and evaluated based on their accuracy and performance. By the end of the testing phase, sentiment extraction techniques should have been implemented and evaluated. Based on the performance evaluation of different sentiment analysis techniques, an effective and most accurate one is used to implement the sentiment extraction tool. The resultant tool built with the most accurate NLP technique is ready to be deployed. After all the building phases, a complete report of selected machine learning techniques will be prepared and submitted.

# Chapter 2: Literature Review

## 2.1. Overview

Customers on Amazon use the platform's world-leading capacity for e-commerce, and Sentiment Analysis is a methodology that collects information about customers' feelings by utilizing natural language processing, which is dependent on the way the text is arranged. Amazon is the world's largest online shopping platform for customers. When a buyer purchases an item from an online platform, sentiment analysis can assist them in making informed decisions. The customer's opinion, which might be either good or negative regarding the product, is being analyzed using sentiment analysis. With the help of natural language processing, sentiment analysis is a branch of data mining that is used to anticipate a customer's opinion or emotion from an online platform such as "Amazon". By employing this method, it is possible to assess the level of demand for the item among customers, which in turn allows the organization to simply increase the sales of its products. It is possible to comprehend and forecast the emotions, feelings, and views of consumers who are witnessing a product with the information included in a text format through the use of sentiment analysis. Organizing a big amount of data that includes the customer's opinion can be difficult, but it is possible to predict the outcome, which subsequently benefits both the customer and the internet platform. When it comes to increasing product sales, brand sentiment analysis is critical for both sides, that is, the company and the customer. According to the firm, understanding the genuine behavior of the consumer regarding the sale of products would allow them to make a significant amount of money. Customers continue to play a vital role in the organization, and clients can express themselves more freely on the internet platform, thanks to the company's extensive internet presence. When it comes to increasing product sales, sentiment analysis is necessary, and the benefit comes from the customer's wise selection of the goods. For better or worse, reviews, good, bad, or neutral, can be predicted by an increase in the product's quality, an increase in the product's quantity, or an increase in the amount of that product that gets sold. Through the use of sentiment analysis, the selling platform can increase client traffic by providing high-quality services and also improving the product's overall quality. Reviews of Amazon products may be of assistance to customers considering the same product and may also assist customers in making more informed purchasing choices on the site. While natural language processing algorithms that apply sentiment, analysis can parse and extract complicated characteristics, they do so only if deep learning methods are also employed. This is relevant in a big way to us since deep learning is the key technology, one will make use to pull information from the internet, and the volume of data involved is massive. Deep learning is used to extract complicated features from large amounts of data, and it can also be used to create patterns from large amounts of data. When users ‘after purchasing the goods’ submit favorable reviews on the platform, the platform is considered successful. If the reviews are positive, then buyers will be more confident in purchasing the product, and the company will be able to quickly improve its commercial operations. Machine learning techniques include LR, SVM, RF, NB, and K mean, as well as supervised techniques such as DT, NB, SVM, RF, LR, LR Apriori, and unsupervised techniques such as K-mean, Apriori Association, clustering, and others. Deep learning techniques such as CNN, LSTM, RNN, RBFNS, and supervised learning are also included. Customer’s product reviews are predicted through feature engineering, which increases the amount of traffic to the product's Amazon page as a result.

**Signifies:**

Sentiment analysis is used to determine the positive, negative, and neutral aspects of a piece of text. Opinion mining is used to extract from the text that is about a particular brand of goods. Sentiment analysis is a sort of data mining that incorporates the use of natural language processing to determine the procedure preferences of customers' assessments, which are kept in a database. To increase the amount of traffic to the website, an extract of consumer opinions from an online platform and other sources has been used, among other things. It is a widely used text analysis tool that has frequently relied on data from a variety of sources, including an online system that collects consumer reviews of the product, as well as deep learning and machine learning approaches.

## 2.2. Background

Amazon is an American multinational company controlling consumer products, sales and production. If things can be purchased on the market, purchasers can rapidly buy these products and exchange information via Amazon. American innovations such as artificial intelligence, cloud computing, and information technology all take place in the context of internet platforms. It was an interesting and exciting experience for user at Amazon (Jeff Bozos in Bellevue, 1994). Seattle is chosen because Microsoft was headquartered. Amazon was first made available for public usage in 1997. The product was first sold as music and video in 1998, marking the beginning of its commercialization. Organizations that provide cloud computing services, web hosting services, and internet patterning have grown increasingly prominent since 2002(Quyyam and Ghous, 2021). In 2006, the organization was granted permission to accept virtual payments from customers and to retain data on the internet. During the 2012 fiscal year, the organization purchased a firm that develops mobile robotics. In the beginning, Amazon was based on an online system that provides customers with products in the market. They included literature, food, cinema, sports equipment and toys, as well as jewellery. Compared to the 2015 "Wal-Mart, Bigbox-Store and Supermarket," in which they delivered a big discount, along with food products for the consumer, the breadth of Amazon was wide. In 2017, Amazon purchased the food business stock for a total of $13.4 billion. In 2018, the worldwide most popular Amazon Prime for two days of goods services. It is the largest internet system in the world and also provides cloud computing and streaming services. The pay-back payment system was introduced and product reviews, books, meals, service providers, and many other items were submitted by Amazon Prime. The Amazon Web Services business is the world's largest internet company in terms of revenue. Amazon has surpassed Apple as the most valuable brand on the planet in 2020. Amazon will establish two centers in Italy in 2021, creating 1100 jobs for the locals while also investing 278 million dollars in the country's infrastructure. Amazon has acquired the online shopping system that was formerly used for the sale of a variety of different products by third-party vendors. Shellfire, which was recently acquired by Amazon.com, was also an online shopping platform used in the sale of clothes and books (Quyyam and Ghous, 2021).

Machine learning, lexicon-based, and hybrid techniques are all used in sentiment analysis. In supervised classification, machine learning algorithms are put into action. The collecting of opinion terms is necessary for a vocabulary-based strategy. A hybrid strategy uses both lexicons and machine learning. Natural language processing refers to the computer techniques that involve the use of languages. The majority of currently accessible research is concerned with data mining from publicly available sources on the internet. Before the advent of the World Wide Web, information or opinions on products were gathered manually through surveys. (Fang and Zhan, 2015) presented a study based on product reviews obtained from Amazon to discover negation terms in the reviews. The classification of data at the sentence and review level is carried out for the data acquired between February and April 2014.Using evaluations of the iPhone 5 obtained from the Amazon website, Bhatt and Patel Gawande, (2015) proposed a rule-based extraction of product feature sentiment analysis for use in future research. The POS technique is used at each sentence-level and results are shown in charts and tables. Kamal(2013) mined opinions from internet product reviews using supervised and rule-based procedures, according to the findings.

## 2.3. Research In Review Helpfulness Evaluation

The majority of previous research on evaluation helpfulness has focused on automating the assessment helpfulness prediction to handle the issues relating to the large and uneven quality of online customer reviews currently being tackled. A great deal of effort has been done using the use of text mining and statistical modeling methods to identify relationships between review assistance and text-based variables. The connection of sentiment and the value of online consumer evaluations were discussed by Salehan et al. (2016). The study examined the effect of feeling polarity on the utility of online reviews and produced some interesting results. Their research model incorporated variables that were related to both the title and the review of the study. Aside from the polarity and attitude of the reviews, the length and longevity of the reviews were the other major elements taken into consideration when developing the model. Considering that websites such as Amazon.com prioritize reviews based on their usefulness rather than their date of publication, it seemed reasonable to include longevity among the characteristics considered in the study. Reviews with neutral polarity, as opposed to positive and negative ones, had a substantial impact on helpfulness, it was discovered. As well as this, researchers discovered that the length of the review was a key effect, as they expect a larger review to have more information. When it comes to another study, Cao et al. (2011) focused on the subject of why some evaluations receive higher ratings for helpfulness than others. To this end, CNET data Download.com has been used to analyze the impact on the number of helpful votes received on basic, stylistic and semantic components of online reviews. This work was carried out using the Ordinal Logistic Regression model, which was defined in three parameters: the error rate, the AIC (Ak like information criterion), and lift ratio. It was concluded that semantic traits were more influential than other characteristics and that examinations with extreme opinions were more useful than examinations with moderate viewpoints. Liu et al. (2008) developed and tested a model of non-linear regression based on variables derived from IMDB movie reviews. The committee took into account aspects such as competence, writing style, punctuality, review duration, the polarity of the review and an overall average rating. The proportion of the number of users who found the review to be valuable to the number of users who voted whether or not the review was useful for assessing the helpfulness measure. The characteristics that were regarded as the most relevant were, among others, the experience of the reviewers, the writing style, and the timeliness of the submission. With the help of the MSE score, these characteristics were quantified and incorporated into the model, which allowed us to evaluate their individual and combined effectiveness. When the characteristics were combined, the best results were obtained.

When Kim et al. (2006) designed their study, to provide review authors with quick feedback, they concentrated on making an instant evaluation of the usefulness of the revision. It was discovered that utilizing SVM regression modelling, they were able to investigate the effect of various text-based variables on the helpfulness of a review. Among the characteristics were lexical and structural characteristics, syntactic characteristics, semantic characteristics, and meta-data. The duration of the evaluation, the number of unigrams in the examination and the assessment that the examiner has given the product has been the most useful factors. Ghose and Ipeirotis(2007) developed ranking techniques to swiftly discover underlying useful reviews in online market platforms to cope with the difficulty of reviewing a huge number of evaluations on online market platforms. This rating process included both user-oriented and manufacturer-oriented approaches, and it was a hybrid of the two. The usefulness of the review was determined by the number of helpful votes it received while the usefulness of a review was judged by the number of sales of the product in the users-oriented approach. Using regression modelling, researcher was able to do an econometric and subjectivity study on the Amazon.com panel data set. Researchers discovered that reviews that had a combination of subjective and objective features received higher ratings for usefulness.

## 2.4. Research in Text Classification

This section offers some insight in other areas than review aid in the work being undertaken in the areas of text mining and natural language processing (NLP). Several text-based features and machine learning methods can be applied in the realm of spam review identification. Furthermore, the usage of Support Vector Machines in the text categorization problems is emphasized in this part. Crawford *et al.*(2015) have done a spam detection survey in which features retrieved from text through natural language processing are examined and several review approaches are compared as ‘spam’ or ‘non-spam’. In the study's lexical and semantic variables were studied, including the bag of terms, term frequency, point of sale tagging, and other lexical and semantic data. To finish the classification process, methods such as SVM (Support Vector Machine), Nave Bayes, and Logistic Regression, among others, were used.In comparison to the other algorithms tested, SVM consistently outperformed Naive Bayes and Logistic Regression, while also being rarely outperformed by them. Furthermore, it has been demonstrated that the performance can be improved by combining various features (Crawford *et al.*, 2015).

Shojaee *et al.*(2014) developed stylometric features to distinguish the writing style of spammers as part of their investigation on detecting misleading perspectives. Spammers attempt to alter their writing style and linguistic use according to the findings of their study by using simpler, shorter, and less mean syllables per word as a base. The characteristics were divided into two categories: lexical characteristics and syntactic characteristics. Some of these characteristics are the total number of tokens, the average sentence length, the average token length, and the number of occurrences of uppercase characters, to name a few. Support Vector Machine and Naive Bayes was the machine learning classification techniques that were used. The F-score measure was used to compare features that were assessed independently as well as together. The highest accuracy was obtained when the features were merged regardless of which algorithm was used. Furthermore, when compared to Naive Bayes, SVM outperformed it in all combinations of the characteristics. Research has also shown that the combination of semantic and word frequency information in some cases can surpass anyone employed alone. A text classification investigation was undertaken by Lilleberg, Zhu and Zhang(2015), in which word vectors obtained from the Word2vec approach were integrated into the TF-IDF measurement of every word. The concept was that the TF-IDF matrix did not capture the semantic linkages between the tokens Word2vec can offer accurately. Aside from that, several combinations of stop words were tested out to see what worked best. The study used the text data from the "20 newsgroups," which were the classification algorithm Linear SVC. In combination with TFIDF without the stop words, Word2vec produced the maximum accuracy of all possible combination testing, other than a few unique cases in which TF-IDF cannot exceed without the stop words. The Naive Bayes and Support Vector Machine techniques are superior for text classification jobs when it comes to machine classification algorithms (Vinodhini, 2012). Theoretical proof as to why SVM works for text categorization may be discovered according to Joachims(1998). In an attempt to strengthen the viewpoint, the following justifications were offered:

* Because its functioning is not dependent on the dimensionality of the problem, SVM can handle the huge feature spaces associated with text categorization challenges.
* Categorisation of texts is an example of a problem, in which SVM is best-suited for sparse occurrences and dense concepts, such as text classification (high dimensionality in few non-zero entries).
* Almost all text classification problems tend to be linearly separable, and the goal of the SVM is to find linear separators for these problems.

Different text-based elements have been identified in the literature as having been employed in text mining investigations, which is consistent with the findings of the literature. While there has been some progress in understanding reviewer helpfulness. Rather than studying and understanding reviewer centric aspects such as review length and polarity, as well as review subjectivity and writing style, the majority of the research has focused on evaluating and comprehending reviewer centric features such as the reviewer's written style. To that end, this study set out to uncover critical features that were highly associated with the review helpfulness measure, to rank reviews according to their perceived usefulness. Even though word features based on embedding were utilized in other domains, they were mostly unexplored in the subject of review utility. The effectiveness of vectorized features and word-based features in forecasting the usefulness of reviews is still not comparable to the results of content-based reviewing features in the best possible way. The present work also includes a comparison of the performance of an extremely randomly randomized ensemble classifier with the performance of established approaches for text classification such as supporting vector machines and logistic regression(Sharedalal, 2019).

## 2.5. Related Work

Text sentiment analysis is traditionally performed using word counts or frequency counts in the text that is assigned a sentiment value by a trained expert (Pang and Lee, 2008). In these systems, the order of words is not taken into consideration. For sequence labeling on sequential data of variable length, recurrent neural networks (RNNs) can be employed(Mikolov *et al.*, 2010). This is particularly useful for sentiment analysis applications in which the input sentence can be considered as a sequence of tokens. On the topic of sentiment categorization, recent research has investigated the Gated Recurrent Units (GRU)neural network (Chung *et al.*, 2015). LSTM neural network architecture is a special case of GRUs, which are a type of long-short-term memory network. In this activity, GRUs is particularly effective due to their capacity to recall long-term dependencies. Besides that, GRUs are significantly faster in training and convergence than LSTM networks.

There has been a lot of recent work on more intricate RNN models, such as the recursive neural tensor network, which is robust in recognizing negating negatives, and the tree LSTM, which incorporates the LSTM's forget gate concept (Socher *et al.*, 2013 and Tai, Socher and Manning, 2015). Looking at the accuracy of the previous year's project, Yuan and Zhou, (2015) states that it ranged from 59.32 percent to 63.71 percent, depending on the Recursive Neural Network model used in the calculation. They created vanilla one hidden layer recursive neural networks, two hidden layer recursive neural networks, and recursive neural networks with a hidden layer on top. Due to the use of a better tree parser and amplified labelling approaches within internal nodes. Meanwhile, Stanford TreeBank, as a result of the rigorous supervision, which is to say, the meticulous labelling of internal nodes, obtained extremely high test accuracy (more than 80 percent). Until now, it is the most effective data set for usage in Recursive Neural Networks. On the other hand, one might suppose that the absence of labeling is one of the major issues facing Recursive Neural Networks. Returning to the Kaggle competition, although there is no current winning accuracy at this time, the data set and question were derived from research published by Stanford University in 2011(McAuley and Leskovec, 2013). Their best test accuracy was approximately 40% in their investigations of consumers' tastes and preferences changing and evolving, even though sentiment analysis was not the primary problem in their study. This poor accuracy also indicated that this was a difficult data set to interpret based on the available information.

Scientists are actively engaged in the study of sentiment analysis, which has risen to become the most important area of research in recent years. Sultana *et al.*(2019) explained that sentimental analysis contains three significant aspects, which are a positive aspect, a negative aspect, and a neutral aspect. Since the last few years, the internet has emerged as a major factor in the evaluation of products and services. Users can post their thoughts, which can be good or negative, on social media and e-commerce websites such as Facebook and Twitter, and these comments can be used to help make judgments about adopting new ideas and purchasing new products. For product reviews, developed a new technique for removing the characteristics of sentiment analysis(Chen *et al.*, 2018). While reading product evaluations, categorize the sequences of feature vectors using clustering methods, which allows to retrieve the most prevalent TF-IDF vectors by utilizing the same form of synonyms as before. It is possible to refine span algorithms for pseudo consecutive phrases with FPCD having word order details by using this technique. With the help of the last few steps, one can collect the text feature. To improve the overall performance, it is necessary to employ several mechanism types. According to Abbas, Ali Memon and Aleem Jamali(2019), when it comes to specific instances, the authors proposed an entirely new heuristic method, as well as a naive bias for those situations. Unsupervised text classification and sentimental analysis are two applications of the MNB classifier, which is a form of NB classifier.

It is proven that the algorithms are efficient based on the results of large data sets. These tactics were used in Twitter sentiment analysis in the study by Neethu and Rajasree(2013), in which the authors investigated Twitter messages by utilizing machine learning algorithms for different products such as a mobile phone, a tablet, a laptop, etc. It is simple to investigate the major repercussions of sentiment analysis while using sentimental analysis techniques.

Some complications can arise, and feature extraction can be performed in two stages following preprocessing to resolve these issues. In the first stage, features are first removed from tweets and then features extraction is performed before being added to the feature vector. Feature classification is accomplished by the use of classifiers such as NB, SVM, and maximum entropy. Indra, Wikarsa and Turang(2017) created web-based apps that categorize tweets from netizens into four different types of machine learning methods applied, referred to as logistics regression, and then classified the tweets into four categories. Extraction methods for tweets, text features, and machine learning approaches are primarily divided into four categories. To create the training dataset for this study, 1800 tweets were collected. The use of real-time processing techniques such as URL transfer, punctuation and stop words, as well as tokenization and stemming, can be advantageous. The set of characteristics that are employed in logistic regression approaches to classify objects. Using a confusion matrix, it is possible to obtain high-efficiency tweets in roughly 92 percent of cases. In the authors employed various sentiment analysis algorithms to a movie review, with the results being published online(Wawre and Deshmukh, 2016). It was determined that Naive Bayes is the most efficient and performs better than SVM when the two techniques were applied to movie reviews. They concluded that SVM was less efficient and performed worse when the two strategies were applied to the same movie reviews. If Naive Bayes is used on training datasets that contain a large number of reviews, one can get the most accurate and reliable results. Customer reviews from Trip Advisor were categorized by the writers in Laksono *et al.*(2019), which is the best restaurant in the port city of Surabaya. In addition, decide the way of examining restaurant customer reviews by executing and differentiating both Naive Bayes and Text Blob analyses on the data. However, Naive Bayes produces the most efficient and valid output when compared to Text Blob, even though both techniques work admirably. Liu *et al.*(2013) employed Naive Bayes classifiers on large datasets, scaling them out as they went. Exploration mechanisms should be applied with NBC for the best performance. The big data analysis system was employed. The precision level of NBC grew as a result of this approach, and the network gained an additional 82 percent. NBC is used to investigate the opinions of reviews with a high level of productivity. By utilizing Apache Spark, Khullar and Prabhat, (2017) were able to undertake an evaluation of the SA based on user reviews. The Apache Spark framework, which is an extendable framework, is employed, and several methods from the MLlib library are applied, such as Naive Bayes, SVM, and logistic regression. Following the application of these approaches, it’s concluded that SVM is more accurate and reliable than Naive Bayes and logistic regression and that it produces the most precise results. To predict the stock market and Black Friday sales effectively Awan, Rahim, Nobanee, *et al.*(2021) employed big data and the Spark ML framework in their recent studies from the year 2021.

Mcauley and Leskovec(2013)states customer opinion on a product is predicted via sentiment analysis, which is performed on the customers' responses. Pre-processing techniques were applied to the customer evaluation of the product to tokenize, remove stop words from it, punctuate it, lower case it, stem it, and lemmatize it. Some of the machine learning algorithms that have been applied there include multinomial NB, Bernoulli NB, LR, SGD, Linear SVM, and RF, among others. One dataset came from Amazon book reviews, while the other was a crawl of the IMDB movie review site. The two datasets were combined to form the final product. Matthew associations' top findings were 0.84364 of average precision: 0.88930 and 0.92332 high precision: 0.8893 and C=2. The results were Linear SVM, Amazon. Increasing precision will be monitored by financial, political, and social networks in the future, and they will be able to see how parameters are optimized and utilized.

Malik and Hussain (2018) said based on the sentiment research, it was possible to estimate what the customer would think of the product. It is possible to extract reviews using a pre-processing technique that removes stop words, quotes, and characters. It was decided to use different machine learning algorithms, such as the Bayes net, the Naive Bayes multinomial, the SVM, and Multilayer Layer. An analysis of the data set revealed that it had the highest degree of precision. 91.1696 percent of the time, the naive Bayes multinomial was accurate. When it comes to pre-processing and attribute selection, different methodologies will be applied in the future.

Chapter 3: Research Methodology

## 3.1. Research Approach

The arrangements and the action taken in the field of research for a particular period and the steps are taken according to board assumption in terms of collecting data, evaluation and interpretation is known as research approaches. The idea carries several decisions and these are not to be considered as the order in which they make understanding and the order of their presentation in front. The whole opinion considering a particular approach taken in use should be analyzed first. This research brings a philosophical corner of assumption, action of audit, and particular researched method for data collection, evaluation and interpretation. The research approach is chosen based on the quality of research problems and must be solved by the researchers according to their skills. Three main points consecutively serve research by expanding and compressing the action of the method, they are research approach, research design, methods of research(Sagepub, 2015).

## 3.2. What is NLTK?

NLTK stands for Natural Language Toolkit. It is a typical kind of library for python having preassembled objectives and services for effortless implementation. It is the most commonly used library for natural language altering and calculation of syntax.

**NLTK Installation Process**

It can be easily installed in windows as well as ios having python pre-equipped.

**Accessing a dataset in NLTK**

A data set is specifically a corpus in nltk. The term corpus is defined as a bunch of pre-written words, sentences that act as an input. After taking inputs the corpus gets limited by breaking accordingly and afterwards gets processed. Many of them have downloaded bi the cited steps, but only used movies \_reviews corpus for the explanation purpose.

## 3.3. Data pre-processing

The process of preparing the machine for easy acceptance and figuring out the input in a more catchy way is known as Data pre-processing. Some typical actions for data pre- processing are(Great Learning Team, 2020):

**1.Tokenization**

Tokenization is a method of tearing the text according to the usage or demand.

The submodule of nltk “tokenize” will use.

* **Word tokenization** *-* The process of tearing a particular sentence into words is known as word tokenization, word \_tokenization is the function used in this process. As an output, get a list of words.
* **Sentence tokenization** - The process of tearing a corpus in a sentence is known as sentence tokenization. Sentence\_tokenization is the function used in this process. Generally, it is used when the input is paragraphs. As an output, several sentences from the paragraphs.

**2.Punctuation Removal**

Punctuations are not used frequently in NLP so they get removed.

**3. Stop words removal**

The words that appear frequently are known as stop words. It gets removed from the corpus or the normalization of the text.

**4. Stemming**

The reduction of articulation from words is known as stemming. Words with the same origin get removed. NLTK has a stemmer that implements other methods.

**5. Lemmatization**

It is another process of removal of articulation from the words. It differs from stemming in a way that it removes words to their origins having meaning.

**6. POS Tagging**

The recognition of part of speech from the sentence is dyed under the POS tagging process. It can identify nouns, pronouns, adjectives etc.in a sentence and allotment a POS tag to words. There are several methods of tag allotment but generally use a universal method of tagging.

**7. Chunking**

This method helps POS tags to gain insights from it. It is also known as shallow parsing. It is operated on the grounds of predefined rules by keeping certain groups of words. Then for phrase creation, the text is prased accordingly by the rule of group data(Great Learning Team, 2020). The method of converting data, it will be useful for the machine. Useful in a way that a computer can get something understandable from it to kick out the unnecessary data.

## 3.4. What are stop words?

These are intentionally programmed words, frequently used to avoid the indexing entries as well as in retrieving them as the cumulative result for the general query. The general example of stop words is ‘the’, ‘a’, ‘an’, ‘in’ and many others.

These words take up a place in the database, resulting in slowing down the speed of processing, taking more and more time. To overcome this issue will save the list of words as a stop word, NLTK has the solution to this problem, in python, and they have a list of words in many different languages. It can be easily found in the data directory of nltk(geeks for geeks, 2021). home/pratima/nltk\_data/corpora/stopwords is the directory address.

**Port Stemmer**

The process of creating the morphological variants of a basic word is known as stemming. It is generally referred to as algorithms or stemmers. It works in such a way that the words ‘chocolates’, ‘chocolatey’, ‘choco’ to the root word, ‘chocolate’ and ‘retrieval’, ‘retrieved’, ‘retrieves’ reduce to the stem ‘retrieve’(GeeksforGeeks, 2021).

In Natural Language Processing stemming and lemmatization is used as a text normalization tool. Language processing typically uses text, words and documents for an ongoing processing. These algorithms of stemming and lemmatization are developed in the late 1960s (Jabeen, 2018).

## 3.5. CountVectorizer in Python

Tokenization is a process of removing certain words which are prased to use textual data for predictive modelling. The words are getting used as inputs in machine learning algorithms, only the ecodement of integers or floating-point values is needed. The process is called future extraction or vectorization.

The conversion of text documents to a vector in terms of token count can be done by Scikit-learn’s count vectorizer. It also gives the option of text data before making the vector form. These characteristics make it a highly elastic feature presentation module for text.

It is a tool provided by the Scikit-learn library. On the grounds of frequency, it converts the text in vector order for each word that is considered in the entire text. It helps when have the numbers of text, then one need to convert each word in each text in the form of a vector (Edpresso Team, 2015).

It also creates a matrix in which every kind of different word serves as a column of the matrix and every particular text sample extracted from the document serves as a row in the matrix. The value of the cell can be defined as the court of the word in that specific text sample.

Some of the points that need to be observed (GeeksforGeeks, 2020c):

* It contains 12 exclusive words in the document characterized as a column of the table.
* It contains 3 text samples in the document, each characterized as a row of the table.
* Each cell was accommodated with a number that characterized the count of the word in that text.
* All characters can be converted to lowercase.
* The characters are arranged alphabetically.

## 3.6. Processing tool from Keras

The pre-processing of the Keras dataset operates on tf.data.Dataset. With the help of this raw data on disk to a tf.data.Dataset parts can be taken into use purposely to train a model. For instance, if 10 folders are there then every folder contains 10,000 different types of

pictures of various categories(Keras API, 2015).

**What is the train test split?**

In the sklearn model, there is a function named train test split in which selection of arrays for splitting data of two subjects takes place. For training and testing the data have to be separated manually.

Sklearn train tests also do random separation of the data into two subsets. It also has various kinds of parameters. For example: Basic syntax are like

train\_test\_split(X, y, train\_siize=0.\*,test\_size=0.\*, random\_state=\*)

In which the first parameter selected X, Y for the use.

* train\_size - This framework sets the size of the dataset under training. In which three options are given: None, is for the back out, Int, is for an exact number of trials, and flot is for the ranges limit from 0.1 to 1.0.
* test\_size - This framework states the dimensions of the testing dataset. The back out-state suits the trial size It will be set to 0.25 if the trial size is set as backout.
* random\_state - The back out mode performs as a random split using np random. The option is given to add an integer using an exact number side by side.

**Why use the Sklearntrain\_test\_split function?**

This data set while using for both training and testing gives room for miscomputation, due to which the chances of improper predictions. It gives the chance to break the data set very easily while on the ideal model. Things to keep in mind are that the model should not be under-fitting or overfitting.

**Overfitting and underfitting**

In the overfitting situation model presenting is perfectly appropriate, at the time of data handling. The conditions take place only when the model has a complicated bunch of rules. At the time of handling new data, it may give inaccurate results. The underfitting model does not fit due to bunches of rules. This model is not reliable.Comparatively less reliable.

**Points to remember while dealing with Train\_test\_split**

* Using train\_test\_split in place of random\_state function can distribute arrays into random bunches or sub bunches.
* On the count of ratio Ideal split are 80:20 for the trail and analysis. Only simple adjustments are needed dealing with the size of the data set and parameter complications (Bitdegree, 2020).
* Standard Normal distributions are followed by the StandardScaler. That’s why it makes mean = 0 and scales the data on unit variance.
* Minmaxscaler scales all the data in the range of [0,1] or others in the range of [1,1] if the data set contains negative values the scaling comprises all the inliers in the range of [0,0.005].

StandardScale does not give any kind of assurance of balanced feature scales when an outlier is present. Deviation leads to compression in the range of upcoming values as an effect of the presence of outliers while figuring out the empirical mean and standard deviation.

On the other hand, RobustScaler() can be used to abolish the outliers and then use one of two StandardScaler or MinMaxscaler for preprocessing of the dataset.

The key features are very booming for the outliers. The data is in a range of 1st quartile and 3rd quartile that is specifically in 25th quantile and 75th quartile range, which is also known as interquartile range with the medium of this method.

Then the ranges are stored in a way it could be reopened and taken into use in the upcoming future as a transform method. If the outliers are present in the dataset then the ranges that provide better results outperform trial mean and variance.

**Parameters of RobustScaler:**

**With\_ceteringboolean:** The data is centered before scaling, It is true by default when getting used to sparse matrices, the change will raise the expectations because of the complex matrix, which is generally too large to get placed in the memory.

**with\_scaling:** It ranges the data in the form of interquartile range, quantile\_range: tuple(q\_min, q\_max), 0.0 <q\_min<q\_max< 100.0

Quantile range is used to calculate scale. By default, it is set as below.

Default: (25.0, 75.0) = (1st quantile, 3rd quantile) = IQR

**copy:** boolean It is an alternative parameter, by default it is true. If the input is already a NumPy array or a scipy. sparse CSC matrix and if axis = 1, avoid it by coping with setting its parameters to false and in place play in place row normalization(GeeksforGeeks, 2020b).

**Confusion Matrix**

In recent observations, it comes out that the transaction monitoring system can also give false-positive alerts. The requirement is to use machine learning to auto close these false alerts. The grounds for the machine learning trial was a negative predicted value that defines the total prediction of the model and how many cases it has recognized correctly.

NPV = True Negative / (True Negative + False Negative)

The price of false negatives is way too high due to the cases where the model is claiming that they are deceitful but in reality, they are deceitful. To get the procedure the quick display of the confusion\_matrix and below the output from the notebook, binary classification model is built with target = 1(Agrawal, 2015).

To check the achievement of the machine learning algorithms a confusion matrix is used, generally properly managed one. The row speaks for instances of the actual class and each column speaks for instances of a predicted class. This can be used alternatively. As take from the name the matrix shows the true thing that while using, the user gets confused at the time of classification of algorithms. When the confusion matrix is constructed the mislabeling of the character will be increased by one(Python-course, 2015).

**Naive Bayes**

It is a store of algorithms related to the Bayes Theorem. Every bunch of features being analyzed as separate from each other is the major principal of the naive Bayes. It is kind of a family of algorithms.

Let us assume a dataset. Assume a dataset that specifies the weather conditions playing a game of golf. Given the weather conditions, each tuple classifies the conditions as fit(“Yes”) or unfit(“No”) for playing golf.

Bayes Theorem finds the probability of the ongoing event based on past events.

P(A|B) = \frac{P(B|A) P(A)}{P(B)}

Bayes theorem in mathematical form, where A and B are events and P(B)? 0.

Generally, seeks to get the probability of event A. On the basis of B, Event B is also named as evidence, because it is true. P(A) is the proof of the attribute value of an unknown instance(GeeksforGeeks, 2020a).

## 3.7. What is a Support Vector machine?

The support vector machine deals with finding a hyperplane in an N-dimensional space that accurately classifies the data points. To split the class of data point numbers there are several possibilities of hyperplanes that could be picked. The main focus is to find a plan that has maximum margin and maximum distance from the data point in both classes. To find the future data points the maximization of the margin distance by providing a reinforcement.

In a classification of the data, point hyperplanes are used because of the decision boundaries. Data points on any side of the hyperplane can be considered as different classes. The dimensions of the hyperplane are done depending on the features. If the input is featured by 2 then the output is a single line. If the input feature is more than 2 then the out featured is a two or three-dimensional line accordingly. It becomes complicated when the number of features exceeds 3.

These are the data points that are closer and affect the position and orientation of the hyperlane. The manipulation of the margin can be done with the help of a classifier. After deleting the vector the positions of the hyperplane change. This point helps in building own SVM(Gandhi, 2018).

## 3.8. Random Forest Classifier

Random forest classifier is a learning algorithm that works on the principle of supervision. It can be used in both the manners for classification as well as the regression. It is the most convenient and user-friendly algorithm. A forest contains trees. Due to the number of trees, the forest is meant or seen as forest. In a very appropriate manner, Random forest makes the decisions based on randomly selected data trials. Using voting, random results are selected. It is also loaded with proper indicators for the feature importance.

It provides several numbers of different features of applications like recommendation engines, image viewers and feature selection. It is majorly used to identify the genuine loan applications. In a classification of any kind of scam-related activity and finding it before its get happens. It lies at the base of the Boruta algorithm, generally selecting features that are important(Navlani, 2018).

## 3.9. Decision Tree Algorithm

The decision tree is a flowsheet-like description of the structure where an internal node represents a feature, the branch represents a decision of the rule, and each leaf node represents the output. The most top node in the tree is called the root node. The partition can be done based on attributes value. The way of partition is recursive. It gets split in recursively. The flow sheet-like structure guides to take an accurate decision. It is designed in such a way that humans can easily grasp it. Just because of this quality, these are easy to understand and decode.

The tree is made on the white box type pattern of the algorithm. It shares logic that are crucial for decision making and are not available in the black box type of algorithms. It works on the white box type logic to share internal calls. Comparing with another algorithm like the neural network algorithms it is faster and more convenient while using. The time fixing of the decision trees is a function of a number of records and the attributes in the given data. The decision tree method is distribution-free as well as non-parametric method. It does not rely on the probability distribution assumptions. The decision tree is capable of handling high dimensional data with good accuracy.

# Chapter 4: Results

## 4.1. About Python

Python is an interpreted language. It is used for extensive purposes which allow using for different things. This language was firstly developed by Mr Guido van Rossum in 1991. Python plays a major role in various fields like Machine Learning, Operating systems, AI, application development and the gaming world. It is very simple and good structured fundamentally. This is the feature that makes it special and used for many development applications. The syntax of python programming is easy and understandable. It is open-source and it uses an interpreter for the compilation of a program (Van Rossum, 2007).

This is the language that is dynamic with high standards. It allows functional, procedural and object-oriented programming which makes it an all-rounder in programming languages. It has a lot of features that made it famous and used in almost every field of technology. The different features of this language are mentioned below (w3schools, 2015):

* Python can be utilized in web development on servers.
* It can be used to generate processes along with the software.
* This language can be attached to the databases which allow reading and writing the files.
* Python can manage complex problems of mathematics and big data.
* Python helps to create quick documentation regarding productive applications.
* Python language can work on any OS such as windows, Mac etc.
* This language syntax is very easy as it looks like a usual English language.
* Unlike other complex coding languages, Python allows the programmer to write a program with lesser lines.
* When the code is being written, it gets compiled automatically in the background as it uses an interpreter.
* It can process functional, object-oriented and procedural kinds of programming.
* Python version 3.7 is the latest in the market. Despite the quite performance of version 2, it is not updated with proper security.
* This language is very flexible so that it can be written in Pycharm, Thonny for managing large python collections.
* It is easy to read the code as it has a lot of similarities with the English language.
* Python uses empty spaces to indent instead of loops and scope of a variable. Flower braces are used in other complex programming languages.

Before entering the programming state for the present malevolent app identification study, the appropriate technology should give valid conclusions. Python is the only best solution for resisting reviews. Python has a lot of pre-built modules and libraries that can make the tasks easy for the user to implement (Harrington, 2012).

The difference between positive and negative views can be easily identified using an advanced python project. Researchers have chosen python to get the best outcomes when compared to other complex coding languages for execution. So, just by observing the program, the researchers could make out the differences between positive and negative views for the study. This leads to obtaining the desired outputs.

## 4.2. Results

In this world one of the most famous and reliable ecommerce company is Amazon, the company use lots of algorithms for product recommendation and other things, in this project is to analyze the sentiment of a review given by the user.

Lots of people buy and sell products on Amazon on a daily basis and many of them satisfy with the products and some of them don’t, so for the communication between the company and a user the company provides the user with a facility to write their comments on the product in the comment section of the specific product.

There will be lots of reviews in such a huge platform for each product, so to make the product more reliable, amazon needs to look into every comment and analyze them and use them to improve sales, but it is nearly impossible to look into such a huge set of comments or reviews, so there must be automatic analysis of the reviews and find the sentiment out of them.

This project tried to mimic the automatic sentiment analysis of user reviews of products with the help of Deep Learning with real-time reviews from the amazon platform.

Below mentioned are some basic steps that are used in the project:

* Setting up the system by changing the runtime as GPU in google colab and checking the memory status of the GPU.
* Downloading and unzipping the dataset from the Kaggle website by uploading the account authentication JSON file.
* Adding the labels for each review in the dataset.
* Preparing data for training by shuffling the data and removing stop words etc.
* Creating a CountVectorizer for our dataset for building a bag of words model to prepare data for training by Vectorizing the words into a vector.
* Splitting the dataset into training and testing set by a ratio of validation\_split = 0.2 with the help of train\_test\_split from a famous library called scikit to learn and particularly from the model\_selection module

**Dataset**

The dataset used in this project is called the Amazon reviews dataset from the Kaggle website.

The dataset contains 3600000 reviews in the training set and 400000 reviews of different products and for each review, there consists of a caption which is the sentiment of the particular review which means it is a ‘positive’ review or a ‘negative’ review.

The dataset is very huge, which is very computational so here used only some of the reviews in the dataset in this project.

Although the dataset is already split into training and test set, divide the training set into a validation set with 0.2 as validation split ratio.

* Building 5 different machine learning models and train them with the dataset
* Validating each model by some metrics like confusion matrix, accuracy score, f1 score and precision score
* Comparing all the machine learning algorithms built with the help of all the metrics, and by plotting some bar graphs between them.

**Step 1 – Setting the GPU in Google Colab:**

A famous machine learning site is used called google colab for running our project. First configured the google colab kernel into a GPU which is provided by google free, the GPU provided has 12GB RAM which is good enough for a real-time Sentiment analysis of amazon reviews.

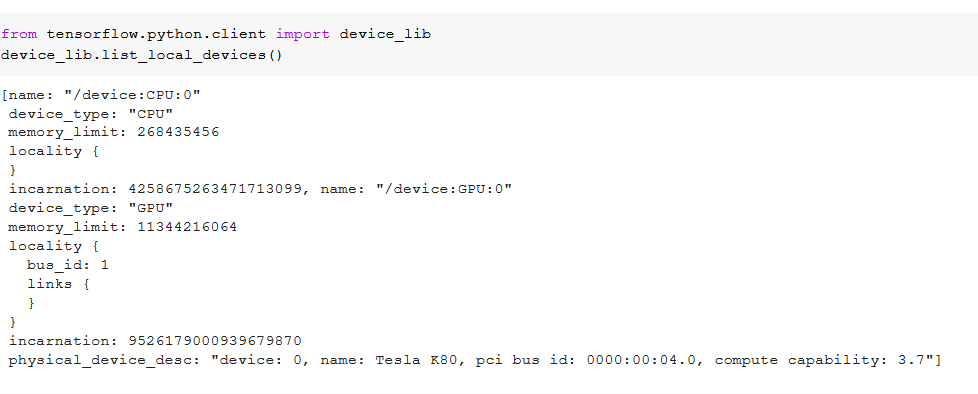


Figure 1: Memory Status of the GPU used in the project

**Step 2 – Downloading the Dataset from Kaggle:**

In this step initially downloaded the dataset from a famous website called Kaggle. It contains many datasets for performing machine learning and deep learning. The dataset used in the project is Amazon Reviews.

The dataset used was 3600000 reviews in the training set and 400000 reviews in the test set, which is huge and highly computational to work on, so used a part of the dataset in this project.

**Step 3 – Importing Required Packages**

This step includes importing all the packages required for building the model and for preprocessing the images and even plotting images and graphs in project.

**Step 4 – Preparing dataset for training.**

The dataset is raw so need to remove some of the unwanted data and process some of the data, such that can use the data well to predict real-time sentiment analysis.

There are a lot of steps in preparing the dataset for training, all the steps are mentioned and explained below:

* **Text Labelling**

Every review in the dataset consists of a caption that explains the sentiment of the review for the product, the caption says the sentiment for the given review by classifying the reviews as positive or negative, but all the captions are attached to the reviews, in this step attached all the relevant captions for all the reviews, so here the captions are the independent variables.

* **Shuffling Dataset**

The dataset contains the reviews in a random way, but for any other future problems like overfitting for a sequence, need to shuffle the dataset in random again.

The shuffling is done with the help of the shuffle function in the utils module from the scikit lean library

* **Processing the Sentences and words**

In this step break all the reviews into words and check for some outliers from the sentence, there are many processes need to perform to process the sentences they are mentioned below

1. Removing the quotation marks which do not make any sense in finding the sentiment of the complete sentence.
2. Keeping only letters in words and removing any other characters like numbers and others.
3. Converting all the sentences into lowercase
4. Split the sentence into words
5. Stemming the words
6. Removing all the stop words like in, the etc which does not help in finding the sentiment of the sentence.
7. Finally, combine the words into a sentence

**Step 5 – Using Bag of Word model for vectorizing the review**

For fitting the dataset into the machine learning model, need to vectorize the dataset with the help of a bag of word models.

A bag-of-words(BoW) model, or BoW for short, is a way of extracting features from the text for use in modeling, such as with machine learning algorithms.

A bag-of-words is a representation of text that describes the occurrence of words within a document. It involves two things:

1. A vocabulary of known words.
2. A measure of the presence of known words.

**Step 6 – Splitting the dataset into training and testing dataset**

The training and test set are already split in the dataset, but in this kernel, ignore this info and re-divide the dataset into a training set and test set with the help of a function called train\_test\_split from scikit learn module with a validation split of 0.2.

**Step 7 – Building different Machine learning models**

Finally, after preprocessing all the data, built 5 strong models for and trained them with the data and checked the performance of each model.

Below are the models that built and their summary.

* **Naive Bayes**

Naive Bayes is one of the best algorithms in classification, which works with Bayes theorem, used it from scikit learn library and navie\_bayes module.

Below mentioned are some of the metrics go with Naive Bayes algorithm

Accuracy: 72.74%

F1 Score: 0.7342

Precision Score: 0.698

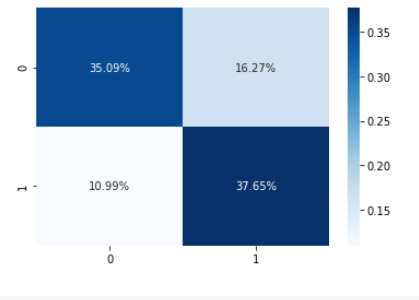


Figure 2: Confusion matrix for Naive Bayes algorithm

**Support Vector Classifier**

Support vector shortly known as SVC classifier is a different classifier from another common classifier and it is also one of the strongest classifiers in the machine learning model, it is imported from the SVM module in scikit learn library.

Below mentioned are some of the metrics go with Support vector classifier algorithm

Accuracy: 82.74%

F1 Score: 0.8120

Precision Score: 0.8287

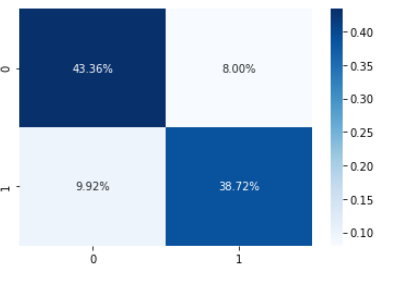


Figure 3: Confusion matrix for Support Vector Classifier.

* **Decision Tree Classifier**

Decision tree classifier user a tree approach to train the given data, used it from the tree module in sklearn module

Below mentioned are some of the metrics we go with Decision tree classifier algorithm

Accuracy: 69.76%

F1 Score: 0.6969

Precision Score: 0.6798

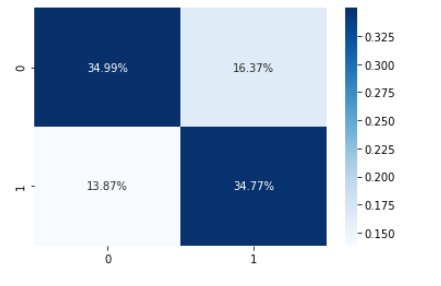


Figure 4: Confusion matrix for Decision Tree algorithm.

* **Random Forest Classifier**

Random Forest Classifier uses a lot of decision trees to train the dataset which gives an optimal result, it is imported from ensemble module from scikit learn library

Below mentioned are some of the metrics go with Random Forest classifier algorithm

Accuracy: 82.66%

F1 Score: 0.8223

Precision Score: 0.8200

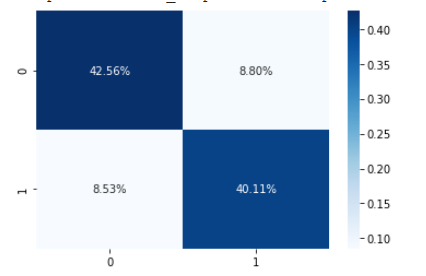


Figure 5: Confusion matrix for Random Forest algorithm.

* **Catboost**

Catboost is one of the famous and trending models for both classification and regression, it is a third party model so first downloaded the required packages and imported them to perform the model.

Below mentioned are some of the metrics go with Catboost algorithm

Accuracy: 81.22%

F1 Score: 0.8093

Precision Score: 0.7997

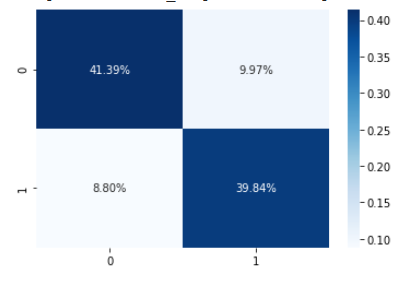


Figure 6: Confusion matrix for Cat Boost algorithm.

**Step 8 – Evaluating the model performance with some metrics and comparing models**

After training and testing all the models it’s time to compare them with some metrics, and through some visualization like bar graphs, the metrics used to compare the models are mentioned and explained below.

* **Accuracy score**

Accuracy score is one of the most intuitive performance measures and it is simply a ratio of correctly predicted values to the total observations.

Accuracy = True Positive + True Negative/ True Positive + False Positive + False Negative + True Negative

Or

Accuracy = TP+TN/TP+FP+FN+TN

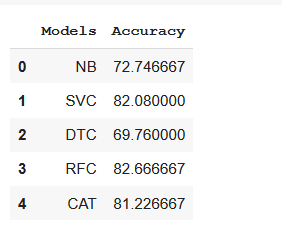


Figure 7: Data frame for all the model's Accuracy Score.

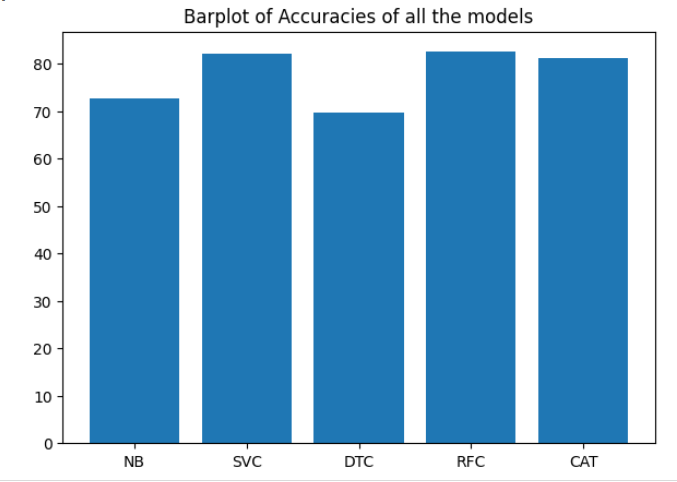


Figure 8:Bar graph which compares Accuracies of models.

* **Precision Score**

Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.

Precision Score = True Positive / True Positive + False Positive

Or

Precision = TP/TP+FP

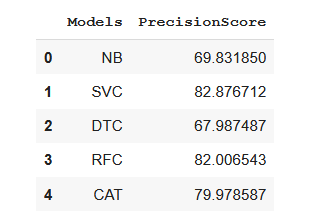


Figure 9: Data frame for all the models Precision Score.

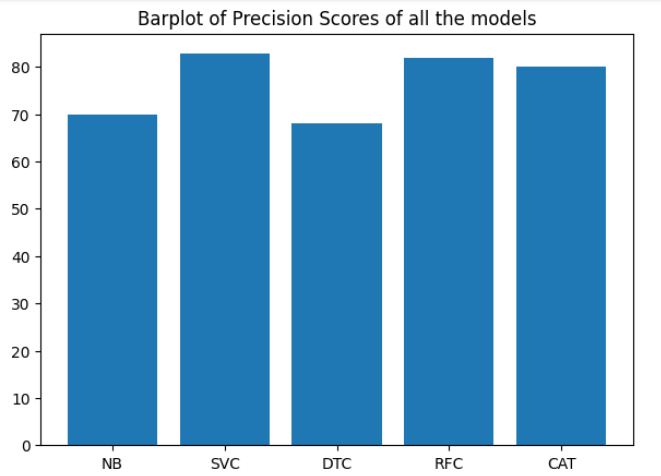


Figure 10: Bar Graph to compare F1 Score.

* **F1 Score**

F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account, F1 is usually more useful than accuracy.

F1 Score = 2\*(Recall \* Precision)/ (Recall + Precision)

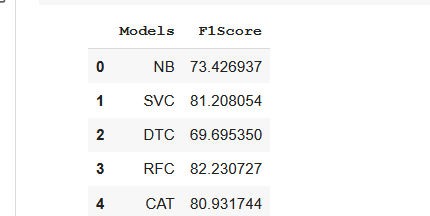


Figure 11: Data frame for all the models F1 Score.

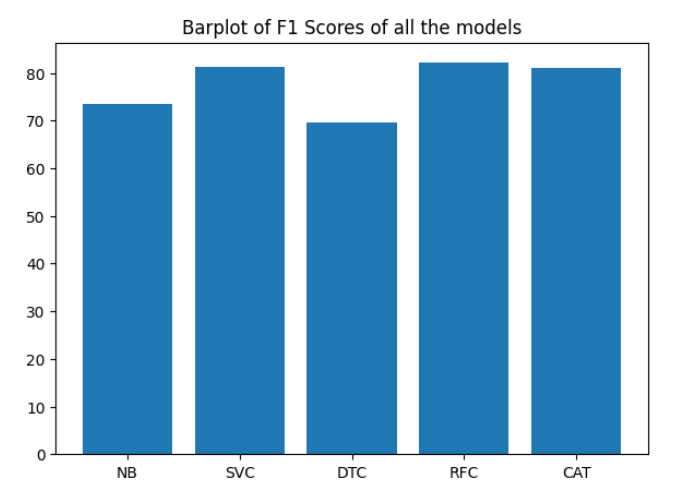


Figure 12:Bar Graph to compare F1 Score.

**Crucial Conclusions:**

* The google colab GPU is used in the project which provides 12GB Ram
* Dataset is downloaded from a famous website named Kaggle
* Important packages are imported which are useful for this project model and other sections.
* Amazon reviews dataset is used in this project which contains 3600000 in the training set and 400000 in the test set.
* Preparing the reviews with the help of some models for training
* Bag of words model is used to vectorize the whole dataset with the help of Count Vectorizer from scikit learn
* 5 different Machine Learning models are built and trained with the dataset have and tested with the validation set
* Compared all models with the help of some metrics namely accuracy, precision and F1 score

# Chapter 5: Discussion

There are many methods to implement sentiment analysis as mentioned earlier. It is learned in this research that the traditional methods not using machine learning like dictionary methods or rule-based analysis methods require more manual involvement, cannot perform analysis on huge sets of data (bigdata) and achieving automation of analysis is quite not possible with them. So, research is carried out on different methods based on machine learning. Out of different machine learning methods, accurate methods like SVC (Support Vector Classifier), NB (Naïve Bayes), Random Forest, Decision Tree, are filtered out. This filtering is done based on the Size of the training data, Accuracy and/or Interpretability of the output, Speed or Training time, Linearity and Number of features.

To implement these methods, the dataset is taken from Kaggle. The reason to choose Kaggle is that it has good integration with Google Colab, a web Python IDE best suitable for Machine learning and data analysis. As this project is built using Google Colab, importing data set directly from the Kaggle link saves time and storage. Once the dataset is imported, it is prepared for use after preprocessing it.

Data in the datasets are shuffled to get better results. The data set consists of # (total) of reviews and it is divided into training, validation, and testing sets. An 80-10-10 or 90-5-5 ratio would be great for large data set and a 60-20-20 ratio split would be ideal for a small dataset. As the data set is medium-large, this Amazon reviews data set is divided in the ratio of 70% -20%-10% respectively. Also, this ratio split mostly gave the best performance. Each part of the data set division should not be skewed. To remove the skewness, it is made sure that an equal ratio of negative and positive reviews is present in all the divided datasets.

In feature extraction, Bag of Words, a simple yet elegant model is used to represent the review text as the bag of its words not taking grammar and word order into account. Word multiplicity is kept and each review is vectorized using that model. For each method, the machine learning model is built, trained, and tested. Scikit-learn or SK learn, an ML library for the Python language, has the most efficient tools for modelling and machine learning. It has a consistent and unified interface to implement these models. The main objective is to build the model and evaluate them.

For evaluating the different models mentioned earlier, accuracy score, precision, recall and f1 score are calculated for each model and compared.

Initially, for the 200 manually labelled feature vectors of sentences, all the models showed similar performance. All the models did quite well in the early testing. This is done to make sure all the models are working properly.

From the 2-million machine-labelled sentences, 2-million feature vectors are generated which have the identical amounts of both positive and negative vectors. This whole set of vectors can be given for a model to train. For viewing the performance of a model and how it trains, the model can be trained with different sized vector sets increasing in size. So, the models are fed with four subsets obtained from the 2 million vector-set along with the total vector itself. The four subsets are designated A, B, C and D and each subset contains 200 vectors, 2000 vectors, 20,000 vectors, and 200,000 vectors respectively. Each subset also contains an equal ratio of positive labels and negative labels. With these 5 sets of feature vectors, models are trained and their performance is then evaluated. For the subsets with smaller sizes, all models took a similar time. The naive Bayes model did exceptionally well for the smaller sets averaging a 0.73 F1 scores while taking less time among all. But it could not keep up the accuracy as the feature vector size increased. With the increase in training data, F1 scores gradually increased. Linear SVC made a significant jump from the initial 0.61 to an overall 0.81 as the feature vector size increased. The Random Forest model followed Linear SVC in terms of accuracy and other scores but could not beat it. But it executed slightly faster than Linear SVC. In the middle phase, the Random Forest model outperformed Linear SVC with faster execution and better overall scores. Decision Tree mode performed decently well with smaller and moderate sets. It was faster than the Random Forest model but at the expense of lesser overall scores. As and when the size of the feature vector increased, the Decision Tree overfit the data.

# Chapter 6: Conclusion and Recommendations

## 6.1. Conclusion

The process of understanding and identifying people opinion or emotions from text data is called sentiment analysis. It can also be called opinion mining. It is used by companies to improve their product or service quality and make what people want. In this project, the sentiment analysis (SA) is implemented on Amazon’s Reviews dataset. Different demonstrations were accomplished for mining the sentiments having large datasets. Several classification techniques like NB, Linear SVC, Random Forest and Decision Tree are implemented using sklearn or scikit learn MLlib. For the project 400000reviews are used in model training, # reviews are used for model validation and divide the training set into a validation set with 0.2 as validation split ratio for testing the models. Few steps were applied for exploring and analyzing the data.

The comparison and evaluation of Naïve Bayes, Random Forest, Decision Tree, and Support Vector Machines are presented in this project. The models are evaluated with the measures like accuracy, precision, recall and F1 score. The scores and performance of each model are presented for concluding them. By performing the analysis, it is clear that the linear support vector classifier works well than other classifiers. Support Vector Classifier dominates the proceedings with above 85% in all the evaluation measures. The only thing that shadows its supremacy is its execution time. If the application of opinion mining requires high accuracy and the dataset is complex, Linear SVC suits the demand. It can also be used when speed is not the factor in developing a sentiment classifier. The results also indicate that Random Forest can also achieve good evaluation values by taking less time than SVC. Random Forest is the overall second-best method and it can be considered to build a sentiment classifier with it if the dataset is moderately large and good accuracy with decent speed is the need. But with the large datasets, Linear SVC outshines the Random Forest. Coming to the decision tree, it is faster than Random Forest and has fewer average scores than its predecessor i.e., Random Forest. Also, it becomes heavy for the Decision tree for large datasets. Lastly, Naïve Bayes is the fastest in terms of execution and provides fairly decent accuracy and overall decent scores. Thus, unless the dataset is complex and the application requires high accuracy, simpler models like Naïve Bayes and Decision Tree can be used as they are faster to train, require fewer computations and give decent results. It is observed thata number of reviews or the size of the dataset became less, naïve Bayes beat linear svc in terms of both accuracy and speed. So, it can be established that a classifier which operates on small datasets can be built using this model as it is the best in this area. But it cannot be scalable with the same accuracy. So as is the case with the Decision tree.

In conclusion, with this project, tried to show how different sentiment analysis techniques in machine learning can be implemented and evaluate their performances by applying them on Amazon review data. It can be concluded that random forest classifier is the most accurate and most complete classifier with high values in all the measures like accuracy, precision, recall and F1 score.

## 6.2. Recommendations

The models did a good job of analyzing the sentiment. So, the models implemented in this project can be used to develop a sentiment classifier depending on the constraints. Further, to improve the performance of the different classifier models, the features set can be modified into different types like bi-gram, tri-gram, and four-gram.

The important thing to note is that these models work well in deriving open sentiments like ratings or scores. Extracting sentiment of texts with sarcasm, negation, word ambiguity and multipolarity is still a challenging task. Even humans sometimes find it hard to understand sarcasm. So, more features need to be implemented to extract implicit sentiment from the text.

In the case of vectorizing the dataset, the Bag of Words (BoW) model is easy to implement and interpret. But another popular model, Term Frequency-Inverse Document Frequency (TF-IDF) performs better and even contains information about the importance of the word.

A hybrid approach of using both traditional methods like rule-based approaches, lexicon methods or dictionary methods, and ml methods can further improve the performance and accuracy. The hybrid approaches also solve the problem of extracting implicit sentiment.

Lastly, the implemented models cannot be extended to other languages as sentiment analysis is still in the infant stage for non-English texts. So, methods to tokenize and vectorize the non-English datasets should be developed. Then these Machine learning models can be extended to non-English languages also.

# Reference:

Abbas, M., Ali Memon, K. and Aleem Jamali, A. (2019) ‘Multinomial Naive Bayes Classification Model for Sentiment Analysis’, *IJCSNS International Journal of Computer Science and Network Security*, 19(3), p. 62.

Agrawal, S. (2015) *Understanding Confusion Matrix sklearn (scikit learn), Machine Learning | Clear explanation | Towards Data Science*. Available at: https://towardsdatascience.com/understanding-the-confusion-matrix-from-scikit-learn-c51d88929c79.

Al-Saqqa, S., Al-Naymat, G. and Awajan, A. (2018) ‘A Large-Scale Sentiment Data Classification for Online Reviews Under Apache Spark’, *Procedia Computer Science*. Elsevier, 141, pp. 183–189. doi: 10.1016/J.PROCS.2018.10.166.

Ali, Y. *et al.* (2019) ‘Detection of schistosomiasis factors using association rule mining’, *IEEE Access*. Institute of Electrical and Electronics Engineers Inc., 7, pp. 186108–186114. doi: 10.1109/ACCESS.2019.2956020.

Awan, M. J., Rahim, M. S. M., Salim, N., *et al.* (2021) ‘Efficient Detection of Knee Anterior Cruciate Ligament from Magnetic Resonance Imaging Using Deep Learning Approach’, *Diagnostics*. Multidisciplinary Digital Publishing Institute (MDPI), 11(1). doi: 10.3390/DIAGNOSTICS11010105.

Awan, M. J., Rahim, M. S. M., Nobanee, H., *et al.* (2021) ‘Social Media and Stock Market Prediction: A Big Data Approach’, *Computers, Materials and Continua*. Tech Science Press, 67(2), pp. 2569–2583. doi: 10.32604/CMC.2021.014253.

Bhatt, A. and Patel Gawande, K. (2015) ‘Amazon Review Classification and Sentiment Analysis’.

Bitdegree (2020) *A Guide on Splitting Datasets With Train\_test\_split Function*. Available at: https://www.bitdegree.org/learn/train-test-split.

Chen, X. *et al.* (2018) ‘A novel feature extraction methodology for sentiment analysis of product reviews’, *Neural Computing and Applications 2018 31:10*. Springer, 31(10), pp. 6625–6642. doi: 10.1007/S00521-018-3477-2.

Chung, J. *et al.* (2015) ‘Gated Feedback Recurrent Neural Networks’, *32nd International Conference on Machine Learning, ICML 2015*. International Machine Learning Society (IMLS), 3, pp. 2067–2075.

Crawford, M. *et al.* (2015) ‘Survey of review spam detection using machine learning techniques’, *Journal of Big Data 2015 2:1*. SpringerOpen, 2(1), pp. 1–24. doi: 10.1186/S40537-015-0029-9.

Dave, K., Lawrence, S. and Pennock, D. M. (2003) ‘Mining the peanut gallery: Opinion extraction and semantic classification of product reviews’, *Proceedings of the 12th International Conference on World Wide Web, WWW 2003*, pp. 519–528. doi: 10.1145/775152.775226.

Dumbleton, R. (2015) *Sentiment Analysis: Definition, Uses, Examples + Pros /Cons*. Available at: https://getthematic.com/insights/sentiment-analysis/.

Edpresso Team (2015) *CountVectorizer in Python*. Available at: https://www.educative.io/edpresso/countvectorizer-in-python.

Fang, X. and Zhan, J. (2015) ‘Sentiment analysis using product review data’, *Journal of Big Data 2015 2:1*. SpringerOpen, 2(1), pp. 1–14. doi: 10.1186/S40537-015-0015-2.

Gandhi, R. (2018) *Support Vector Machine — Introduction to Machine Learning Algorithms | by Rohith Gandhi | Towards Data Science*. Available at: https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47.

geeksforgeeks (2021) *Removing stop words with NLTK in Python - GeeksforGeeks*. Available at: https://www.geeksforgeeks.org/removing-stop-words-nltk-python/.

GeeksforGeeks (2020a) *Naive Bayes Classifiers*. Available at: https://www.geeksforgeeks.org/naive-bayes-classifiers/.

GeeksforGeeks (2020b) *StandardScaler, MinMaxScaler and RobustScaler techniques - ML -*. Available at: https://www.geeksforgeeks.org/standardscaler-minmaxscaler-and-robustscaler-techniques-ml/.

GeeksforGeeks (2020c) *Using CountVectorizer to Extracting Features from Text - GeeksforGeeks*. Available at: https://www.geeksforgeeks.org/using-countvectorizer-to-extracting-features-from-text/.

GeeksforGeeks (2021) *Python | Stemming words with NLTK - GeeksforGeeks*. Available at: https://www.geeksforgeeks.org/python-stemming-words-with-nltk/.

Ghose, A. and Ipeirotis, P. G. (2007) ‘Designing novel review ranking systems: Predicting the usefulness and impact of reviews’, *ACM International Conference Proceeding Series*, 258, pp. 303–310. doi: 10.1145/1282100.1282158.

Gokce, E. (2020) *Sentiment Analysis on Amazon Reviews*. Available at: https://towardsdatascience.com/sentiment-analysis-on-amazon-reviews-45cd169447ac.

Great Learning Team (2020) *Natural Language Toolkit (NLTK) Tutorial with Python*. Available at: https://www.mygreatlearning.com/blog/nltk-tutorial-with-python/.

Harrington, P. (2012) ‘Machine Learning in Action’, *Machine Learning*, 37(3), pp. 1–20.

Indra, S. T., Wikarsa, L. and Turang, R. (2017) ‘Using logistic regression method to classify tweets into the selected topics’, *2016 International Conference on Advanced Computer Science and Information Systems, ICACSIS 2016*. Institute of Electrical and Electronics Engineers Inc., pp. 385–390. doi: 10.1109/ICACSIS.2016.7872727.

Jabeen, H. (2018) *Stemming and Lemmatization in Python - DataCamp*. Available at: https://www.datacamp.com/community/tutorials/stemming-lemmatization-python.

Joachims, T. (1998) ‘Text categorization with Support Vector Machines: Learning with many relevant features’. Springer, Berlin, Heidelberg, pp. 137–142. doi: 10.1007/BFB0026683.

Kamal, A. (2013) ‘Subjectivity Classification using Machine Learning Techniques for Mining Feature-Opinion Pairs from Web Opinion Sources’.

Keras API (2015) *Dataset preprocessing*. Available at: https://keras.io/api/preprocessing/.

Khullar, V. and Prabhat, A. (2017) ‘Sentiment classification on big data using Naïve bayes and logistic regression Computer based Interventions for Individuals with Autism Spectrum Disorders View project Sentiment classification on Big Data using Naïve Bayes and Logistic Regression’. doi: 10.1109/ICCCI.2017.8117734.

Laksono, R. A. *et al.* (2019) ‘Sentiment analysis of restaurant customer reviews on tripadvisor using naïve bayes’, *Proceedings of 2019 International Conference on Information and Communication Technology and Systems, ICTS 2019*. Institute of Electrical and Electronics Engineers Inc., pp. 49–54. doi: 10.1109/ICTS.2019.8850982.

Lilleberg, J., Zhu, Y. and Zhang, Y. (2015) ‘Support vector machines and Word2vec for text classification with semantic features’, *Proceedings of 2015 IEEE 14th International Conference on Cognitive Informatics and Cognitive Computing, ICCI\*CC 2015*. Institute of Electrical and Electronics Engineers Inc., pp. 136–140. doi: 10.1109/ICCI-CC.2015.7259377.

Liu, B. *et al.* (2013) ‘Scalable sentiment classification for Big Data analysis using Naïve Bayes Classifier’, *Proceedings - 2013 IEEE International Conference on Big Data, Big Data 2013*, pp. 99–104. doi: 10.1109/BIGDATA.2013.6691740.

Malik, M. S. I. and Hussain, A. (2018) ‘Exploring the influential reviewer, review and product determinants for review helpfulness’, *Artificial Intelligence Review 2018 53:1*. Springer, 53(1), pp. 407–427. doi: 10.1007/S10462-018-9662-Y.

Mazhar Javed, A. *et al.* (2019) *Acceleration of Knee MRI Cancellous bone Classification on Google Colaboratory using Convolutional Neural Network*. Available at: http://www.warse.org/IJATCSE/static/pdf/file/ijatcse13816sl2019.pdf.

Mcauley, J. and Leskovec, J. (2013) ‘From Amateurs to Connoisseurs: Modeling the Evolution of User Expertise through Online Reviews’.

McAuley, J. and Leskovec, J. (2013) ‘From Amateurs to Connoisseurs: Modeling the Evolution of User Expertise through Online Reviews’, *WWW 2013 - Proceedings of the 22nd International Conference on World Wide Web*, pp. 897–907.

Mikolov, T. *et al.* (2010) ‘Recurrent neural network based language model’.

Navlani, A. (2018) *Random Forests Classifiers in Python - DataCamp*. Available at: https://www.datacamp.com/community/tutorials/random-forests-classifier-python.

Neethu, M. S. and Rajasree, R. (2013) ‘Sentiment analysis in twitter using machine learning techniques’, *2013 4th International Conference on Computing, Communications and Networking Technologies, ICCCNT 2013*. doi: 10.1109/ICCCNT.2013.6726818.

Pang, B. and Lee, L. (2008) ‘Opinion Mining and Sentiment Analysis’, *Foundations and Trends in Information Retrieval*. Now Publishers Inc. PUB4850 Hanover, MA, USA, 2(1–2), pp. 1–135. doi: 10.1561/1500000011.

Prabhat, A. and Khullar, V. (2017) ‘Sentiment classification on big data using Naïve bayes and logistic regression’, in *2017 International Conference on Computer Communication and Informatics, ICCCI 2017*. Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/ICCCI.2017.8117734.

Python-course (2015) *Machine Learning with Python: Confusion Matrix in Machine Learning with Python*. Available at: https://www.python-course.eu/confusion\_matrix.php.

Quyyam, T. and Ghous, H. (2021) ‘Sentiment Analysis of Amazon Customer Product Reviews: A Review’, *International Journal of Scientific Research and Engineering Development*, 4.

Rafea, A. and Morsy, S. A. (2015) ‘A Hybrid Approach for Automated Document-level Sentiment Classification (Proposal)’.

Reddy, V. (2018) *Sentiment Analysis using SVM. Sentiment Analysis is the NLP technique… | by Vasista Reddy | Medium*. Available at: https://medium.com/@vasista/sentiment-analysis-using-svm-338d418e3ff1.

Roldos, I. (2020) *5 Sentiment Anlysis Examples in Business*. Available at: https://monkeylearn.com/blog/sentiment-analysis-examples/.

Van Rossum, G. (2007) *Python Programming Language | USENIX*. Available at: https://www.usenix.org/conference/2007-usenix-annual-technical-conference/presentation/python-programming-language.

Sagepub (2015) ‘The Selection of a Research Approach’.

Sharedalal, R. (2019) ‘AMAZON FINE FOOD REVIEWS-DESIGN AND IMPLEMENTATION OF AN AUTOMATED CLASSIFICATION SYSTEM’.

Shojaee, S. *et al.* (2014) ‘Detecting deceptive reviews using lexical and syntactic features’, *International Conference on Intelligent Systems Design and Applications, ISDA*. IEEE Computer Society, pp. 53–58. doi: 10.1109/ISDA.2013.6920707.

Socher, R. *et al.* (2013) ‘Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank’.

Sultana, N. *et al.* (2019) ‘SENTIMENT ANALYSIS FOR PRODUCT REVIEW’, *ICTACT JOURNAL ON SOFT COMPUTING*, p. 3. doi: 10.21917/ijsc.2019.0266.

Tai, K. S., Socher, R. and Manning, C. D. (2015) ‘Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks’, *ACL-IJCNLP 2015 - 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, Proceedings of the Conference*. Association for Computational Linguistics (ACL), 1, pp. 1556–1566.

TangHuifeng, TanSongbo and ChengXueqi (2009) ‘A survey on sentiment detection of reviews’, *Expert Systems with Applications: An International Journal*. Pergamon Press, Inc. PUB1185 Elmsford, NY, USA, 36(7), pp. 10760–10773. doi: 10.1016/J.ESWA.2009.02.063.

Vinodhini, G. (2012) ‘Sentiment Analysis and Opinion Mining : A Survey’.

w3schools (2015) *Introduction to Python*. Available at: https://www.w3schools.com/python/python\_intro.asp.

Wawre, S. V and Deshmukh, S. N. (2016) ‘Volume 5 Issue 4, April 2016 www.ijsr.net Licensed Under Creative Commons Attribution CC BY Sentiment Classification using Machine Learning Techniques’, *International Journal of Science and Research (IJSR) ISSN*.

Yuan, Y. and Zhou, Y. (2015) ‘Twitter Sentiment Analysis with Recursive Neural Networks’.

z\_ai (2019) *Targeted Sentiment analysis vs Traditional Sentiment analysis*. Available at: https://towardsdatascience.com/targeted-sentiment-analysis-vs-traditional-sentiment-analysis-4d9f2c12a476.

# Appendix (Code)

# Access to the code and files are uploaded to the below google drive link.

# GOOGLE DRIVE :

# <https://drive.google.com/drive/folders/1MoNIi6_QEBQUCD1P2NnsZqf2sazeXLop?usp=sharing>