



Sentiment analysis on Customer review and News headlines

Presented by

Ashwini Sawarkar

20211120

Supervisor

Dr. Neil Buckley

January 18, 2022

## Table of Contents

Abstract.....	3
1 Introduction.....	4
1.1 Motivation .....	4
1.2 Aims and objectives .....	4
1.3 Layout of the dissertation .....	5
2 Background.....	5
2.1 Technical Details.....	5
2.1.1 Lexicon Approach or ruled based approach .....	5
2.1.2 Machine learning approach.....	6
2.1.2.1 Binary Classification .....	6
2.1.2.2 Multiclass Classification .....	6
2.1.2.3 Multi-label classification.....	6
2.1.2.4 Imbalanced Classification .....	6
2.1.3 Deep Learning .....	6
2.2 Feature Extraction .....	7
2.3 Evaluation Measures .....	7
2.3.1 Accuracy.....	7
2.3.2 Recall .....	8
2.3.3 Precision .....	8
2.3.4 F1 Score.....	8
2.3.5 Specificity .....	8
2.3.6 ROC (Receiver Operator Characteristic) .....	8
2.3.7 AUC (Area under ROC curve) .....	9
2.3.8 confusion matrix.....	10
2.4 Models Used in the Study.....	10
2.4.1 Naive bayes.....	10
2.4.2 Logistic regression.....	11
2.4.3 SVM .....	12
2.4.4 Decision Tree.....	12
2.4.4.1 Entropy:.....	13

2.4.4.2	Gini Impurity.....	14
2.4.5	KNN .....	14
2.4.6	Random Forest Classifier .....	15
3	Literature Review.....	17
3.1	Related Works .....	17
3.2	Identified Challenges and Gaps.....	20
3.3	Recommendations and Future Research Directions .....	20
3.4	Conclusion.....	22
4	Methodology .....	23
4.1	Architecture of the study .....	23
4.2	Data Preparation.....	24
4.2.1	Data Extraction.....	24
4.2.2	Pre-processing Data.....	25
4.2.3	Feature Extraction.....	27
4.3	Data Visualization .....	36
4.3.1	Bar Diagram and sentiment table for Customer review and News Headlines .....	36
4.3.2	Box plot Length of the Customer review and News Headlines .....	37
4.3.3	Word Cloud .....	38
4.4	Model Development .....	39
4.4.1	Naïve Bayes for Customer review .....	40
4.4.1.1	Naïve Bayes when stopwords present with BagOfwords.....	40
4.4.1.2	Naïve Bayes when stopwords present with TFIDF .....	40
4.4.1.3	Naïve Bayes when stopwords not present with BagOfwords.....	41
4.4.1.4	Naïve Bayes when stopwords not present with TFIDF .....	42
4.4.2	Logistic regression for Customer review .....	44
4.4.3	SVM-Support vector Machine for Customer Reviews: .....	45
4.4.4	Decision Tree for Customer Reviews .....	46
4.4.5	KNN for Customer review.....	47
4.4.6	Random Forest Classifier for Customer review.....	48
4.4.7	RandomForestClassifier for News Headlines .....	50
4.4.8	Naïve Bayes for News Headlines .....	51
4.4.9	Decision Tree for News Headlines .....	52
4.4.10	Logistic regression for News Headlines .....	53
5.	Result and Discussion .....	55
5.1	Result Table.....	59

5.4.1	Naïve Bayes.....	59
5.4.2	Logistic regression.....	59
5.4.3	Decision Tree.....	60
5.4.4	Random forest.....	60
5.4.5	SVM .....	60
5.4.6	KNN .....	60
5.2	Accuracy and AUC Result Table .....	61
5.3	Customer Review ROC curve .....	61
5.4	Summary .....	63
6.	Conclusion .....	64
6.1	FurtherWorks.....	65
7.	References.....	65

## Abstract

Over the last decade, sentiment analysis has become widespread in many areas, including business, social networking, and education. The use of sentiment analysis is increasing, but challenges remain. Despite the identified challenges, this area is growing rapidly, especially in relation to the latest trend, the application of DL. Under these aspects, he emphasized the need for a growing focus on structured datasets, standardized solutions, and emotional expression and cognition.

Sentiment Analysis is one of the subsets of Natural Language Processing (NLP). With the appearance use of the internet, it's far very not unusual place for customers to publish opinions and feedbacks approximately acquired carrier and product. Being capable of decide and extract beneficial statistics from those opinions is largely what sentiment evaluation entails. While E-trade has grown in large part in making use of sentiment evaluation, the schooling zone also can use this evaluation to benefit useful perception into scholar engagements and educational achievements, even as potential college students may benefit insights from the enjoy of different college students. The goal of this paper is to explain the utility of sentiment evaluation on college students' opinions the usage of numerous system studying and deep studying techniques.

Keywords: Natural Language Processing; Sentiment Analysis; Deep Neural Network; Word2Vec; Recurrent Neural Network; Convolutional Neural Network; Long-Short Term Memory; Bi-Directional Long-Short Term Memory

# 1 Introduction

Sentiment analysis is a way for detecting the underlying feelings of a textual content. This is the technique of classifying textual content as both positive, negative, or neutral. Machine mastering strategies are used to assess textual content and decide the temper in the back of it. Lonely people just ask, but organizations do surveys, conduct polls, or hire dedicated consultants. The terms opinion analysis and sentiment analysis are used synonymously. The first is overused in the sense of industrial sector. The essence of analysis is to determine a person's attitude, point of view, emotion, or judgment. The polarity of a specific topic or the general context of a document.

## 1.1 Motivation

Sentiment analysis is a research area that has received a great deal of attention over the last decade. This area includes various applications that have been featured in various research studies. With great effort over the last few years, it was created to study the impact of various media on the financial world. In the financial world, different types of information are important. For example, investors are worried about financial news related to their investment. Companies are interested in news about competitors, suppliers, resources, and customer feedback. In return, customers are interested in other customer ratings for the products they want to buy. To address this, different sentiment analysis applications are emerging in different areas such as financial news sentiment analysis, product reviews and political elections.

As the Internet grows in popularity, sentiment analysis research is becoming increasingly important as it provides a simple and effective way to exchange ideas. With the development of the internet, it is now possible to post opinions and thoughts on different topics on different websites. Maximum no of people can share their point of view in or the thoughts on the web and recognize and get the opinions or the point of view of others. Therefore, the quantity of the data is in a large amount which is containing opinions generated from various sources such as customer reviews, discussions in the forum, online blogs or microblogs, and the social media posts<sup>3</sup> (George, 2015).

And this data is scattered all over the internet so the motivation behind this research is to analysis the sentiment and get the information at the central place so that customer or the particular organization before taking the decision check the details at central place rather that checking each website for the reviews or the news headlines.

With this study the number of organizations or the no of client's requirements can be achieved for customer review sentiment as well as news headlines data.

## 1.2 Aims and objectives

The main aim of this study is to explore or describe an best or the effective way to conduct sentiment analysis by filtering out the positive and negative reviews from the online website sources and working on improving the performance of sentiment classification and extracting aspects related with the sentiments. To cater for this aim, there are three objectives that this research has tried to achieve.

1. Objective to handle the text that having the positive and negative point of views or the opinions, because most of the data in the real word data describe or mentioned the positive and negative sentiments in the same document. Most of the time the same documents contain the positive as well as negative point of views. Besides the aspects (attributes of an entity that a review is about) of the opinions can be various, and therefore, it is essential to separate the mixed opinion reviews.
2. Next, you need to follow the semantic orientation approach of sentiment analysis to build a domain sentiment lexicon that is used to determine the polarity of the document. Emotional lexicons contain words that tend to be emotional. Because the domains are different, the words are used differently and the mood is reversed for each domain. Therefore, emotional lexicons for sentiment analysis are key to more accurate results.

3. In addition, online product reviews include various aspects. Therefore, the third goal is not to predefine the product, but to extract aspects of the product within the review and identify opinions about them. Achieving these three objectives provides a consistent framework for sentiment analysis proposed in this study aimed at improving sentiment classification performance and providing in-depth analysis at the aspect level.

### 1.3 Layout of the dissertation

**Chapter 1:** Introduction – In this chapter of the study I have given a brief introduction about the study which includes the introduction of the study aims and objectives of the study and the layout of the study.

**Chapter 2:** Literature Review – This chapter is all about the dispassionate review of the existing research in the sentiment analysis field. This has been again divided into relevant topics such as Related Work, Identified Challenges and Gaps, Recommendations and Future Research Directions and the Conclusion

**Chapter 3:** Background – This chapter includes the technical detailed discussion of the implementation in this study, like algorithms used for study, evaluation measures used in the study to measure the result.

**Chapter 4:** Methodology – As we know the methodology is all about the technical procedure of implementation and the execution for the study. In this I have discussed the data preparation, data extraction and data visualization and the implementation of the models. Also measures the result of the models implemented.

**Chapter 5:** Result and discussion –As name says here in this chapter of the study the detailed discussion of the result has been held and it shows the comparison of the result of the different models and summarize the result of the study.

**Chapter 6:** Conclusion – Here in the conclusion chapter, I have discussed about the conclusion that what has been done in the study

## 2 Background

### 2.1 Technical Details

In this section of the study, I am going to explain the details about the multiple techniques were used at the time of implementation where the accuracy and the other parameter for the evaluation of the model has been discuss in the study for different machine learning models.

Sentiment analysis falls under the supervised machine learning and it evaluates the formats of the data describe below:

Evaluation of the text formatted content which is may be the content on the internet for particular topic like “the details about daily news or the details about any educational content eg: details about the data science topics”. Feedbacks on the internet for the products like in our study we are using review as a text where people give their feedback for the product or feedback is for any organization, product, services. Suitability which defines the particular service, product is suitable for the person or the user who wish purchase it and this can achieve with the help of no of different approaches and in this study, I have included mostly used three as describe below:

#### 2.1.1 Lexicon Approach or ruled based approach

This works on the set of the predefined where there is no need to use machine learning algorithms or the model there are several python libraries SentiWordNet, Vader, TextBlob with the help of which the lexicon approach has been implemented.

## 2.1.2 Machine learning approach

This is the mostly used approach. In this study I have use the same approach as this approach provides the better result in case of accuracy of the model also there are no of algorithms we can used under this approach. In case of machine learning approach Sentiment analysis works on the evaluation of the different of the classified data which include four classification types describe below.

### 2.1.2.1 Binary Classification

It is working on 2 label classes like "Positive" and "Negative", "Yes" or "No", "Male" or "Female", "Spam", "Not spam" and this should be mapped to the numerical values as 2 and 1 respectively and that numerical values must be numeric which is called as label encoding.

The algorithms which are used to work on the binary classification data as follows:

Naive Bayes

Logistic Regression

1. K-Nearest Neighbors: KNN
2. Decision Trees
3. SVM-Support Vector Machine
4. Random forest classifier

### 2.1.2.2 Multiclass Classification

multiclass classification is not like binary classification multiclass classification can work on the particular range of the known classes like if we will take the example of fruits then it labels it into "Apple" Guava", " orange", " Kiwi" and so on.

The algorithms which are used to work on the binary classification data as follows:

1. Naive Bayes
2. Logistic Regression
3. K-Nearest Neighbors: KNN
4. Decision Tree.
5. Random Forest.
6. Gradient Boosting
7. SVM-Support Vector Machine.

### 2.1.2.3 Multi-label classification

To understand the multi label classification the best example is to "identifying the given images is of cat, or a dog or a Lion or a tiger identifying the image is also called as photo classification.

Multi label Gradient Boosting

Multi label Decision Trees

Multi label Random Forests

### 2.1.2.4 Imbalanced Classification

In imbalance classification the number of records belongs to each class is distributed unequally. Examples are fraud detection, medical diagnostic test because this type of problem statement requires specialized calculations or the techniques. Below are the algorithms that has been used to solve the multi label classification problems.

Cost-sensitive Logistic Regression.

Cost-sensitive Decision Trees.

Cost-sensitive Support Vector Machines.

## 2.1.3 Deep Learning

Deep learning is basically a subset of machine learning. Deep learning is mostly about the neural network. Deep learning or the neural networks come in picture while thinking how the human brain works as in case of thinking human brain is having the

most powerful brain on the earth so Deep learning is basically designed to mimic the human brain means to make the machine learn how the human brain being learn. The example is AlphaGo.

**In deep learning there are three main techniques:**

1. ANN
2. CNN
3. RNN

1. ANN is nothing but artificial network which works on tabular form of the data. whatever task with the tabular form of data has been done by Machine learning also done by Artificial neural network.

2. CNN is convolutional neural network which works on the images, videos data collection.

Some of the applications of the convolutional neural network are

1. Image classification
2. Object detection
3. Object Segmentation
4. Tracking
5. GAN

3. RNN is recurrent neural network works on the text and the time series data. This is the technique of the deep learning which works on the sentiment analysis of the NLP.

Some of the techniques of the RNN as follows:

1. RNN
2. LSTM RNN
3. Bidirectional LSTM RNN
4. Encoder-Decoder
5. Transformers
6. BERT
7. GPT 1, GPT 2, GPT 3

## 2.2 Feature Extraction

Feature extraction is the process in which the selection of the required features from the dataset has been performed for training the model. selecting the exact data to prepare the model is very important as the accuracy of the model is dependent on the feature selection and it impacted the result may be will get the incorrect result in the accuracy and may calculate the wrong measure parameters like recall, precision, f1-score.

Initially the dataset used in the study is raw and it has so many stop words [4.1] which is giving the same result for both positive and negative reviews.

Also, to make machine learning model understand the data should be in vector format. And to perform this vectorization of the text data the CountVectorizer and TFIDF vectorizer has been used in this study discussed in section [4.2.3]

## 2.3 Evaluation Measures

As we know every machine learning model is have an evaluation measure same as that the classification problems have an evaluation measure which as describe below:

### 2.3.1 Accuracy

The formula for accuracy is:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

Or, It can be defines as the total number of correct classification divided by the total number of classifications.



### 2.3.2 Recall

Recall is useful for measuring the completeness of the classifier. The higher recall value gives less value for the false negative.

The formula for recall is

$$\text{Recall} = \frac{TP}{TP + FN}$$

Or, as per the name, it is a measure of: from the total number of positive results how many positives were correctly predicted by the model. It shows how relevant the model is, in terms of positive results only.

### 2.3.3 Precision

Precision is a measure of amongst all the positive predictions, how many of them were actually positive. The formula for precision is:

$$\text{Precision} = \frac{TP}{TP + FP}$$

Precision use for the calculating the classifier exactness. If the value of precision is high it means it is giving less value for false positive

### 2.3.4 F1 Score

The metric that use both the Precision and Recall for evaluating a model. One such metric is the F1 score. F1 score is defined as the harmonic mean of Precision and Recall. The mathematical formula is:

$$\text{F1 score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

### 2.3.5 Specificity

Specificity is nothing but the true negative rate prediction. It represents or gives the model specification or how much specify model is for predicting the true negative.

The mathematical formula is:

$$\text{Specificity} = \frac{TN}{TN + FP}$$

### 2.3.6 ROC (Receiver Operator Characteristic)

As we know that the classification works on the basis of probability concept. and the value of the probability occurs between 0 and 1. Zero id for no probability occurs and one is for there is certain occurrence.

In real word scenario like when working with the real time dataset most of the time we get the decimal values which are between 0 and 1 like 0.90,0.38,0.80,0.69. some of the time only we get the perfect values like 1 and 0 now here the question arises that if the binary values are not getting then how to determine the class in the classification problem?

And then the term threshold the concept of the threshold comes in picture. The threshold is having a fixed value which is set and any value of the probability below the threshold value is consider as the negative value and the value is above the threshold value is positive value.

The threshold value has been decided on the basis of the requirement.

The ROC curve plotted for all the classification threshold an it plots two parameter which are true positive rate which is we can say recall and false positive rate.

$$\text{TPR} = \frac{TP}{TP + FN} \text{ and,}$$

$$FPR = \frac{FP}{FP + TN}$$

Typical ROC curve looks like:

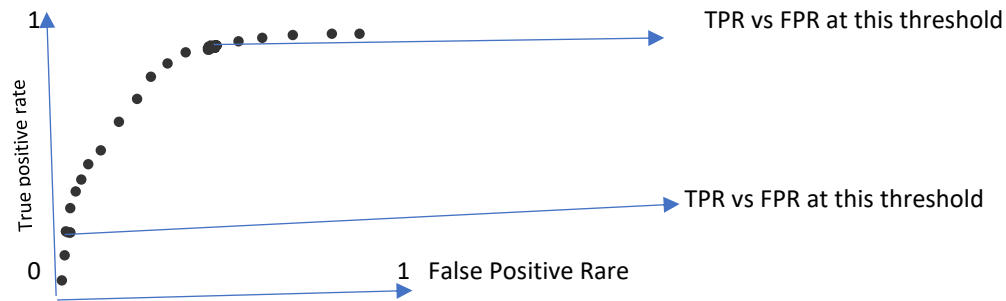


FIGURE 2.1: ROC CURVE

### 2.3.7 AUC (Area under ROC curve)

It helps to identify the model which is best suited for the problem statement using the ROC curve.

The model which is having the maximum area under roc is the best model

In below diagram the small dotted model is covering the maximum area so it is the best model we can consider

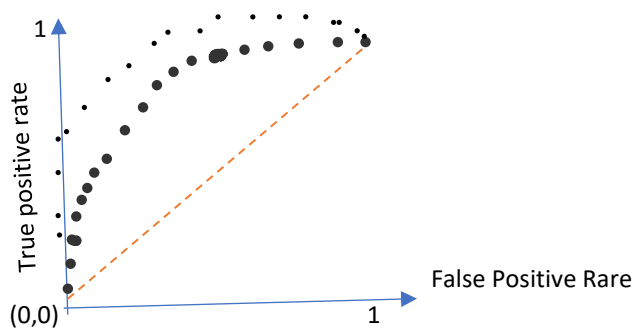


FIGURE 2.2: AUC

While working with real time dataset for the classification we can create no of models using different algorithms AUC is easy way to determine which is model is working best for the dataset given.

### 2.3.8 Confusion matrix

And the credibility of the model in case of the classification problem statement is measured using the confusion matrix generated, i.e., how accurately the true positives and true negatives were predicted. Below is the confusion matrix representation:

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

FIGURE 2.3: CONFUSION MATRIX

Fig: confusion matrix representation.

Where,

- **TP-True Positive:** It is the count of the result which is predicted as positive and the actual result also positive.
- **TN -True Negative:** It is the count of the result which is predicted as negative and the actual result also negative.
- **FP -False Positive:** It is the count of the result which is predicted as positive and the actual result is negative.
- **FN-False Negative:** It is the count of the result which is predicted as negative and the actual result is positive.

The Credibility of the model is based on how many correct predictions did the model do

## 2.4 Models Used in the Study

In this study, I have used Machine learning Binary Classification techniques to work on the sentiment analysis of the customer review in which dataset defines two classes which is nothing but binary classes" Positive" and "negative" and is mapped (label encoding) to the numerical values as 2 and 1 respectively.

As discussed above to work on the binary classification there are the algorithms listed below has been used in this study:

- Naive bayes
- Logistic regression
- SVM
- Decision Tree
- KNN
- Random Forest Classifier

### 2.4.1 Naive bayes

It works very well for the supervise machine learning classification problems. Naive Bayes Algorithm is work on the basis of Bayesian

$$P(\text{label} | \text{text}) = \frac{P(\text{label}) * P(\text{text1}|\text{label})...P(\text{textn}|\text{label})}{p(\text{text})}$$

thermos as the dataset used in this study having the 2 labels as

positive and negative which nothing but the classes and the features in the dataset is the text nothing but the “Text” column in the dataset.

And for news headlines there are label and headline\_corpus. Naive bayes works on the Naive assumption like the independent probability of the given words from other word, so the probability of label (positive or negative) from the given word of the review calculated as given below:

There are two types of the naive bayes Bernoulli, Gaussian and Multinomial all of them work differently to extract the features from the documents here in this study I am using the Multinomial naïve bayes technique.

When I have implemented the Naive, I have measured the different parameters for positive and negative reviews of the model given below:

1. No of features
2. Accuracy
3. Precision
4. Recall
5. F1-score
6. Support

**Advantages:** Naive Bayes is executed fast for both train and predicted as they do not have to learn to create separate classes.

Naive Bayes provides a direct probabilistic prediction.

Naive Bayes is often easy to interpret.

Naive Bayes has fewer (if any) parameters to tune

**Disadvantages:** The algorithm assumes that the features are independent which is not always the scenario

Zero Frequency i.e. if the category of any categorical variable is not seen in training data set even once then model assigns a zero probability to that category and then a prediction cannot be made.

## 2.4.2 Logistic regression

Logistic regression is a classification algorithm for supervised Machine learning. I have implemented the Logistic regression. From the name of the algorithm some people say it’s a wrong naming convention or it is using the regression line to find the value and for that value it finds the probability so its correct so that’s altogether different discussion.

The objective of the logistic regression is to find out the class for the given input in other words it finds the probability to identify the given input is belongs to or close to which class. In case of this study, it is finding the probability to identify the “Text” belongs to class “2” which is nothing but positive or class “1” which is nothing but Negative.

Logistic regression works with the help of probabilistic function which is nothing but the sigmoidal function which is  $\sigma(t) = \frac{1}{1+e^{-t}}$  sigmoid function’s range is bounded between 0 and 1 and this is why it is useful in calculating the probability of the logistic function. Mathematical representation of sigmoidal function:

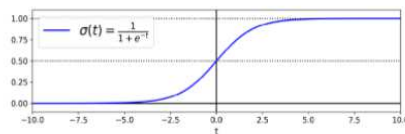


FIGURE 2.4: SIGMOIDAL FUNCTION

and straight-line equation which is  $Mx + C$ . straight line because it divides the binary classes with the help of straight line and the equation of the straight line is nothing but  $Y = Mx + C$  and after getting the probability  $\sigma(t)$  it compares it with the threshold and decides the class for the given input.

for ex. In our case 1 if  $P(x) < 0.5$

Or  $2P(\kappa) \geq 0.5$

And 0.5 is nothing but the threshold discussed in section no 2.3.6. Threshold is not the constant sometime it may be 0.8 like if  $\sigma(t) < 0.8$  it will be for 1 and  $\sigma(t) \geq 0.8$  it will be for 2. Threshold basically depends on the problems statement.

### 2.4.3 SVM

Support vector machine. It solves both the supervised machine learning classification and the regression problems statements. In the case of supervised classification problem, the main aim of SVM i.e Support vector machine is separate classes into two parts with the help of hyperplane as this is the Binary classification study. Hyperplane is nothing but the decision boundary which separate the given dataset with labels like in our case positive, negative points as shown in Figure 2.5

Support vector Machine in case of this study separate the class in two parts which is positive and negative or 1 and 2 respectively as shown in below diagram. support vector machine make sure that when it created the hyper plane it also creates two margin line that is parallel to hyperplane which is shown by blue dotted line in the below diagram and this two lines are having some distance so that it can be linearly separable from both the classification points and while creating this margin line it make sure that it passes through one of the nearest possible classification points as shown in below diagram both the marginal line passes through one of the positive point and negative points and the distance between this two line is called Marginal distance.

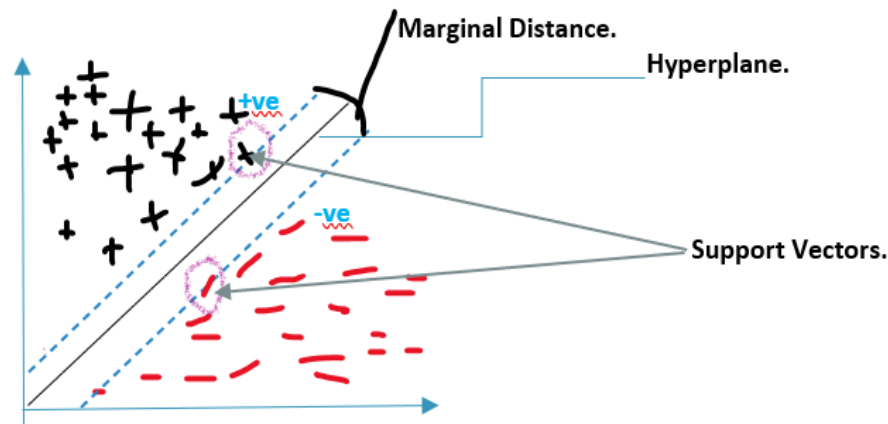


FIGURE 2.5: SVM

Support vector Machine is slow and the reason behind this is the libraries of the Support vector Machine algorithms not incremental. They need the whole dataset in the machine RAM all a time and then if the data is having millions of rows, then it executes slow.

### 2.4.4 Decision Tree

Decision tree algorithm is most popular algorithm in Machine learning. Decision tree algorithm is an algorithm which works on both the cases classification and regression this is one of the versatile algorithm it is known to work on the dataset which are complex in structure also its easy to read understand and this all are the reason behind its popularity in case of use. As per the name of the algorithm this divides the entire dataset into a tree structure by following some condition and the rules.

Example: the basic and simplest example suppose customer wants to decide should buy the product according to the review of the product then decision tree can help to decide as per below diagram. There are some conditions. Positive review count. Positive review count if the Positive review count is greater than or equal to 40 then can buy the product else should not buy the product.

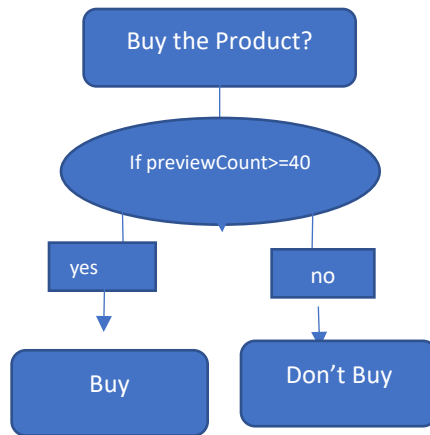


FIGURE 2.6: DECISION TREE EXAMPLE

1. It basically, on the given condition selects root node in above example a root node is if the positive review count is greater than 40.
2. Then, the root node is splits into child nodes by considering the given condition and in above example child node with label "No" do not fulfilled the given condition so its but obvious the customer should not buy the product.
3. The child node with label "Yes" fulfilled the given condition so customer can buy the product.

As decision tree work on both the regression and classification in case of regression it works on quantitative type of data in regression it works on the basis criteria of RSS. In case of classification, it works with the help of the error rate found in classification Entropy and Gini impurity.

#### 2.4.4.1 Entropy

Entropy helps to measure the purity of the splits or we can say that Randomness in the data has been measured with the help of entropy.

In case of binary classification, the out feature is "yes" or "No" and for constructing the binary tree there is use of ID3 algorithm and this says that the first step to select the best or correct attribute to split the decision tree and to decide which feature to select first the entropy technique comes in picture.

Entropy Formula:

$$E = -P * \log_2(p) - q * \log_2(q)$$

$P$  % of positive class and  $q$  % of negative class

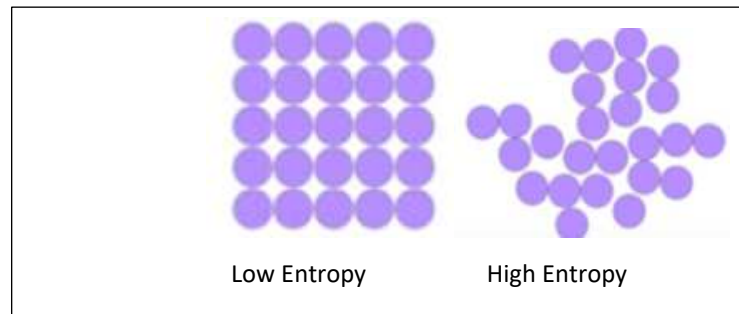


FIGURE 2.7: ENTROPY

#### 2.4.4.2 Gini Impurity

Gini impurity measure the occurrence of incorrectly labelled element of the dataset.

Gini Impurity Formula

$$GI = 1 - \sum_{i=1}^n (P_i)^2$$

#### 2.4.5 KNN

K- nearest neighbor is a supervised machine learning algorithm , KNN works in both the cases classification and regression KNN is the wonderful algorithm for solving non-linear classified data points like if the data points are disturbed in nonlinear manner can't draw a straight line and classify the datapoints then KNN can be use.

##### Working of KNN in classification

1. It first finds out the K-value: how many nearest data points to be consider in terms of distance and distance can be calculated by 2 parameters Euclidian distance and Manhattan distance.
2. Calculate the distance between data points select the points

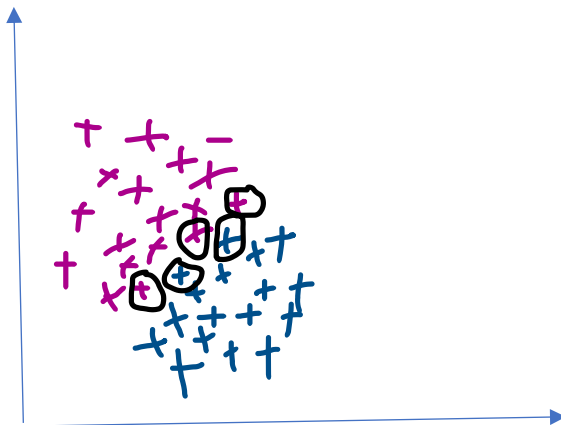


FIGURE 2.7: KNN

**Euclidian distance:** suppose there are two data points at the particular location like  $p1(x_1, y_1)$  and  $p2(x_2, y_2)$  so to calculate distance between these two points the formula for **Euclidian distance** is

$$ED = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

**Manhattan distance:** to calculate the two data points at the particular location like  $p1(x_1, y_1)$  and  $p2(x_2, y_2)$  so Manhattan distance create the right angle triangle and then add the distance of this two data points.

$$MD = \sum_{i=1}^n |x_i - y_i|$$

As per the fig 2.7 consider the K-value =5 and then as per this k-value=5 it has been find that points belongs to one class which is pink color and 2 points belongs to other class which is blue in color.

So in this KNN algorithm whichever will be having the highest points as per k value her we have pink it terms to the neighboring point.

Now if the new point or new data comes then it will consider as the pink category point as because the maximum number of categories got is pink color categories here.

3. Assign the class to the newly added points which is having maximum value.

There are different ways with which KN can be performed

- 1 k-Dimensional Tree (kd tree)
- 2 Ball Tree

#### **Advantages and Disadvantages of k-NN Algorithm**

Advantages:

- It is working for regression as well as classification.
- The implementation is very easy and simple.
- It is very easy to understand the math's behind the KNN.

Disadvantages:

- Optimum K-value finding
- Lot of time has been taking in computing the distance between two points.

#### **2.4.6 Random Forest Classifier**

Random forest is a supervised Machin learning algorithm it can be work for both regression and the classification problem statement. It is working with the help of concept of ensemble learning.

ensemble learning combines the multiple classifiers.

Random forest classifier uses the decision tree with the help of row sampling and feature samples (column).



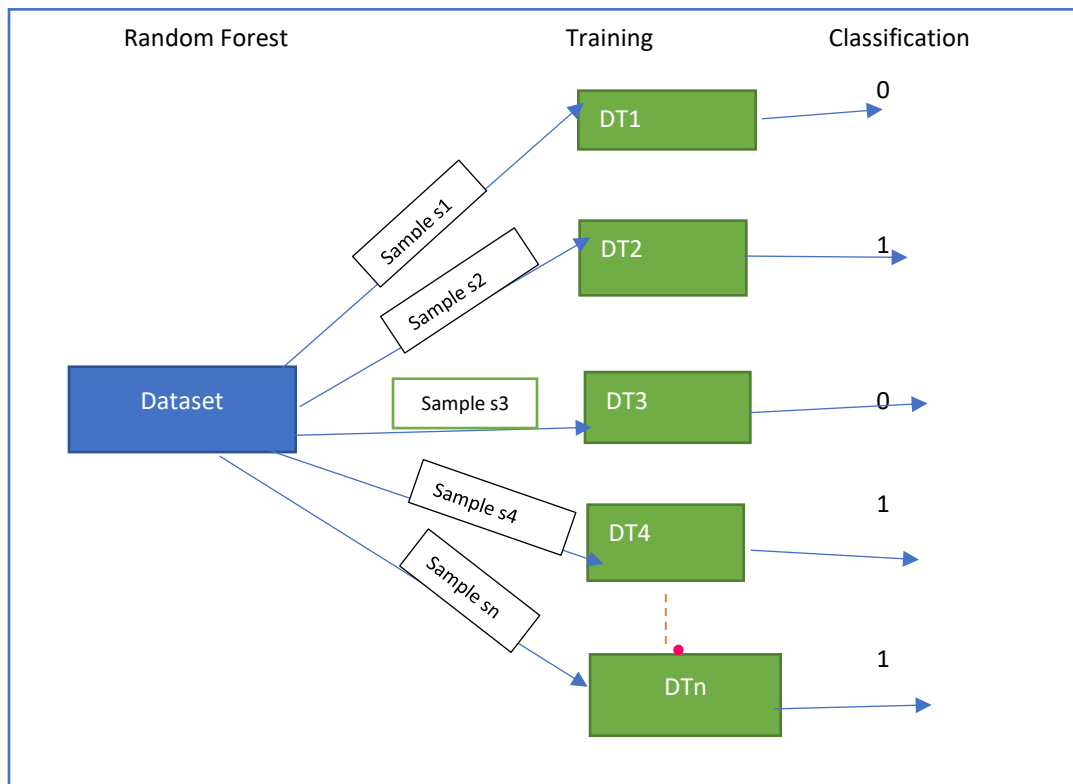


FIGURE 2.8: RANDOM FOREST CLASSIFIER

The data sampling is done here in the form of Decision Tree DT1, Decision Tree DT2, Decision Tree DT3, Decision Tree DT4 and so on Decision Tree DTn which is nothing but bagging and then each Decision Tree gives the output features as 0,1,1,1,1 respectively and the finally it needs to be aggregate with the majority vote has been used.

So here the majority vote is 1 as the more no of decisions tree gives the output 1. And this is nothing but how the random forest classifier work.

### 3 Literature Review

Sentiment Analysis is a one of the main fields of NLP that tries to identify and extract opinions within a given text across blogs, reviews, social media, forums, news etc. Natural language processing (NLP) refers to the branch of computer science more specifically, the branch of artificial intelligence which give computers the ability to understand text and spoken words in the similar way humans can

Over the years, the usage of Microblogging sites, social networking application as well as website and reviews website are increased. Due to which researchers and developers have explored many NLP areas on these platforms.

Twitter is the most used social networking platform and the micro blogging services, which supplies more data. Sentiment analysis is a job that focuses on polarity detection and the validation of emotion toward an entity, which could be an individual, topic, and event.

Generally, the aim of sentiment analysis is to find user's view, identify the sentiments they express, then classify their polarity into positive, negative, and neutral categories. Sentiment analysis systems apply NLP and ML methodologies to discover, retrieve, and refine information and opinions from infinite amount of textual information.

There are 3 main approaches used in SA that is Lexicon-based, machine learning and deep learning.

Machine learning framework include of traditional approaches like deep learning and conditional random field approaches.

However, the rule-based models consist of lexicon-based approach. Object detection, network optimization, image recognition, system security, sensor networks and transportation formulated on deep learning methods, which are commonly utilized across many different fields. A handful researchers have integrated deep learning and machine learning algorithms into text sentiment analysis by sentiment lexicon formulation and best results are obtained.

#### 3.1 Related Works

- A. In the year of 2019, Saad and Yang [1] have given a tweet sentiment analysis developed on ordinal regression along with MI algorithms. The model advised by them, contain pre-processing tweets as first step and with the feature extraction model, an effective feature was produced. The methods like SVR, RF, Multinomial logistic regression (SoftMax), and DTs were employed for categorizing the sentiment analysis. Additionally, twitter dataset was used for experimenting the advised model. The test results shown that the advised model has attained the more precision, and DTs were pull off well when compared to other methods. During 2018, Fang et al. [2] have proposed multi-strategy sentiment analysis models making use of the semantic fuzziness to fix the issues. The results have illustrated that the proposed model has attained high accuracy and efficiency.
- B. In the year of 2019, Afzaal et al. [3] have proposed a novel approach of aspect-based sentiment classification, which identified the features in a accurate manner and attained the best classification accuracy. Besides, the program was developed as a mobile application, which assisted the travelers in identifying the best Motel, hostels and restaurants in the town. And the suggested model was researched using the real-world data sets. The results have proven that the presented model was successful in both recognition as well as classification
- C. In the year of 2019, Feizollah et al. [4] have worked on tweets related to two halal products for-instance halal cosmetics and halal tourism. By making use of Twitter search functionality, Twitter information was taken out, and a new model was deployed for data filtering. Slowly, with the assistance of deep learning models, a test was executed for computing and assessing the tweets. Additionally, to improve the accuracy and building prediction techniques, RNN, CNN, and LSTM were implemented. From the conclusion, it was seemed that the integration of LSTM and CNN attained the best precision.

- D. In the year of 2018, Mukhtar et al. [5] have worked on the sentiment analysis to the Urdu blogs acquired from many domains with Supervised Machine learning and Lexicon-based models. In Lexicon-based models, a well-fuctioning Urdu sentiment analyzer and an Urdu Sentiment Lexicons were deployed, even though, DT, KNN, and SVM were deployed in Supervised Machine learning algorithm. The data were consolidated from the two sources for achieving the best sentiment analysis. Based upon tests performed, the results were shown in the Lexicon-based model was more accurate to the supervised machine learning algorithm.
- E. In the year of 2020, Kumar et al. [6] have introduced a hybrid deep learning method named ConVNetSVMBoVW that in cooperated with the real-time data for forecasting the fine-grained sentiment. Beneficial to measure the hybrid polarity, an aggregation model was developed. Furthermore, SVM was used for training the BoVW to predict the sentiment of visual content. Ultimately, it was concluded that the advised ConvNet-SVMBoVW was surpassed by the conventional models.
- F. In the year of 2018, Abdi et al. [7] have presented a machine learning method to sum up the opinions of the users brought up in the reviews. The advised technique combined multiple kinds of features into a unique feature set for modelling accurate classification framework. Hence, a performance analysis was done for four best feature selection models for attaining the best fulfilment and seven classifiers for choosing the relevant feature set and recognized an effective machine learning algorithm. The advised technique was implemented in numerous datasets. The end results have given an idea of the combination of IG as the feature selection approach and SVM-based classification approach strengthened the performance
- G. In the year of 2019, Ray and Chakrabarti [8] have presented a deep learning algorithm to uproot the features from text and the user's sentiment analysis with respect to the feature. In opinionated sentences, a seven-layer Deep CNN was deployed for labeling the features. Beneficial to enhance the performance of sentiment scoring and feature extraction models, the writers combined the deep learning techniques using a set of rule-based models. Eventually, it was seen that the advised method performed the best accuracy. During 2019, Zhao et al. [9] have suggested a novel image-text consistency driven multimodal sentiment evaluation method, which inspected the correlation among the text and image. Slowly, a multi-modal adaptive sentiment analysis model was employed. With the conventional SentiBank model, the mid-level visual features were extracted and those were deployed for representing the International Journal of Advanced Science and Technology Vol. 29, No. 7, (2020), pp. 1462-1471 ISSN: 2005-4238 IJAST 1465 Copyright © 2020 SERSC visual theories by merging the non-identical characteristics like social, textual, and visual features for presenting a machine learning model. The advised model has gained best accuracy when compared to traditional models.
- H. In the year of 2019, Park et al. [10] have introduced a semi-supervised sentiment-discriminative objective to resolve the issue by documents partial sentiment data. The advised method not only mirrored the partial data, but also secured the local constructions gained from real data. The advised technique was assessed on real time datasets. The outcomes have shown that the advised model was performing well. During 2019, Vashishtha and Susan [11] have performed the sentiment correlated to social media posts by a new set of fuzzy rules consisting of several datasets and lexicons. The developed model integrated Word Sense Disambiguation and NLP models with a new unsupervised fuzzy rule-based model for classifying the comments into negative, neutral, and positive sentiment class. The experiments were performed on 3 sentiment lexicons, four existing models, and nine freely available twitter datasets. The end results have shown that the introduced method has the best results.

- I. In the year of 2019, Yousif et al. [12] have introduced a multi-task learning model based on CNN and RNN. The framework of the advised model was helpful for denoting the citation context and feature extraction was done in an automated way. Keeping in mind two freely accessible datasets, the suggested method was assessed. The end results have shown that the suggested technique was improved than traditional models. During 2020, Hassonah et al. [13] have proposed hybrid machine learning algorithm for enhancing the sentiment analysis, since a classification approach was built based on "Positive, Negative, and Neutral" classes with SVM classifier, simultaneously two feature selection methods were integrated by the MVO and Relief models. Additionally, Twitter data was deployed for assessing the suggested model. The outcomes proved that the suggested technique was performing better than traditional techniques.
- J. In the year of 2020, Xu et al. [14] have recommended a NB model for large-scale Ecommerce and multi-domain platform product review classification of sentiment. Accordingly, the parameter evaluation process was extended in NB for continuous learning fashion. Slowly, to polish up the learned distribution on the basis of three types of theories, multiple ways were initiated to acquire the best performance. The outcomes proved that the proposed model has gained high accuracy in Amazon product and movie review sentiment datasets.
- K. In the year of 2018, Smadi et al. [15] have recommended existing models based upon supervised machine learning algorithms for identifying the limitation of feature-based sentiment analysis of Arabic hotel's review. Furthermore, SVM and Deep RNN were trained and developed along with word, lexical, morphological, semantic, and syntactic features. The reference dataset of Arabic hotel's review dataset was used for evaluating the suggested model. The results proved that SVM was performing better compared to RNN model. During 2020, Maqsood et al. [16] have explained the impact of multiple events happened in the year 2012-2016 on stock markets. Twitter dataset was deployed for computing the sentiment analysis to these events. The dataset has millions of tweets, which were employed to define the event sentiment.
- L. In the year of 2019, Abdi et al. [17] have proposed a deep-learning-based method for classifying the sentiments of the user mentioned in reviews. Additionally, a deep learning method was a unified feature set that was representative of sentiment shifter rules, word embedding, sentiment knowledge, linguistic and statistical knowledge has not been continuously inspected for a sentiment analysis. In addition to, the advised model used RNN that included of LSTM for considering the advantages of sequential processing and conquered multiple issues in traditional algorithms. During 2020, Park et al. [18] have invented a deep learning method for enhancing performance. To improve the performance, two major questions have come into picture. The content attention was required for being sophisticated for combining numerous attentions results non-linearly and presume the whole context for mentioning the complex sentences. The outcomes proved that the suggested model was attained as the best performance.
- M. In the year of 2019, Bardhan et al. [19] have proposed a quasi-qualitative technique to recognize the underlying the effects of gender mainstreaming in SRH management. Moreover, to explore the stakeholder issues, verbal narratives from semi-structured interviews and concentrated on group discussions. For resolving the emotions over the stakeholders, sentiment analysis with machine learning algorithm of NLP is employed. During 2017, Araque et al. [20] have suggested a deep learning model for improving

International Journal of Advanced Science and Technology Vol. 29, No. 7, (2020), pp. 1462-1471 ISSN: 2005-4238 IJAST 1466 Copyright © 2020 SERSC

the performance by integrating the existing surface models with deep learning models based upon manually extracted features. Similarly with the assistance of linear machine learning and word embeddings methods, a deep learning-based sentiment classifier was introduced. Almost, 7 datasets were used to demonstrate the efficiency of the proposed model. The outcomes have confirmed that the presented model was more effective compared to conventional methods.

### 3.2 Identified Challenges and Gaps

There are some major challenges and gaps faced during the study of different sentimental analysis. Below are discussed some of them.

- **Fine-grained sentiment analysis:** Many studies have concentrated their observation on a complete review to examine a sentiment rather than going deeper to recognize fine-grained teaching/learning-related views and sentiments related with them.
- **Figurative language:** To associate figurative speech, for-instance sarcasm and irony, from student opinion text is lacking and needs more investigation.
- **Generalization:** Many Popular techniques are domain-specific and hence do not accomplish well in different domains.
- **Complex language constructs:** There is an inability to handle complex language involving constructs for-instance unknown proper names, double negatives, abbreviations and words with dual and numerous meanings.
- **Representation techniques:** There is a need of research effort on the use of general-purpose word embedding together with contextualized embedding methods.
- **Scarcity of publicly available benchmark datasets:** There is absence of benchmark datasets and an inadequate dataset size. Even though there are some open datasets available, no benchmark dataset is available that is functional for testing deep learning methods due to the minimum number of samples those datasets supply.
- **Limited resources:** There is a shortage of resources such as lexica, corpora, and dictionaries for low-resource languages (most of the studies were conducted in the English or Chinese language).
- **Unstructured format:** basically, datasets discovered in the studies considered in this survey paper were unstructured. Recognizing the key entities to which the views were directed is not practical until an entity extraction method is applied, which makes the existing dataset's applicability restricted.
- **Unstandardized solutions/approaches:** We noticed in this review study that a wide range of frameworks, packages, tools and libraries are applied for sentiment analysis.

### 3.3 Recommendations and Future Research Directions

This segment provides several suggestions, recommendations and proposals for worthy and effective systems that may help in developing generalizable solutions for sentiment analysis in the education domain. We observe that the recommendations appropriately address the challenges identified in above section.

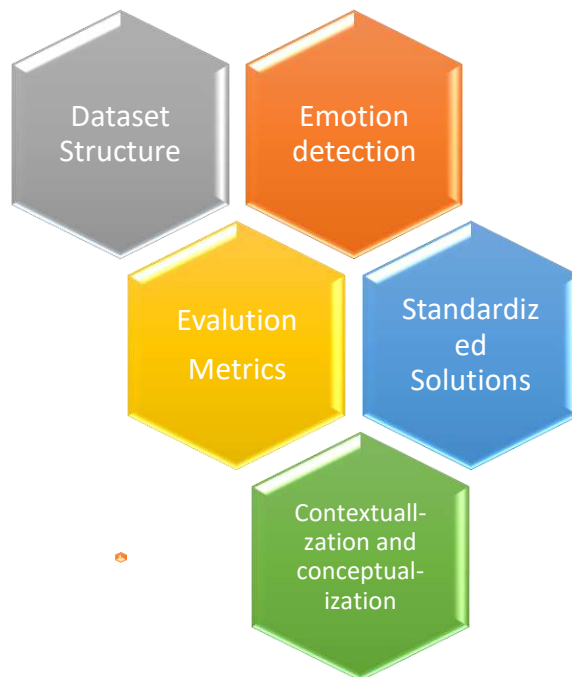


Image : Recommendations for developing effective sentiment analysis systems.

- **Datasets Structure and Size:**

There is a demand for a structured format to represent feedback datasets, even if they are collected at the sentence level or document level via a survey or a quiz form. A structured format in either an XML or a JSON file would be very useful to regulate dataset generation for sentiment analysis in this domain. Moreover, there is a need to integrate the meta-data acquired at the time of the feedback responses. The meta-data would assist to give a illustrative analysis of the feedbacks conveyed by a group of people for a given subject (aspect). Additionally, more than half (56.7%) of the datasets used in the reviewed papers were of a minimum size, with only 5000 samples or less, which influences the dependability and significance of the results. Moreover, many of these datasets are not globally available, meaning that the outcomes are not reproducible. Hence, we strongly suggest the collection of huge scale labeled datasets to develop universal deep learning models that could be utilized for multiple sentiment analysis jobs as well big data analysis in the education domain.

- **Emotion Detection:**

We discovered only a little number of articles concentrated on emotion detection. We feel, there is a bigger need to accommodate the emotions expressed in opinions to identify finer and resolve the issues related to the target subject, as has been explored in many other text-based emotion detection works. Moreover, there are standard globally available datasets such as ISEAR (<https://www.kaggle.com/shrivastava/isearsdataset>), and SemEval-2019 that can be used to educate deep learning models for text-based emotion detection jobs utilizing the Plutchik model merged with emoticons. Human beings frequently use emoticons to identify emotions; hence, one of the aspects that researchers could investigate is to build use of emoticons to address the emotions expressed in an opinion.

- **Evaluation Metrics:**

Study explained that researchers have used several evaluation metrics to count the capability of sentiment analysis systems and models. Moreover, a significant number of papers (27%) failed to provide the information regarding the metrics used to evaluate the performance of their systems. Hence, we need a special focus and emphasis should be placed on including the effective use of metrics in order to improve the transparency of Appl. Sci. 2021, 11, 3986 18 of 23 the

research results. Information retrieval evaluation metrics for example the precision, recall, and F1-score would be the best method for the performance evaluation of sentiment analysis systems depending on polarity datasets. Accuracy would be another metric that could be used to assess the performance of systems educated on balanced datasets. Statistic metrics for instance the Kappa statistic and Pearson correlation are other metrics that can be used to measure the correlation between the output of sentiment analysis systems and data labeled as ground truth. Furthermore, this could assist and advantage other analysts when managing comprehensive and comparative performance examines between different sentiment analysis systems

- **Standardized Solutions:**

We have proven record that the current landscape of sentiment analysis is specified by a vast range of solutions that are yet to develop as the field is obviously novel and rapidly expanding. These solutions were mostly (programming) language-dependent and have been used to achieve certain tasks i.e., tokenizing, part-of-speech etc. in divergent scenarios. Therefore, standardization will play main role as a means for promising the quality, safety, and reliability of the solutions and systems developed for sentiment analysis.

- **Contextualization and Conceptualization of Sentiment:**

Machine learning/deep learning approaches and methods developed for sentiment analysis should give more attention to embed the semantic context using lexical resources such as Wordnet, SentiWordNet, and SenticNet, or semantic representation using ontologies to capture users' feedback, thoughts, and attitudes from a text more efficiently. Additionally, state-of-the-art static and contextualized word embedding approaches for example fastText, GloVe, BERT, and ELMo should be further reviewed for exploration by researchers in this field as they have shown to perform well in other NLP-related programs.

### 3.4 Conclusion

Over the decade, sentiment analysis enabled by NLP, machine learning, and deep learning techniques has been attracting the attention of researchers in the various domain including Customer reviews and News Headlines, academic domain. In order to examine customer's attitude, feedback, and behavior towards several product and headlines, we provided an analysis of the related theory by applying a systematic mapping study technique. Particularly, in this mapping study, we selected many relevant papers and analyzed them with respect to different dimensions such as the investigated entities/aspects on the customer reviews and news headlines, the most frequently used bibliographical sources, the research trends and patterns, what tools were utilized, and the most common data representation techniques used for sentiment analysis. The present paper has developed the review of earlier contributions with various machine learning models using discrete information. The present review has explored 15 plus research works that covered various implementations employed for sentiment analysis. Initially, the assessment has concentrated on clarifying the contribution of every task and observed the type of machine learning algorithm utilized. The evaluation also focused in recognizing the type of data employed. Later, the environment utilized, and the performance metrics covered in each contribution was analyzed. Finally, the research gaps and challenges were mentioned that were useful for recognizing the non-saturated implementation for which the sentiment analysis was required in further research.

## 4 Methodology

This section of the study is all about the working of the model from end to end like for the architecture design to the result of the model.

### 4.1 Architecture of the study

Sentiment analysis is a supervised machine learning classification problem. Classification problems means the result in this is getting into categorical form like "Yes", "No" or Positive Negative, Male female and so on, so the Customer Reviews and News Headlines data gives the label as positive and negative so this dataset has been used to work in this study. In this section of the study, I will be looking into the dataset details which I have used, how the data pre-processing has been done. The dataset is of two classes "Positive" and "Negative" with the label 1 for Negative and 2 for positive for customer review dataset and label 0 and 1 for the News Headlines dataset. The program for the study has been designed to work or to follow the supervised machine learning classification model algorithm which are described in detailed in the [section 3.4](#) of this study. Here, Python 3.9.7 programming language has been. Below are some of the libraries use for the study.

The **seaborn library** is imported as sns. Seaborn is very good and most amazing library for graphical and statistical visualization. Seaborn is having lots of color and beautiful styling by default to make statistical charts in Python more attractive. The Seaborn library aims to visualize the central part of data understanding and exploration in a more attractive way. Built on the core of the **Matplotlib library**, it also provides a dataset-oriented API. The Seaborn is also tightly integrated with Panda's data structures, so you can easily switch between different visual representations of a particular variable to better understand the dataset provided.

**NumPy**, nothing but Numerical Python, is a library of multidimensional array objects and a collection of features for processing the arrays. You can use NumPy to perform mathematical and logical operations on arrays. Then I installed the **plotly** library.

In this study, I have implemented model with stopwords and without stop words: **A stop-word** is a set of words commonly used words in a language. These are the English words that does not add much meaning to the sentence. They can be safely ignored without sacrificing the meaning of the sentence. Examples of English stop words include "a", "the", "is", and "are". Stop-words are commonly used in text mining and natural language processing (NLP) to eliminate frequently used words. There add little or no meaning to the sentence. The meaning of a sentence can be perceived without these words too. For example, a search query in the context of a search system says "What is a preposition? This keeps a list of prepositions (which can be curated manually or automatically) and everything in the preposition list. You can do this by preventing the words in. In this example, you can omit the word "a" and leave only the word "preposition". This will rank the documents related to the topic higher in the search results. After calculating an overall frequency of the stop-words, I have use the matplotlib library to display the bar graph of the most frequent stop-words. It can be easily observed from the graph that 'of the', 'this book', 'in the', 'it is', 'this is' etc. are some of the very common stop words with the stop word 'of the' having a frequency of around 24 thousand.



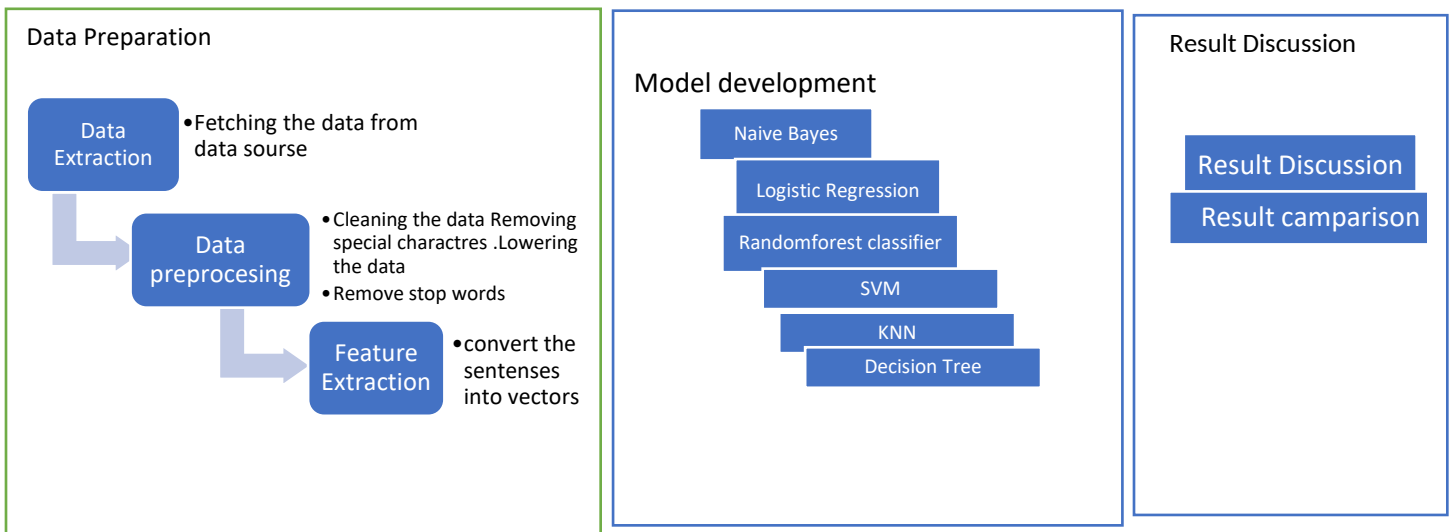


FIGURE 2.9: ARCHITECTURE OF STUDY

## 4.2 Data Preparation

Data preparation involves the several no of steps where we have to extract the data from the datastores then clean the data means removing the special characters lowering the data for good result select the required features or extract the features from the data set to make model understand the data.

### 4.2.1 Data Extraction

I have collected the dataset from the Kaggle website. The data that I am using is the Amazon data for Customer Reviews and stock news headlines data for News Headlines. After collecting the dataset from the Kaggle website, I have then divided the data into two datasets for train and test data. By using pandas library, read the 'test.csv' dataset with ISO-8859-1 encoding and store it in the train variable. Encoding the data to remove the special characters, otherwise the data will not work. The label 1 is for positive and the label 2 is for negative. Then by following the same procedure the 'test.csv' data is also read and stored in test variable.

**Train Dataset for customer reviews looks like:**

```
train=pd.read_csv("train.csv", encoding="ISO-8859-1",header=None, names=['Label', 'Title', 'Text'])
train.head() # lable is 1 for negative 2 for positive.
```

	Label	Title	Text
0	2	Stuning even for the non-gamer	This sound track was beautifull It paints the ...
1	2	The best soundtrack ever to anything.	I'm reading a lot of reviews saying that this ...
2	2	Amazing!	This soundtrack is my favorite music of all ti...
3	2	Excellent Soundtrack	I truly like this soundtrack and I enjoy video...
4	2	Remember, Pull Your Jaw Off The Floor After He...	If you've played the game, you know how divine...

IMAGE 4.1: TRAIN DATA

**Test Dataset for customer reviews looks like:**

```
test=pd.read_csv("test.csv", encoding="ISO-8859-1", header=None, names=['Label', 'Title', 'Text']) # encoding to remove the spe
test.head()
```

	Label	Title	Text
0	1	Bratz and Booze????	Ok, the Bratz dolls are supposed to be teens, ...
1	1	JUNKIIIIIIIIII	I bought this for my daughter's birthday, we d...
2	1	Flashy, but a HUGE disappointment!!	My daughter purchased this item with the money...
3	1	junk waste of money	My 7 year old daughter had to have this bought...
4	2	cute as can be	My daughter received this for christmas. It is...

IMAGE 4.2: TEST DATA

Dataset for News Headlines looks like:

```
df=pd.read_csv("StockNewsData.csv", encoding="ISO-8859-1") # encoding to remove the special characters it must otherwise data wil
df.head(5)
```

	Date	Label	Top1	Top2	Top3	Top4	Top5	Top6	Top7	Top8	...	Top16	Top17	Top18	Top19
0	2000-01-03	0	A 'hindrance to operations' extracts from the...	Scorecard	Hughes' instant hit buys Blues	Jack gets his skates on at ice-cold Alex	Chaos as Maracana builds up for United	Depleted Leicester prevail as Elliott spoils E...	Hungry Spurs sense rich pickings	Gunners so wide of an easy target	...	Flintoff injury piles on woe for England	Hunters threaten Jospin with new battle of the...	Kohl's successor drawn into scandal	The difference between men and women
1	2000-01-04	0	Scorecard	The best lake scene	Leader: German sleaze inquiry	Cheerio, boyo	The main recommendations	Has Cubie killed fees?	Has Cubie killed fees?	Has Cubie killed fees?	...	On the critical list	The timing of their lives	Dear doctor	Irish court halts IRA man's extradition to Nor...
2	2000-01-05	0	Coventry caught on counter by Flo	United's rivals on the road to Rio	Thatcher issues defence before trial by ...	Police help Smith lay down the law at	Tale of Trautmann bears two more retellings	England on the rack	Pakistan retaliate with call for video of Walsh	Cullinan continues his Cape monopoly	...	South Melbourne (Australia)	Necaxa (Mexico)	Real Madrid (Spain)	Raja Casablanca (Morocco)

IMAGE 4.3: NEWS HEADLINES DATA

## 4.2.2 Pre-processing Data

Data pre-processing is required most of the time to make the data quality good. In the real-world data most of the time data is not complete meaning it is having some missing column or the features or blank values or errors like empty data or null values. Data pre-processing helps in cleaning the data, arrange the data and well format the data, which makes the data suitable for machine learning models. In this case, data pre-processing is done using pandas. Pandas is a library created for the Python programming language for data manipulation and analysis. In particular, it provides data structures and operations for working with numeric tables and time series. Pandas are mainly used for data analysis.

1. For customer review as the data is very huge, so I have taken a subset of 50000 records of train dataset and 10000 rows from the test dataset and combined them. The 'pd.concat' function joins the two columns of data and puts them one after the other also combined the Title and the Text columns and put them in a single column under the heading Text and dropped the Title column now the dataset looks like below image .

```

]: #combining the two coloum text and title
df['Text'] = df['Title'] + '. ' + df['Text']
df.drop('Title', axis=1, inplace=True)
#here the Label is categories into 2 class 1 for negative and 2 for positive.
df.head()

```

```

]:

```

	Label	Text
0	1	Horrible - DO NOT WASTE YOUR TIME!. I read a g...
1	1	Sound bite literature. This book has about as ...
2	1	No Information. The book held no substantive i...
3	1	3-User license is non-renewable as a 3-user li...
4	1	waste of time. The book starts out pretty grip...

IMAGE 4.4: CLEANED DATA

2. For News Headlines data I have divided the data in to the train and the test dataset according to year of news and the dataset looks like below:

```

#divided the data in to the train and the test dataset according to year of news
train=df[df['Date']<'20150101']
test=df[df['Date']>'20141231']
train.head()

```

	Date	Label	Top1	Top2	Top3	Top4	Top5	Top6	Top7	Top8	...	Top16	Top17	Top18	Top19
0	2000-01-03	0	A 'hindrance to operations' extracts from the leaked reports	Scorecard	Hughes' instant hit buoys Blues	Jack gets his skates on at ice-cold Alex	Chaos as Maracana builds up for United	Depleted Leicester prevail as Elliott spoils Everton's party	Hungry Spurs sense rich pickings	Gunners so wide of an easy target	...	Flintoff injury piles on woe for England	Hunters threaten Jospin with new battle of the Somme	Kohl's successor drawn into scandal	The difference between men and women
1	2000-01-04	0	Scorecard	The best lake scene	Leader: German sleaze inquiry	Cheerio, boyo	The main recommendations	Has Cubie killed fees?	Has Cubie killed fees?	Has Cubie killed fees?	...	On the critical list	The timing of their lives	Dear doctor	Irish court halts IRA man's extradition to Northern Ireland
			Coventry	United's	Thatcher issues	Police help	Tale of	Pakistan	Cullinan						

And at the next step, for both the dataset, I have worked on cleaning the data by removing the spaces, special characters and null values from the dataset. I have used pandas library and the Regular expression for cleaning the dataset. So now, the prepared data with the text and the corresponding labels. All the records in the data is being converted to lower case.

And for News headline dataset I have find out the corpus of the news headlines for count vectorizer to convert the sentences into vector or document matrix as shown below:

```

headlines_corpus=[]
for i in range(0,len(data.index)):
    headlines_corpus.append(' '.join(str(x) for x in data.iloc[i,0:25]))

```

```
headlines_corpus
```

```

['a hindrance to operations extracts from the leaked reports scorecard hughes instant hit buoys blues jack gets his skate
s on at ice cold alex chaos as maracana builds up for united depleted leicester prevail as elliot spoils everton s party hun
gry spurs sense rich pickings gunners so wide of an easy target derby raise a glass to strupar s debut double southgate strik
es leads pay the penalty hammers hand robson a youthful lesson saints party like it s wear wolves have turned into lamb
s stump mike catches testy gough s taunt langer escapes to hit flintoff injury piles on woe for england hunters threaten
jospin with new battle of the somme kohl s successor drawn into scandal the difference between men and women sara denver nur
se turned solicitor diana s landmine crusade put tories in a panic yeltsin s resignation caught opposition flat footed russi
a n roulette sold out recovering a title',
'scorecard the best lake scene leader german sleaze inquiry cheerio boyo the main recommendations has cubie killed fees h
as cubie killed fees has cubie killed fees hopkins furious at foster s lack of hannibal appetite has cubie killed fees a
tale of two tails i say what i like and i like what i say elbows eyes and nipples task force to assess risk of asteroid coll
ision how i found myself at last on the critical list the timing of their lives dear doctor irish court halts ira man s extra
dition to northern ireland burundi peace initiative fades after rebels reject mandela as mediator pe points the way forward t
o the ecb campaigners keep up pressure on nazi war crimes suspect jane ratcliffe yet more things you wouldn t know without th
e movies millennium bug fails to bite',
'coventry caught on counter by flo united s rivals on the road to rio thatcher issues defence before trial by video police h
elp smith lay down the law at everton tale of traumann bears two more retellings england on the rack pakistan retaliate with
call for voley of walsh cullinan continues his cape monopoly mcgrath puts india out of their misery blair witch bandwagon rol
ls on pele turns up heat on ferguson party divided over kohl slush fund scandal manchester united england women in record s

```

IMAGE 4.5: HEADLINE CORPUS

Now will move towards the feature extraction in next below section.

### 4.2.3 Feature Extraction

As the study is about the sentiment analysis on customer review and News headlines, so the “Text” column of dataset customer review and headlines corpus of the News headlines dataset contains the natural language like text, sentences, emotions, words, paragraphs and the machine learning or deep learning model will not understand those text or emotions or the sentences which are in natural language and to make the model understand this data this needs to be convert into the vectors that is nothing but the numerical format which is also called as Word Embeddings or Word vectorization. There are several techniques to perform the word vectorization as below.

1. Bag of words or Count Vectorizer
2. TFIDF: Term Frequency-Inverse Document Frequency
3. Unigrams
4. Bigrams
5. N-grams
6. Word2Vec
7. Continuous Bag of Words
8. Skipgram

and to perform this I am using Bag of words (Count Vectorizer) with the help of Count Vectorizer and TFIDF vectorizer library of the sklearn library feature extraction.

To implement Word vectorization first step is to clean the data like removing the special characters lowering the data which has been already done in data pre-processing section.

#### 4.2.3.1 How Bag of words works or Count Vectorizer

Basically, It converts the text into the vectors so that the machine learning model understand the data.

1. Converts the all the sentences into the List of words presents in the vocabulary
2. Finds the frequency for sentences.
3. Calculate the vectors from the frequency column
4. Forms the bag of word matrix or the document matrix using vectors.

Ex: Consider, there are 2 sentences in the dataset:

Sentence 1: I am implementing the sentiment analysis for my MSc study

Sentence 2: I am learning the sentiment analysis for my study.

List of words: [I, am, implementing, the, sentiment, analysis, for, my, MSc, study, learning].

Frequency of the words for sentence 1

Word	Frequency
I	1
Am	1
Implementing	1
The	1
Sentiment	1
analysis	1
for	1
my	1
MSc	1

study	1
learning	0

TABLE 4.1: WORD AND THEIR FREQUENCY

Vector for the Sentence 1  $\rightarrow [1,1,1,1,1,1,1,1,1,0]$

Now finds the frequency of the words for sentence 2: I am learning the sentiment analysis for my study

word	Frequency
I	1
am	1
implementing	0
the	1
sentiment	1
analysis	1
for	1
my	1
MSc	0
study	1
learning	1

TABLE 4.2: SENTENCE 2WORD AND THEIR FREQUENCY

Vector for the Sentence 2  $\rightarrow [1,1,0,1,1,1,1,0,1,1]$

and with the help of this vectors, it forms the bag of word matrix or the document matrix

#### Bag of word or document matrix Representation:

Sentence	I	am	implementing	the	sentiment	analysis	For	My	MSc	Study	learning
Sentence 1	1	1	1	1	1	1	1	1	1	1	0
Sentence 2	1	1	0	1	1	1	1	1	0	1	1

TABLE 4.3: BAG OF WORD OR DOCUMENT MATRIX REPRESENTATION

In sentiment analysis the semantic meaning is most important and major thing we need to get is which word in the given sentence is having more importance like in above example learning, implementing, study is more important word and it should get more value more importance that other words but here we can see that these words having the same importance like other less important words (my, I, am, the). To derive the sentiment analysis, it is very important to get a exact value to the most important word in the sentences. and this is the disadvantage of bag of words that:

1. It gives the same importance to all the words in the sentences.
2. If we add the new sentence and if this new sentence having new words, then the length of the vector would increase
3. The vector contains the more no of zeros which may results into sparse matrix which will not work for sentiment analysis
4. And there is no information for the grammar and the words also not in the order.

And to solve these problems with the bag of words I have use TFIDF Term Frequency-Inverse Document Frequency for better result of the problem.

In this study for news headline below is the example how count vectorizer works: in below image 4.6 and image 4.7 it marks as 1 heighted by yellow color because the word **chines** the feature **zoe Williams** and is present in that particular data row so it marks as 1

```
df_count_features.head(100)
```

	000 000	000 000	000 children	000 chinese	000 civilians	000 dead	000 euros	000 fest	000 fine	000 hectares	...	your life	your life week	unemployment	youth video	yr old	yr ago	zealand government	zetas drug	zimbabwe president
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
95	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0
96	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0
97	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0
98	0	0	0	1	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0
99	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0

IMAGE 4.6: SPARSE MATRIX FOR NEWS HEADLINE DATA

```
df_count_features=pd.DataFrame(X_train,columns=countvector.get_feature_names())
```

```
df_count_features.head(100)
```

	000 000	000 000	000 children	000 dead	000 euros	000 jobs	000 miles	000 new	000 palestinians	000 people	...	young man	young men	young people	young woman	young women	your life	unemployment	youth video	yr old	zoe williams
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	1

ws x 5000 columns

IMAGE 4.7: SPARSE MATRIX FOR NEWS HEADLINE DATA

#### 4.2.3.2 How TFIDF Term Frequency Inverse Document Frequency works

It also works on converting the text into the vectors and then we can pass this vector to the machine learning model understand the data.

TFIDF basically have two techniques: TF and IDF

TF: Term frequency and IDF: Inverse document frequency

Term Frequency of TF is calculated by the formulae:

$$TF = \frac{\text{No of repetition words in sentences}}{\text{No of words in the sentence}}$$

Inverse  
the formulae:

document frequency IDF is calculated by

$$IDF = \log \left( \frac{\text{No of sentences}}{\text{No of sentence containing words}} \right)$$

And then

$$TFIDF = TF * IDF$$

Let's understand this with example:

Consider the movie review analysis with 2 or 3 sentences.

Review 1: The movie is very good and scary and long

Review 2: the movie is very slow and not that scary

Review 3: the movie is long and horror.

For Review 1 the TF is:

List: The, movie, is, very, good, and, scary, slow, not, that, long, horror

For Review 1:

Total no of words in review 1: 9

For word "The" the TF is: 1/9

Same for the rest of the word TF is as below:

- Tf(movie) =1/9
- Tf(is) =1/9
- Tf(very) =1/9
- Tf(good) =1/9
- Tf(and) =2/9
- Tf(scary) =1/9
- Tf(long) =1/9

Same way I have calculate the TF for all the reviews as shown in below table 4.4

Words/Term/Feature	Review1	Review2	Review3	TF(Review1) total words 9	TF(Review2) total words 9	TF(Review3) total words 6
The	1	1	1	1/9	1/9	1/6
Movie	1	1	1	1/9	1/9	1/6
Is	1	1	1	1/9	1/9	1/6
Very	1	1	0	1/9	1/9	0
Good	1	0	0	1/9	0	0
And	2	1	1	2/9	1/9	1/6
Scary	1	1	0	1/9	1/9	0
Slow	0	1	0	0	0/9	0
Not	0	1	0	0	1/9	0
That	0	1	0	0	1/9	0
Long	1	0	1	1/9	0	1/6
Horror	0	0	1	0	0	1/6

TABLE 4.4: Tf CALCULATION

Calculate the IDF for review1: no of review is 3 and no of reviews containing the word "The" Review 1 and for word "The" the IDF(The)=log (3/3) =0

Same for all other words as follows:

IDF for "movie"=log (3/3) =0

IDF for "is"=log (3/3) =0

IDF for "very"=log (3/2) =0.18

IDF for "good"=log (3/3) =0

IDF for "and"=log (3/3) =0

IDF for "scary"=log (3/2) =0

IDF for "long"=log (3/2) =0.18

Same way I have calculate the IDF for all the reviews as shown in below table 4.5:

Words/Term/Feature	Review1	Review2	Review3	IDF( total no of sentence 3)
The	1	1	1	3/3=0
Movie	1	1	1	3/3=0
Is	1	1	1	3/3=0
Very	1	1	0	3/2=0.18
Good	1	0	0	3/1=0.48
And	2	1	1	3/4=0.75
Scary	1	1	0	3/2=0.18
Slow	0	1	0	3/1=0.48
Not	0	1	0	3/1=0.48
That	0	1	0	3/1=0.48
Long	1	0	1	3/2=0.18
horror	0	0	1	3/1=0.48

TABLE 4.5: IDF CALCULATION FOR REVIEWS

Now it has been seen from the table the words like “the”, “is” value is come to zero and the words like “good”, “scary”, “slow” get the more importance with higher values.

Now lets calculated the TF-IDF vector for the word presents in Review1:

TF-IDF(‘the’, Review 1) = TF(‘the, Review 1) \* IDF(‘the’) = 1/9 \* 0 = 0

Same way for other words in the review 1 is as follows:

- TF-IDF(‘movie’, Review 1) = ) = TF(‘movie’, Review 1) \* IDF(‘movie’) = 1/9 \* 0 = 0
- TF-IDF(‘is, Review 1) = ) = TF(‘is, Review 1) \* IDF(‘is’) = 1/9 \* 0 = 0
- TF-IDF(‘very, Review 1) = ) = TF(‘very, Review 1) \* IDF(‘very’) = 1/9 \* 0.18 = 0.02
- TF-IDF(‘good, Review 1) = ) = TF(‘good, Review 1) \* IDF(‘good’) = 1/9 \* 0.48 = 0.05
- TF-IDF(‘and, Review 1) = ) = TF(‘and, Review 1) \* IDF(‘and’) = 2/9 \* 0.75 = 0.16
- TF-IDF(‘scary, Review 1) = ) = TF(‘scary, Review 1) \* IDF(‘scary’) = 1/9 \* 0.18 = 0.02
- TF-IDF(‘long', Review 1) = ) = TF(‘long', Review 1) \* IDF(‘long') = 1/9 \* 0.18 = 0.02

Similar, we can calculate for TFIDF review 2 and review 3 the TFIDF in table 4.6



Words	TF(Review1)	TF(Review2)	TF(Review3)	IDF	TFIDF(Review1)	TFIDF(Review2)	TFIDF(Review3)
The	1/9	1/9	1/6	0	0	0	0
movie	1/9	1/9	1/6	0	0	0	0
is	1/9	1/9	1/6	0	0	0	0
very	1/9	1/9	0	0.18	0.02	0.02	0
good	1/9	0	0	0.48	0.05	0	0
and	2/9	1/9	1/6	0.75	0.16	0.16	0.12
scary	1/9	1/9	0	0.18	0.02	0.02	0
slow	0	0/9	0	0.48	0	0.05	0
not	0	1/9	0	0.48	0	0.05	0
that	0	1/9	0	0.48	0	0.05	0
long	1/9	0	1/6	0.18	0.02	0	0.03
horror	0	0	1/6	0.48	0	0	0.08

TABLE 4.6: Tf-IDF CALCULATION

**For customer review dataset:**

In this study I have use both bag of words and TFIDF with the help of CountVectorizer () function of Count Vectorizer (Bag of words) and TfidfVectorizer () function of TFIDF and then I find out the tokens and the total count like how many time the token is present in the whole text. Tokens are nothing but the pieces (more like a phrase) of a sentence. It usually is a combination of 2 to 4 words. We can also consider a token as 1 word, which also called as 1-grams. For the sentiment analysis, we consider grams. A gram is just like pieces of words like a 1 gram consists of 1 word, 2 gram consists of 2 words.

(1, 1) unigrams has only one word. (1, 2) sequence of 1 or 2 words which means unigrams and bigrams. (2, 2) sequence of 2 words only which we can say as only bigrams. I have used the count-vectorizer method to divide the dataset into three segments of (1, 1), (1, 2) and (2, 2) gram, So the whole dataset has been divided into three sections of traintdataset 1, 2 and 3.

The traintdataset 1 contains all the 1-gram corpus. Similarly, the traintdataset 2 and 3 contains the 1or 2 and 2-gram corpus respectively. I have then printed the total count of the n-gram corpus in each of the sets.

Below is the result of both the bag of words and TFIDF when stop words discussed in 4.1 section of this study are presents and find that there is not that much difference on the result in the n-grams found as shown in below image of bar diagram image no

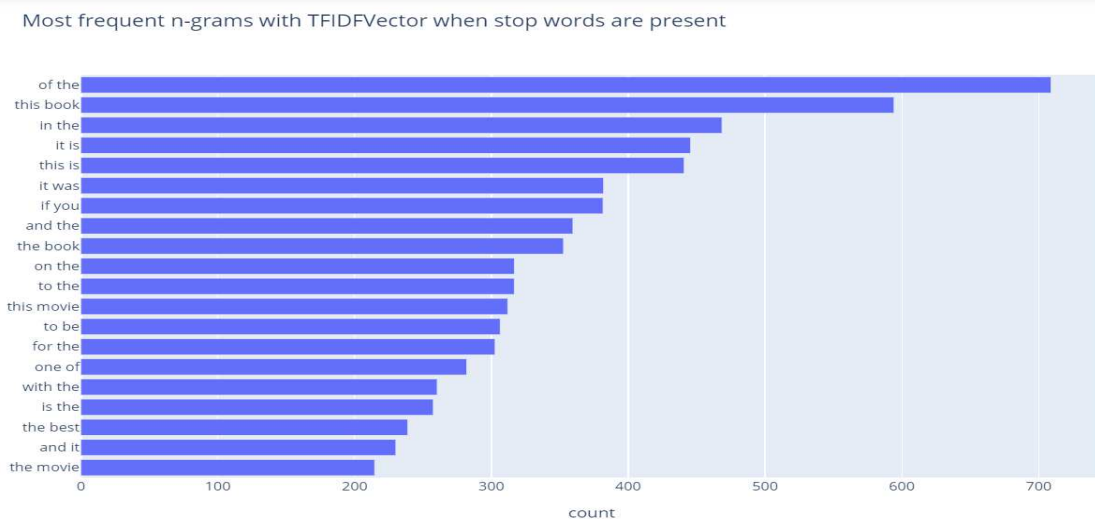


Image 4.8: N-GRAMS WITH TFIDF VECTORIZER WHEN STOP WORDS ARE PRESENT

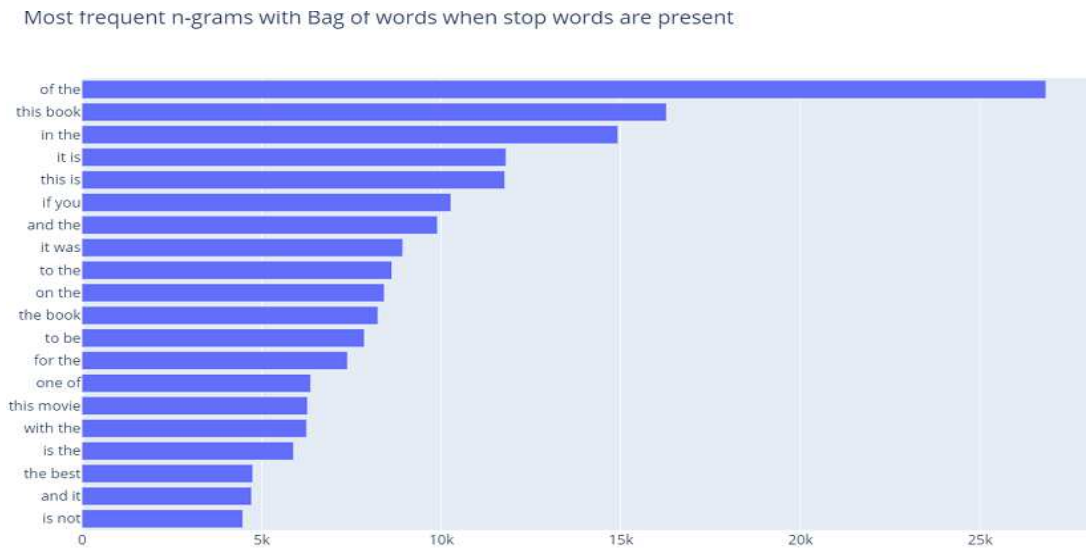


Image 4.8: N-GRAMS WITH COUNT VECTORIZER WHEN STOP WORDS ARE PRESENT

Here, I have noticed that in both the cases Bag Of Words and TFIDF vector the 2-grams like "of the", "the same", "to be", "to the" is frequent and this may cause the issue in results. So, need to remove the stop words.

#### 4.2.3.3 Removal of Stop words:

Here I have created a 'stopwords' variable, I stored the set of English stopwords from the nltk stopwords package. And then I took a separated copy of the dataset to operate on with the no stop words and copied the Text and the label columns from the original dataset to the new dataset named "nostpwords\_df". By using the lambda function, I filtered the dataset for all the stop words and removed them from the dataset.

And then the new cleaned prepared dataset has been ready for further use.

```
import nltk
from nltk.corpus import stopwords
stopwords = stopwords.words('english')
nltk.download("stopwords")
nostpwords_df["Text"] = nostpwords_df["Text"]
.apply(lambda x: ' '.join([word for word in x.split()
if word not in (stopwords)]))
]: nostpwords_df["Text"] = nostpwords_df["Text"].apply(lambda x: ' '.join([word for word in x.split() if word not in (stopwords)]))
```

After removing stop words the result is getting valuable with the help of which we can get the model with good accuracy. Same, I have applied on the News Headlines dataset shown below:

```
]: df1["NewsHeadlines"] = df1["NewsHeadlines"].apply(lambda x: ' '.join([word for word in x.split() if word not in (stopwords)]))
```

For Customer reviews below are the image 4.10 for most frequent 2-grams with count vectorizer after removing stopwords

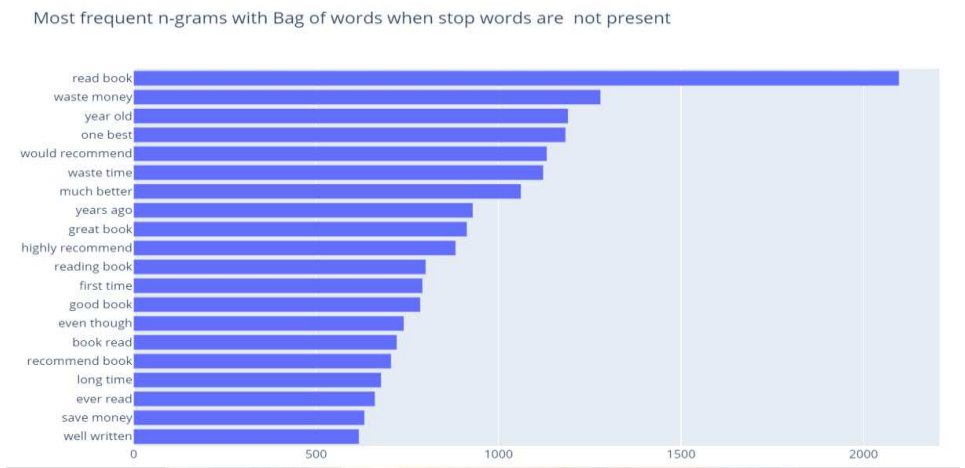


Image 4.10: N-GRAMS WITH COUNT VECTORIZER WHEN STOP WORDS ARE NOT PRESENT

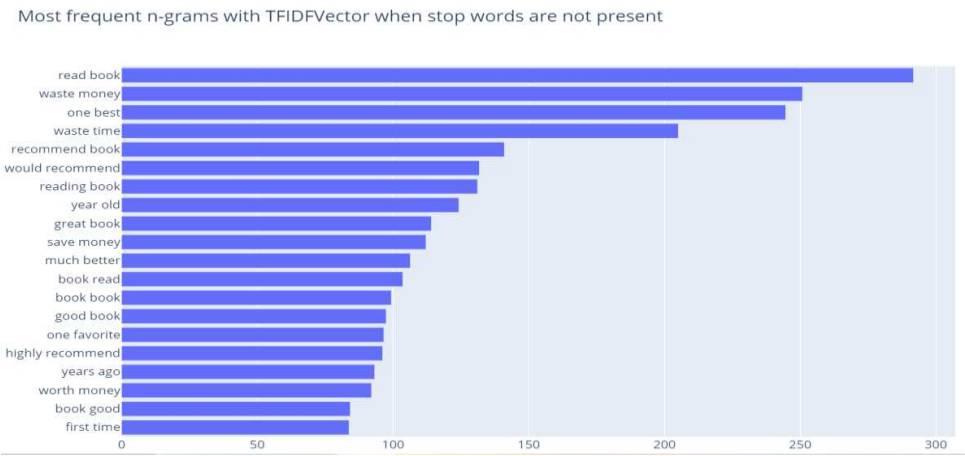


Image 4.11: N-GRAMS WITH COUNT VECTORIZER WHEN STOP WORDS ARE NOT PRESENT

As shown in the above images it has been shown that this is showing the mixing of the positive and negative reviews.as the study is about the supervise machine learning classification for sentiment analysis I have find out the most frequent positive and negative reviews of 2/3 grams when stop words are present in the dataset and when the stopwords are not there in the dataset as shown in below images

Most frequent Positive reviews 2/3-grams when stopwords present

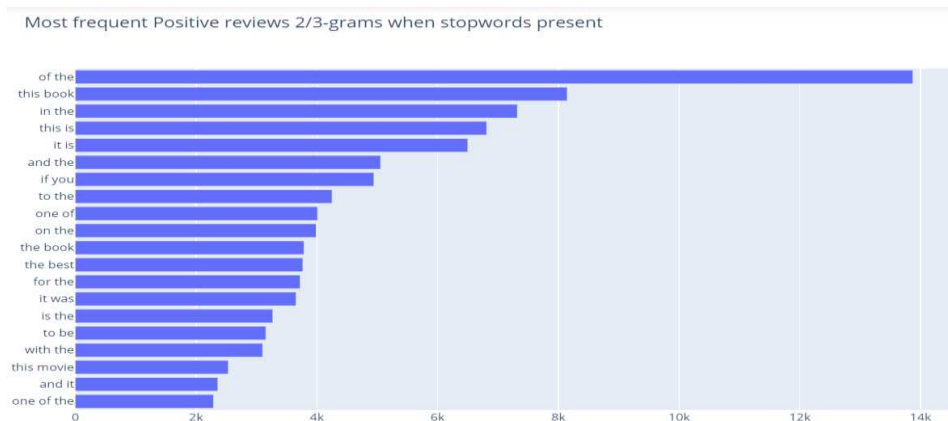


Image 4.12: POSITIVE REVIEWS WHEN STOP WORDS ARE PRESENT

#### Most frequent Negative reviews 2/3-grams when stopwords are present

Most frequent Negative reviews 2/3-grams when stopwords are present

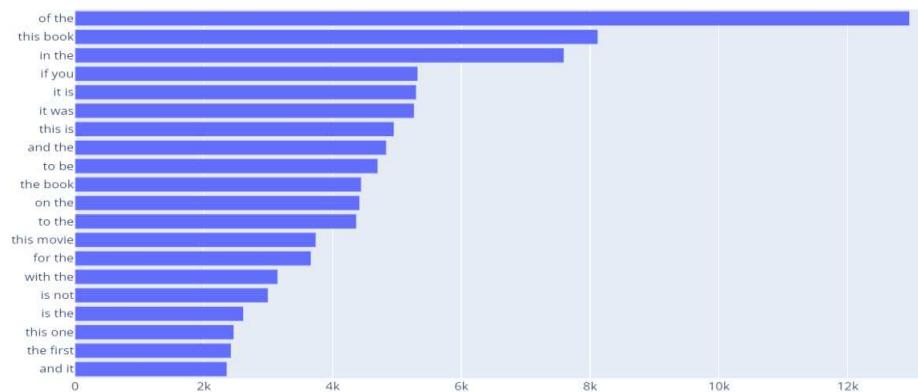


Image 4.13: NEGATIVE REVIEWS WHEN STOP WORDS ARE PRESENT

Again, from the result of negative and positive reviews with stopwords there is no difference in the positive and negative sentiment. The word “of the”, “this book”, “in the”, “it was” present for both negative and positive review and it looks the same result for both positive and negative and then because of that the model will give the wrong result so here it has been seen that how the stopwords affect the result of the sentiment analysis and why stop words should be removed from the dataset and then as I have built the separate dataset (to compare the result throughout the study I need to kept the dataset separate like dataset with stopwords and dataset without stopwords) from the original df dataset and then I remove the stopwords. after removing the stopwords here getting the exact result for positive and negative sentiment.

**Most frequent Positive reviews 2/3-grams when stopwords not present:**

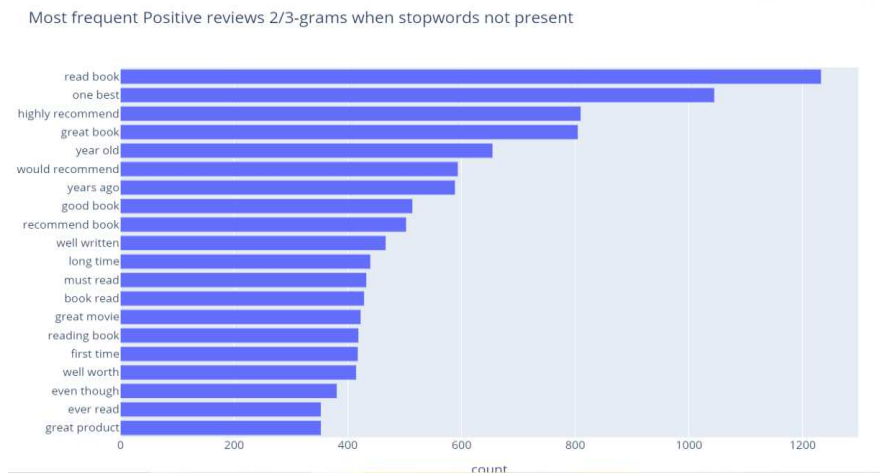


Image 4.14: POSITIVE REVIEWS WHEN STOP WORDS ARE NOT PRESENT

### Most frequent Negative reviews 2/3-grams when stopwords are not present

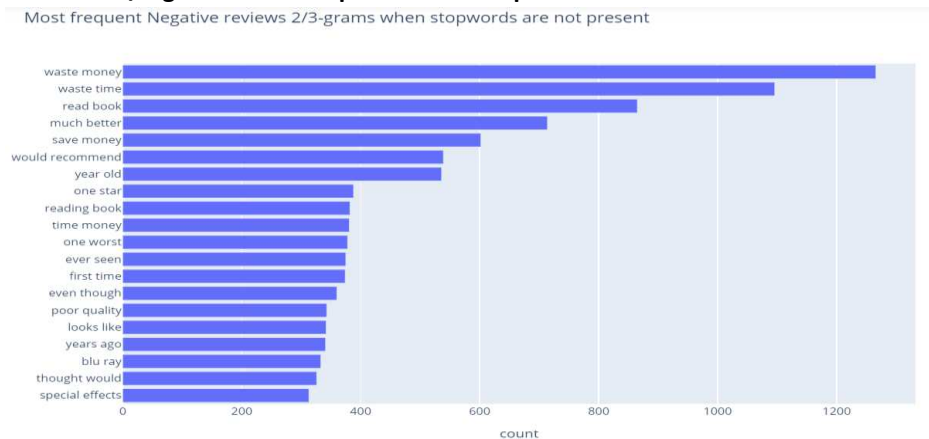


Image 4.15: NEGATIVE REVIEWS WHEN STOP WORDS ARE NOT PRESENT

As shown in above result it has been clearly seen that after removing the stopwords we are getting the actual meaningful positive and negative review differentiated with proper meaning. For example, from above images of negative review “waste money” clearly describes the sentiment as a negative sentiment and for positive review “good book” indicates the positive sentiment.

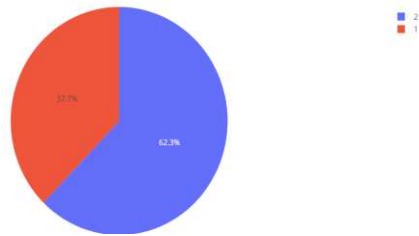
## 4.3 Data Visualization

### 4.3.1 Bar Diagram and sentiment table for Customer review and News Headlines

Customer Review:

```
fig = px.pie(train, names='label', title='Pie chart of different sentiments of reviews')
fig.show()
```

Pie chart of different sentiments of reviews



reviews	%
Positive	62.3%
Negative	37.9%

Image 4.16: PIE CHART FOR CUSTOMER REVIEW

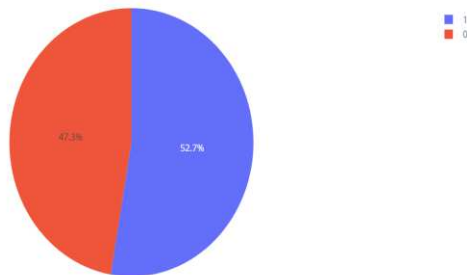
Before moving to Model development, I have plotted the bar diagram for both the dataset.

In case of Customer review for showing the percentage of the positive and negative reviews present in the original dataset.

Label 1 for: Negative reviews which is 37.7 % and Label 2 for: Positive reviews which is 62.3

#### News Headlines:

Pie chart of different sentiments of News Headlines



News Headlines	%
1-Positive	52.7.3%
0-Negative	47.3%

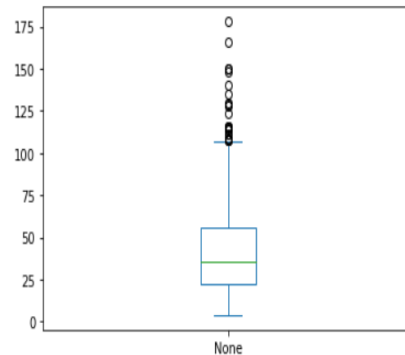
Image 4.17: PIE CHART FOR NEWS HEADLINES

### 4.3.2 Box plot Length of the Customer review and News Headlines

Here I have plotted the length of the review the longest length of the review is about 175 as shown in below figure:

```
In [18]: # Calculate review text lengths
reviewLen = pd.Series([len(review.split()) for review in df['Text']])
# The distribution of review text lengths
reviewLen.plot(kind='box')
```

Out[18]: <AxesSubplot:>



## Customer Review

Image 4.18: BOX PLOT FOR MAX LENGTH OF CUSTOMER REVIEW

## News Headlines:

```
[24]: # Calculate news headlines text lengths
newsLen = pd.Series([len(news.split()) for news in df1["NewsHeadlines"]])
# The distribution of news headlines text lengths
newsLen.plot(kind='box')
```

: [24]: <AxesSubplot:>

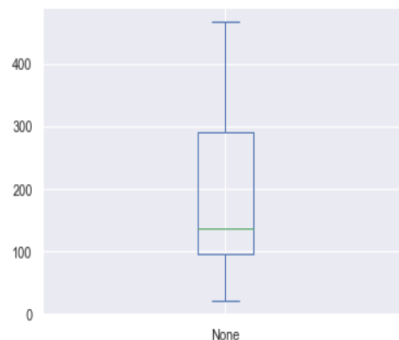


Image 4.19: BOX PLOT FOR MAX LENGTH OF NEWS HEADLINE

### 4.3.3 Word Cloud

Word cloud is one of the techniques of data visualization which represents the text and the word size indicates the frequency and the importance of the particular word.

for example, in the below both the positive and negative reviews images the size of the word "book" indicates that the frequency is highest and as shown in the below images for negative review the word "waste" is having a more frequency and for positive reviews the word "good" is having greater frequency.



**Image 4.21: WORD CLOUD FOR CUSTOMER POSITIVE REVIEWS**

**Word cloud for News Headlines:**



## 4.4 Model Development

In model development to measure the different parameter I have again develop the model with different techniques of word vectorization BagOfWords and TFIDF vectorizer as mention in the section 4.2.3 for customer review dataset with stop words and without stopwords and for news headlines the dataset has been used without stopwords only. I have implemented below algorithms for Customer review data

- 39 | Page



## 6. Random Forest Classifier

And for news headlines below are the list of the algorithm implemented in this study:

- 1 Random Forest Classifier
- 2 Naïve Bayes
- 3 Logistic regression
- 4 Decision Tree

### 4.4.1 Naïve Bayes for Customer review

As discussed in [section 2.4.1](#) Naïve bayes works on the supervised machine learning classification problems so I have implemented the same for different types of vectorizer discuss in [section 4.2.3 Feature Extraction](#) and below the list of Naïve Bayes Implementation with different vectorizer and manipulation of dataset in case of stopwords.

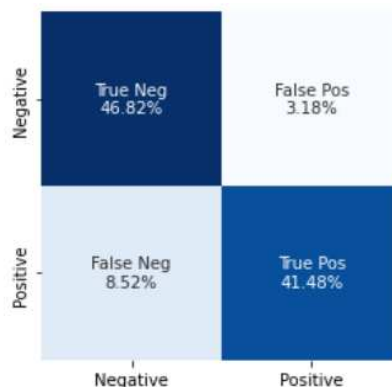
#### 4.4.1.1 Naïve Bayes when stopwords present with BagOfwords

In this case, I have created the pipeline with the countvectorizer (text vectorizer technique) and Naïve bayes model on the training dataset and then train the model with the training dataset by using the countvectorizer and Naïve bayes algorithm of classification. After the model has been trained, I passed on the testing data to the model to get the predictions and store the output in the 'preds' variable. And then I have evaluated the model on the different metrics and the confusion matrix shown below:

**Accuracy of the model is: 88.30%**

	Precision	recall	f1-score
Negative 1	0.84	0.94	0.89
Positive 2	0.93	0.82	0.88

**confusion matrix:**

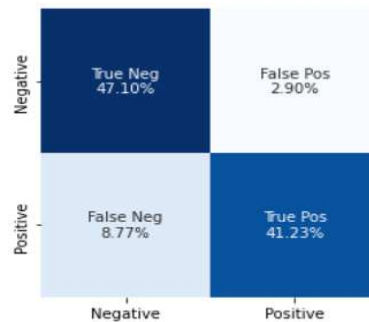


#### 4.4.1.2 Naïve Bayes when stopwords present with TFIDF

In this case I have trained the model with the TFIDF vectorizer and MultinomialNB algorithm. The same procedure has been followed as before but this time I have use the TFIDF vectorizer discuss in section no 4.3.1and create the pipeline for MultinomialNB algorithm using the make\_pipeline function. I have taken the n-grams range from 1 to 3. Now the use the same training dataset containing the stop words to train the MultinomialNB model. Once the model has been train, I have passed on the test dataset for the output values to be predicted and stored in the 'preds' variable. Now, in this model too the accuracy of 88.33% has been achieved which is minutely better than using MultinomialNB using countvectorizer but there is no difference in precision, recall and f-1 square where as there is also a slight difference in the confusion matrix as shown in below image:

And then again below are the different metrics and the confusion matrix shown below: **Accuracy of the model is: 88.33%**

	precision	recall	f1-score
Negative 1	0.84	0.94	0.89
Positive 2	0.93	0.82	0.88



As per above result, I have notice that there is a very slightly difference or very minute difference in the above two models when the stop words are present and then I have decided to use the dataset with no stopwords "nostpwords\_df" (discussed in section 4.1.1) and the apply the same model structure

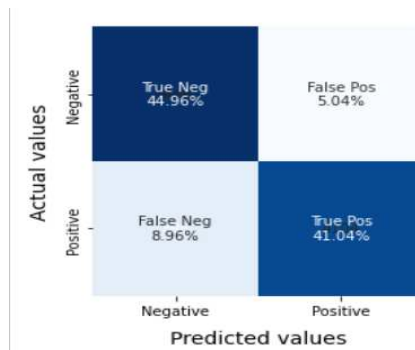
#### 4.4.1.3 Naïve Bayes when stopwords not present with BagOfwords

As discussed above using nostpwords\_df dataset I have the prepared a new model. The new database has been stored under the 'nostpwords\_df' variable. Splitting the data into training and testing set to train the model and testing it out once the model is ready. In the first step, I am using the Count Vectorizer and the Naïve Bayes algorithm to train the model. Now, by creating the pipeline of the Count Vectorizer and the Multinomial Naïve Bayes algorithm, I trained the model with the training dataset. After the completion of the model training, I passed on the testing data to the model and stored the predictions in the 'preds' variable. By using the vocabulary function, I got the total count of the features as 3091554. And then calculate the accuracy by using the accuracy\_score(), function, here the **accuracy of the model is '86.00%'**. And then again below are the different metrices and the confusion matrix shown below:

precision recall f1-score

Negative 1	0.83	0.90	0.87
Positive 2	0.89	0.82	0.85

Confusion Matrix:



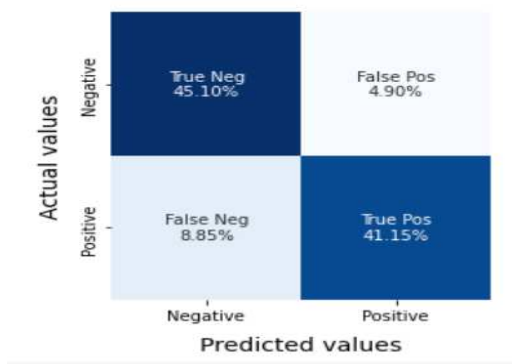
In comparison to the result of **when stopwords present with BagOfwords** accuracy is less, where as the accuracy should be greater as the dataset is cleaner than the dataset used in case **when stopwords present with BagOfwords** model creation will discuss the reason in the **section 4.4.1.4 when stopwords not present with TFIDF**.


#### 4.4.1.4 Naïve Bayes when stopwords not present with TFIDF

Previously I have used the Count Vectorizer and the Naïve Bayes method to train the model. Now, lets prepare another model by using the TFIDF vectorizer and naïve bayes method. After preparing a pipeline with the TFIDF vectorizer and the multinomial naïve bayes algorithm, I trained the model using the training dataset through the prepared pipeline. I then passed on the testing data into the model after it has been trained completely to get the predictions and stored them in the 'preds' variable. There are 3091554 features in the model and this model has achieved and **accuracy score of 86.25 %**.

	precision	recall	f1-score
Negative 1	0.84	0.90	0.87
Positive 2	0.89	0.82	0.86

Confusion matrix:



Now here, I have notice that the accuracy in both the cases Naïve Bayes with Countvectorizer and Naïve Bayes with TFIDF after removal of stop words is less than when the stopwords present and the may be the reason for this is stop word also contains the words "no","not","nor" as shown below  and as we know this words plays an most important role in the sentiment analysis.

```
stopwords
['any',
'both',
'each',
'few',
'more',
'most',
'other',
'some',
'such',
'no',
'not',
'only',
'own',
'same',
'so',
'than',
'too',
'very',
...]
```

Let's evaluate how model will work after removing the word "no","not","nor" from stopwords, Here I am going to remove these words form the stopwords and then again train the model and evaluate the metric and then the below is the for the accuracy and the other matrices has been found in this 2 cases

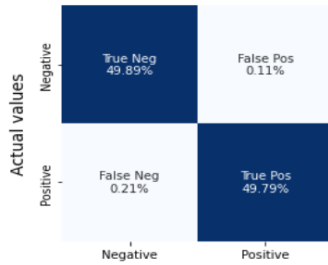
**1 Result: Naïve Bayes when stopwords not present with BagOfwords: Accuracy: 99.68% AUC: 99.68%**

Model after removing stopwords except no,nor,not: CountVectorizer + MultinomialNB  
 Number of features: 847866  
 Accuracy: 0.9968

AUC :Area Under Curve: 0.9968

	precision	recall	f1-score	support
1	1.00	1.00	1.00	15000
2	1.00	1.00	1.00	15000
accuracy			1.00	30000
macro avg	1.00	1.00	1.00	30000
weighted avg	1.00	1.00	1.00	30000

Confusion Matrix : CountVectorizer + MultinomialNB



## 2 Result: Naïve Bayes when stopwords not present with TFIDF: Accuracy: 94.98%

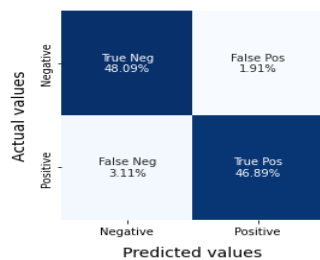
**AUC :94.98%**

Model after removing stopwords except no,nor,not: TfidfVectorizer + MultinomialNB  
 Number of features: 847866  
 Accuracy: 0.9498

AUC :Area Under Curve: 0.9498

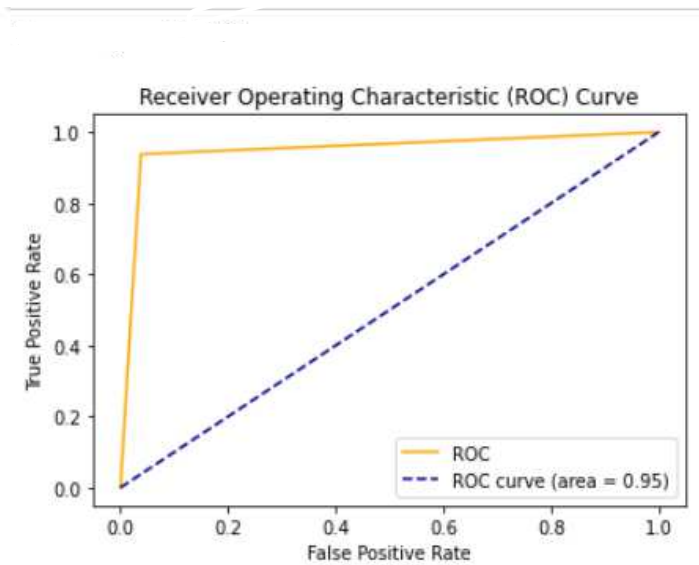
	precision	recall	f1-score	support
1	0.94	0.96	0.95	15000
2	0.96	0.94	0.95	15000
accuracy			0.95	30000
macro avg	0.95	0.95	0.95	30000
weighted avg	0.95	0.95	0.95	30000

Confusion Matrix : TfidfVectorizer + MultinomialNB



I have then plotted the ROC Receiver Operating Characteristic Curve for the Model: Naïve Bayes when stopwords not present with TFIDF: **Accuracy: 94.98%** on the new cleaned dataset as shown below.

**AUC :94.98%**



And then here after removing the word “no”, “not”, “nor” from stopwords we are getting very good accuracy and hence it has been proved that how the words “no”, “not”, “nor” plays an most important role in the sentiment analysis classification problems. Further in the study, I have implemented the below algorithms Logistic Regression, SVM, KNN, Random Forest Classifier, Decision tree on the dataset which is very well prepared like without the stopwords except word “no”, “not”, “nor”

#### 4.4.2 Logistic regression for Customer review

As discussed in section 3.4.2 Logistic regression: is a classification algorithm for supervised Machine learning. I have implemented the Logistic regression. The objective of the logistic regression is to find out the class for the given input in other words it finds the probability to identify the given input is belongs to or close to which class. In case of this study, it is finding the probability to identify the “Text” belongs to class “2” which is nothing but positive or class “1” which is nothing but Negative.

Here for Logistic regression, I have use the TFIDF vectorizer and trained the model using “SAG” solver. “SAG” solver is classification of the linear type that works for logistic regression as well as support vector machines of the linear type. The solver implements the Coordinate Descent algorithm that used to solve the optimization issues with successful approximate minimization along coordinate hyperplanes.

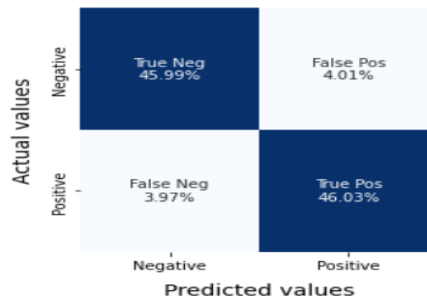
The result I have got with Logistic regression for the metrices shown below:

**Accuracy: 92.02%**

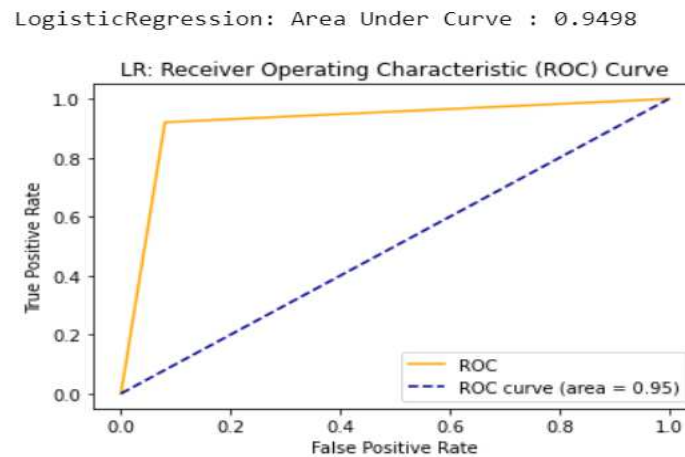
**AUC: Area Under Curve: 94.98%**

	precision	recall	f1-score
Negative 1	0.92	0.92	0.92
Positive 2	0.92	0.92	0.92

**Confusion Matrix:**



So as per the result the accuracy is less than the Naïve Bayes and then I have again same as calculated the ROC and plot the ROC curve for Logistic regression as below:



After Logistic regression I have then implemented the SVM algorithm as below.

#### 4.4.3 SVM-Support vector Machine for Customer Reviews

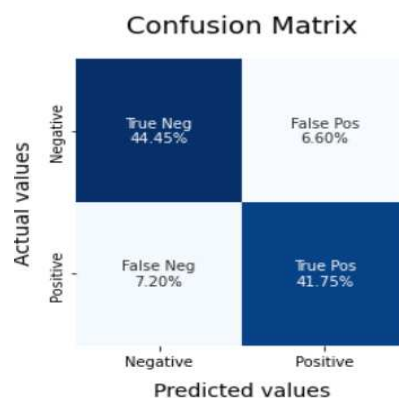
The same procedure I have used to implement the SVM model but there is difference in the dataset size as discuss earlier Support vector Machine is slow So I have taken sample of 10000

rows. And then train the model by following the same procedure as in above models like created the pipeline with TFIDF vectorizer and evaluate the metrics like Accuracy, precision, recall, f1-score and confusion matrix as shown below:

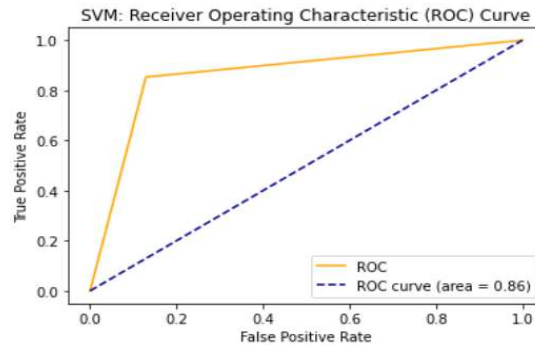
**Accuracy: 86.20% and AUC: Area Under Curve: 86.18%**

	precision	recall	f1-score
Negative 1	0.86	0.87	0.87
Positive 2	0.86	0.85	0.86

Confusion Matrix:



ROC Curve for SVM:



Next Model I have prepared is using Decision tree algorithm and TFIDF vectorizer as explained below

#### 4.4.4 Decision Tree for Customer Reviews

The same procedure I have followed to implement the Decision tree use the TFIDF vectorizer and created the pipeline and train the model calculated the confusion matrix, evaluate the metrics like Accuracy, precision, recall, f1-score and confusion matrix as shown below:

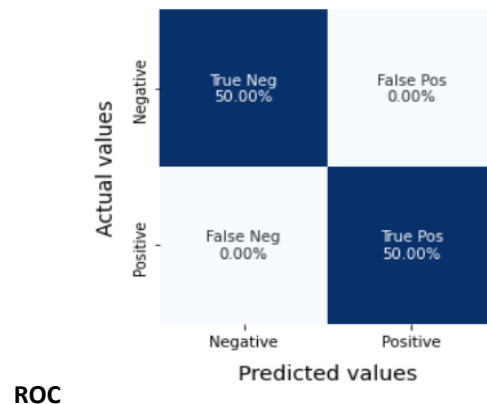
and here I have got the very good accuracy of 100% with AUC of 100%

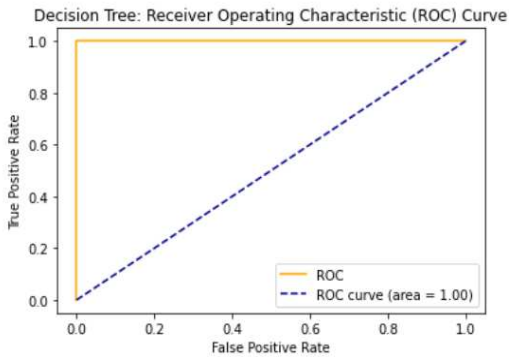
**Accuracy: 100%**

**AUC:Area Under Curve: 100%**

	precision	recall	f1-score
Negative 1	1.00	1.00	1.00
positive 2	1.00	1.00	1.00

Confusion Matrix:





The decision tree AUC result shows that this is best fitted model for this problem statement.

#### 4.4.5 KNN for Customer review

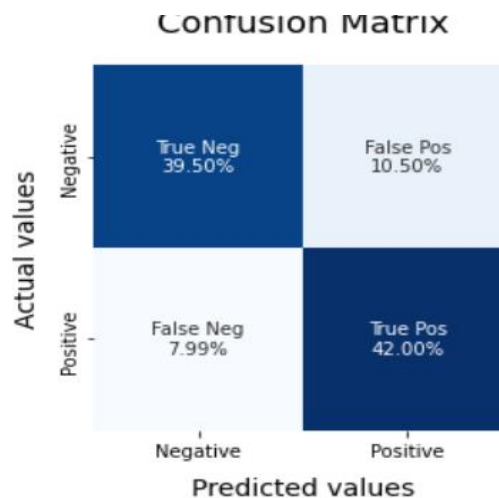
As discuss in the section 2.4.5 of this study KNN works with the help of K value and the distance between datapoints. Here I have followed the same process as that of other algorithms to implement the KNN. I have taken the k-value as 5 and the Euclidean distance has been used to calculate the distance between the points.

**Accuracy: 81.11%**

**KNN: AUC: Area Under Curve: 81.50%**

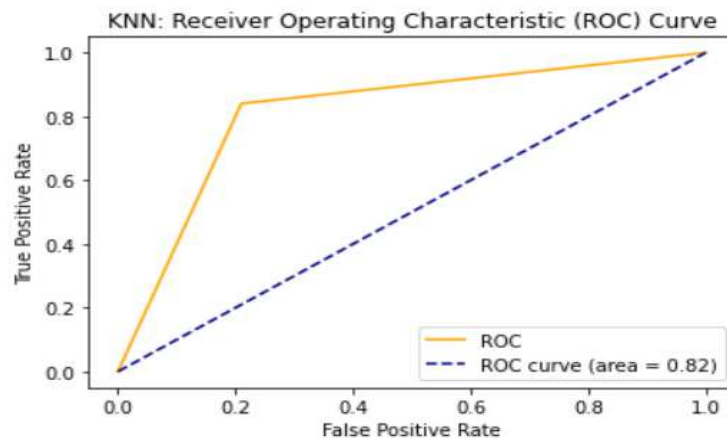
	precision	recall	f1-score
Negative 1	0.83	0.79	0.81
Positive 2	0.80	0.84	0.82

Confusion Matrix:



ROC Curve:





#### 4.4.6 Random Forest Classifier for Customer review

The Random Forest classifier uses the decision tree with the help of row sampling and feature samples (column).

It divides the data into no of samples and use the decision tree on each sample and then aggregate the output values. In this study dataset I have use the entropy to measure the purity of the split. And n-estimator=200 is the count of the trees that want to split the dataset if the higher no of estimators has been given it gives a very good performance but its slows down the execution of the algorithm.

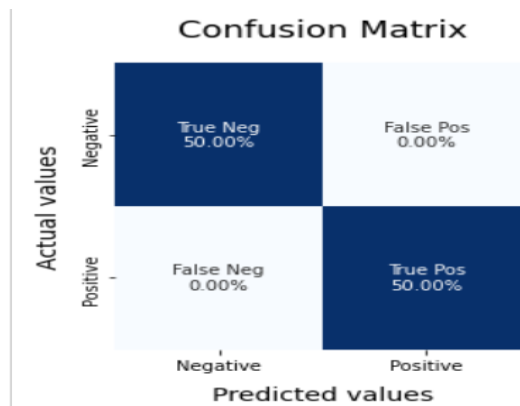
**Accuracy: 100%**

**RFC: AUC: Area Under Curve: 100%**

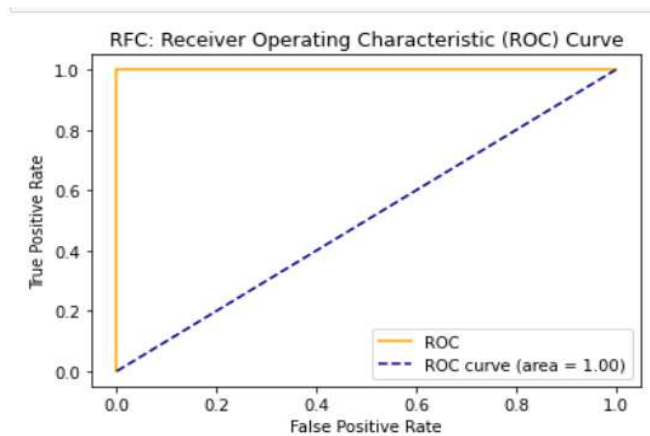
	Precision	recall	f1-score
Negative 1	1.00	1.00	1.00
Positive2	1.00	1.00	1.00

And here it has been seen that the accuracy and the AUC is very good which is 100%.

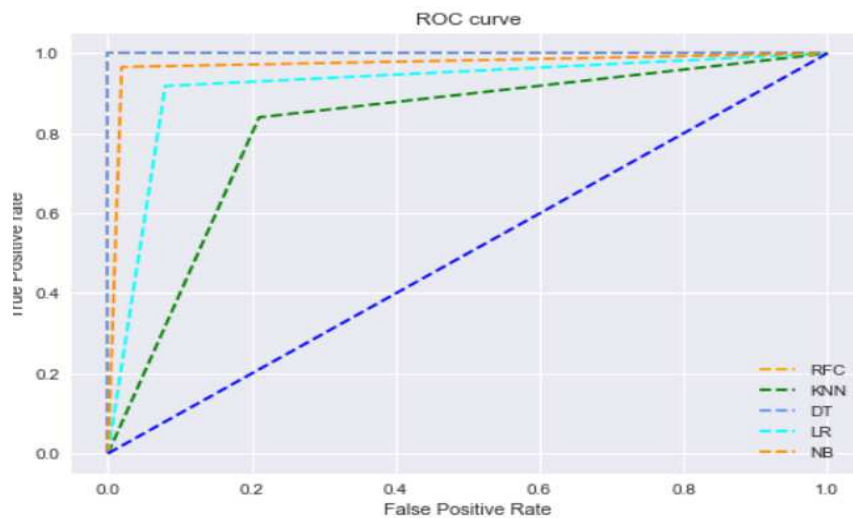
**Confusion Matrix:**



ROC:



I have plotted the ROC for the five models as shown below where it has been seen the five different colour for five models as shown below:



Here in above image of the ROC there five roc curves for random forest classifier,

KNN, Decision tree, logistic regression, Naïve bayes and the image is showing only four because the model decision tree and Random Forest classifier is giving the same accuracy of 100 % as we know random forest classifier uses the decision tree internally so the Decision tree curve coverup the random forest classifier curve as I have calculated the Random forest first and Decision tree at nest step.

```
# roc curve for models
fpr6, tpr6, thresh6 = roc_curve(y_test_n, preds_n, pos_label=2)
#fpr5, tpr5, thresh5 = roc_curve(y_test_n, svm_preds, pos_label=2)
fpr1, tpr1, thresh1 = roc_curve(y_test_n, Rfc_predict, pos_label=2) #RFC
fpr2, tpr2, thresh2 = roc_curve(y_test_n, knn_predict, pos_label=2) # knn
# roc curve for models
fpr3, tpr3, thresh3 = roc_curve(y_test_n, Dt_predict, pos_label=2) #DT
fpr4, tpr4, thresh4 = roc_curve(y_test_n, predictions, pos_label=2) # LR
# roc curve for tpr = fpr
random_probs = [0 for i in range(len(y_test_n))]

# "aqua", "darkorange", "cornflowerblue"
p_fpr, p_tpr, _ = roc_curve(y_test_n, random_probs, pos_label=1)
#p_fpr, p_tpr, _ = roc_curve(y_test_n, random_probs, pos_label=-1)
# plot roc curves
plt.plot(fpr1, tpr1, linestyle='--', color='orange', label='RFC')
plt.plot(fpr2, tpr2, linestyle='--', color='green', label='KNN')
plt.plot(fpr3, tpr3, linestyle='--', color='cornflowerblue', label='DT')
plt.plot(fpr4, tpr4, linestyle='--', color='aqua', label='LR')
plt.plot(fpr6, tpr6, linestyle='--', color='darkorange', label='NB')
plt.plot(p_fpr, p_tpr, linestyle='--', color='blue')
# title
plt.title('ROC curve')
# x label
plt.xlabel('False Positive Rate')
# y label
plt.ylabel('True Positive rate')

plt.legend(loc='best')
plt.savefig('ROC', dpi=300)
plt.show();
```

Code for the ROC curve.

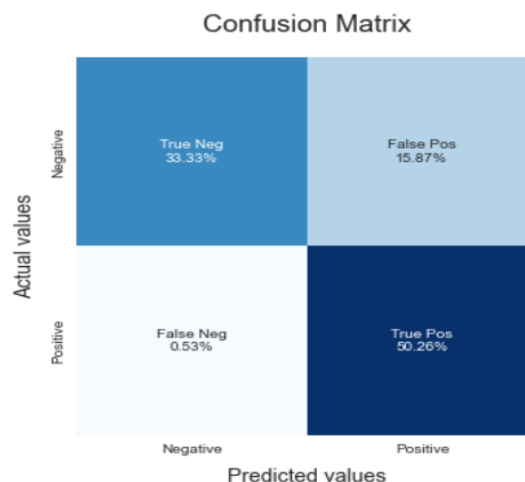
#### 4.4.7 RandomForestClassifier for News Headlines

Here in case of News headlines there I have implemented the Random forest classifier with the estimators=200 and criterion='entropy' same as that of customer review data. it is then divided the dataset into 200 decision trees and calculate the distance with the help of entropy. With this I am getting the **accuracy of 84% and AUC is 83.35%**

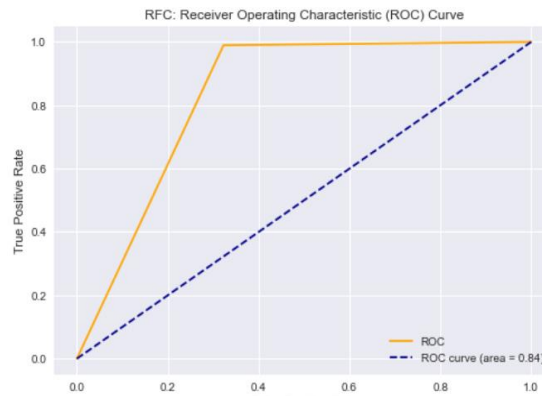
And below are the other metrices evaluation

	precision	recall	f1-score
Negative0	0.98	0.68	0.80
Positive 1	0.76	0.99	0.86

Confusion Matrix:



ROC Curve:

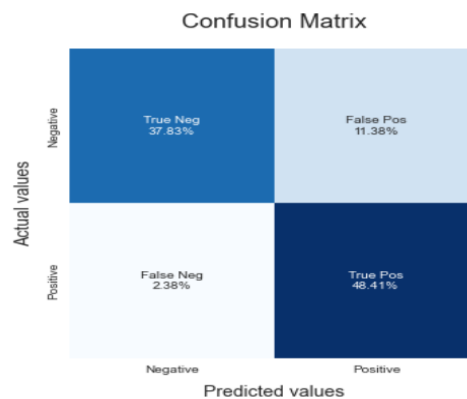


#### 4.4.8 Naïve Bayes for News Headlines

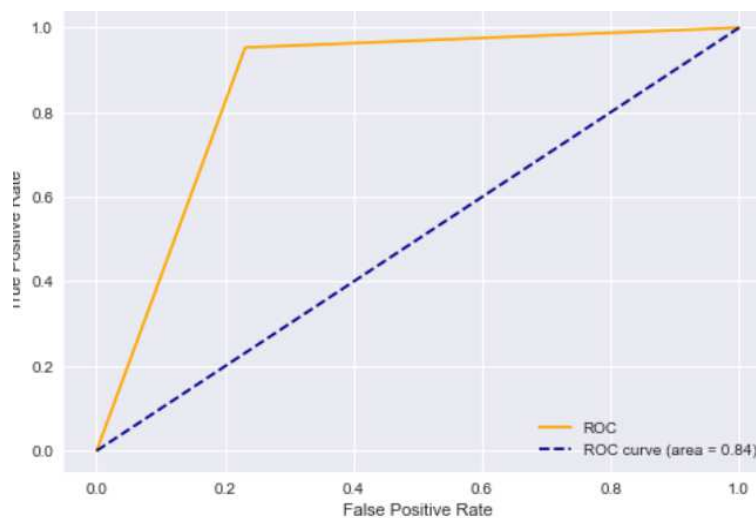
For news headlines I have implemented the naïve bayes with countvectorizer as Naive Bayes Algorithm is work on the basis of Bayesian thermos as the dataset used in this study having the 2 labels as positive and negative which nothing but the classes and the features in the dataset is the text nothing but the “headline\_corpus” column in the dataset and here I am getting a **accuracy of 86%** and the **AUC is 86%** and below are the metrics calculated:

	precision	recall	f1-score
Negative0	0.94	0.77	0.85
Positive 1	0.81	0.95	0.88

Confusion Matrix:



ROC Curve:



#### 4.4.9 Decision Tree for News Headlines

The same procedure I have followed to implement the Decision tree using the countvectorizer and train the model calculated the confusion matrix, evaluate the metrics like Accuracy, precision, recall, f1-score and confusion matrix as shown below:

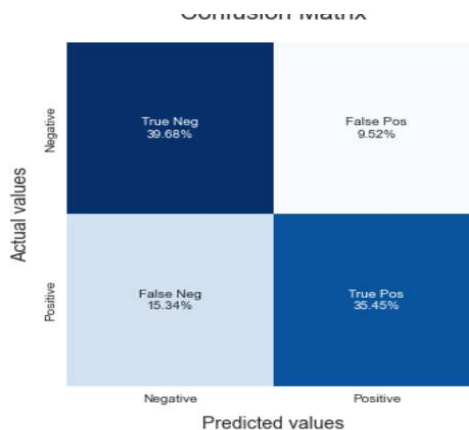
and here I have got the very good accuracy of 100% with AUC of 100%

**Accuracy: 75.13%**

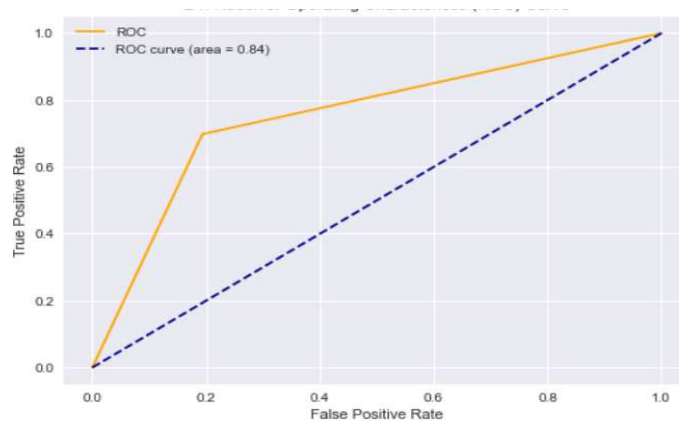
**AUC: Area Under Curve: 75.22%**

		precision	recall	f1-score
Negative	0	0.72	0.81	0.76
positive	1	0.79	0.70	0.74

Confusion Matrix:



ROC:



#### 4.4.10 Logistic regression for News Headlines

As discussed in section 3.4.2 Logistic regression: is a classification algorithm for supervised Machine learning. I have implemented the Logistic regression. The objective of the logistic regression is to find out the class for the given input in other words it finds the probability to identify the given input is belongs to or close to which class. In case of this study, it is finding the probability to identify the “NewsHeadlines” belongs to class “2” which is nothing but positive or class “0” which is nothing but Negative.

Here for Logistic regression, I have trained the model using “SAG” solver. “SAG” solver is classification of the linear type that works for logistic regression as well as support vector machines of the linear type. The solver implements the Coordinate Descent algorithm that used to solve the optimization issues with successful approximate minimization along coordinate hyperplanes.

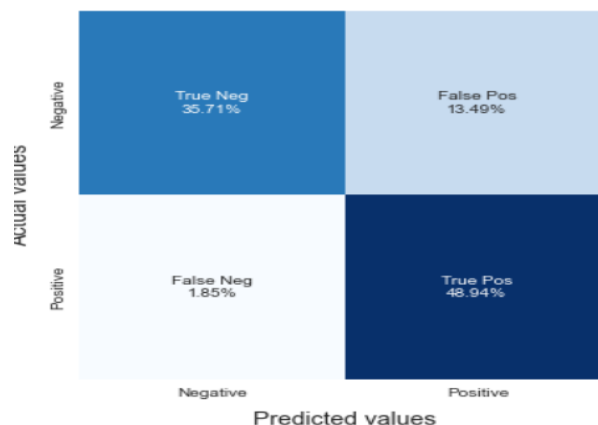
The result I have got with Logistic regression for the metrics shown below:

**Accuracy: 84.66%**

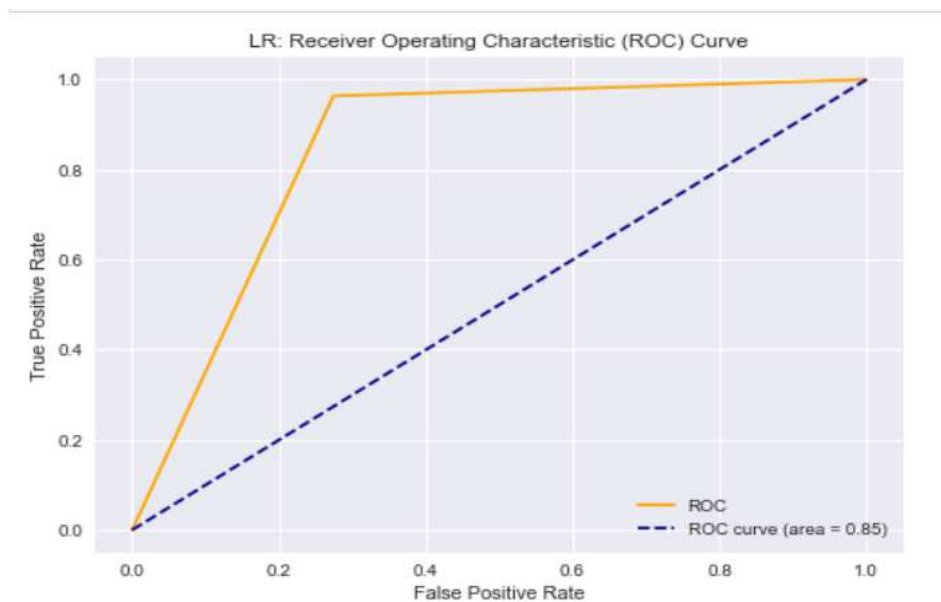
**AUC: Area Under Curve: 84.47%**

	precision	recall	f1-score
Negative 0	0.95	0.73	0.82
Positive 1	0.78	0.96	0.86

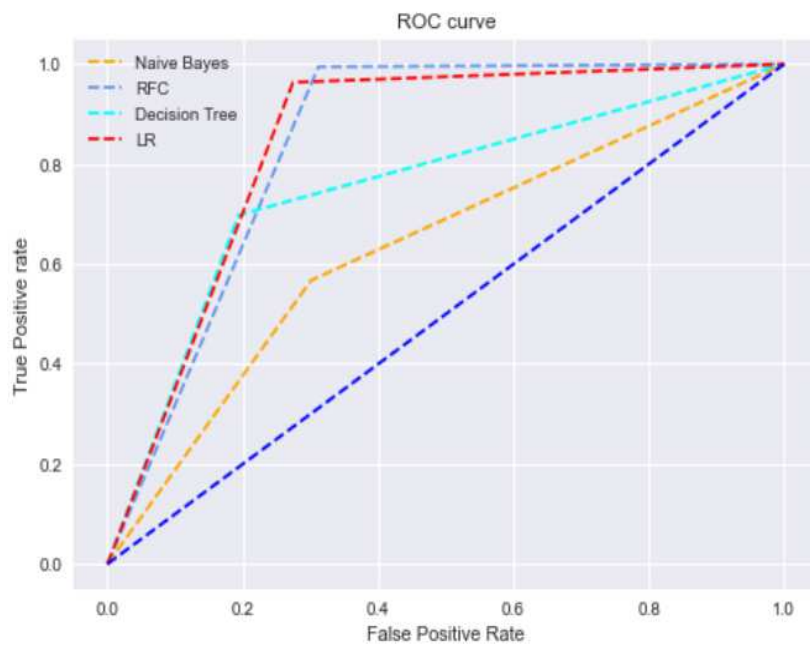
**Confusion Matrix:**



ROC : LR:



I have plotted the ROC for all this four models as shown below where it has been seen the four different colour for three models as shown below:



From the above diagram random forest classifier covers the more area and can say that it performs well for this problem statement of News Headlines Sentiment analysis.

In this way the model development has been done for both Customer review and the News Headlines. Now will move towards the result section of the study where I am going to discuss about the result and will do the comparison of the result of models used in the study for both Customer review and the News Headlines.

## 5 Result and Discussion

As sentiment analysis works with the NLP and NLP is nothing but the natural language processing. It means with this we can process the natural languages which is nothing but the online data like the comments on the social media, tweets, reviews, news and here the data is literally uncleaned, unmanage, unformatted, contains lots of special characters. Also, sentiment analysis is the way of calculating the sentiment behind this data online and it measures it in different type of class like "Yes", "No", "Positive", "Negative", "Neutral".

And to calculate this finding the meaning of the sentences and word is most important and most of the time sentences having the words like "is", "or", "an", "would", "and" and this words makes it difficult for calculating the

The actual sentiment behind the sentence or the text so the main thing for sentiment analysis is data pre-processing which includes cleaning of data like removing special characters make the text in the same case like lower case, and the removal of stopword.

The cleaned, formatted data gives the better result so it is very important to pre-process the dataset.

Once the dataset has been prepared like after cleaning, pre-processing and formatting the dataset. The feature extraction has been done. For customer review there are two csv files test.csv and train.csv the data is huge so I prefer to take subset for the 50000 rows of train dataset and 10000 rows from test dataset.

And combine this data into one dataframe which is df as shown below.

```
#data is huge so taking subset for the 50000 rows of train dataset and 10000 rows from test dataset
train['Title'].fillna('', inplace=True)
test['Title'].fillna('', inplace=True)
train_len = 50000
test_len = 10000
rs = 42 # random set of data start
df = pd.concat([train.loc[train['Label'] == 1].sample(train_len//2, random_state=rs),
               train.loc[train['Label'] == 2].sample(train_len//2, random_state=rs),
               test.loc[test['Label'] == 1].sample(test_len//2, random_state=rs),
               test.loc[test['Label'] == 2].sample(test_len//2, random_state=rs)].reset_index(drop=True))
```

As it is very common the training dataset should be greater in size as compare to the test dataset so considering this, I have taken 50000 records for train data and 10000 record for test data.

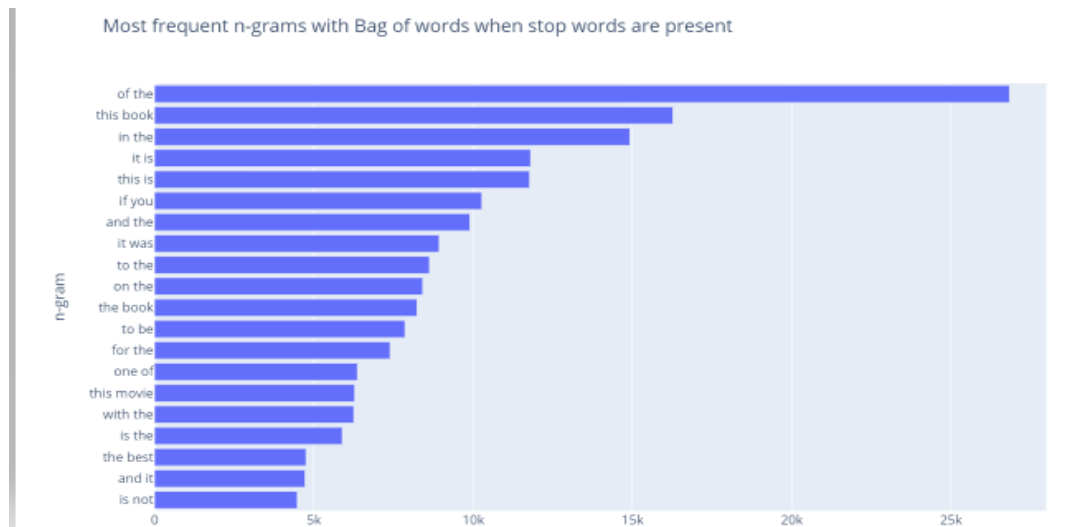
Same for the News headline model I have split the dataset as year of news for news headlines as shown below:

```
: df=pd.read_csv("StockNewsData.csv", encoding="ISO-8859-1") # added the encoding to
:
: #divided the data in to the train and the test dataset according to year of news
train=df[df['Date']<'20150101']
test=df[df['Date']>'20141231']
```



Now will discuss about the stopwords removal importance and how it makes the difference in the result

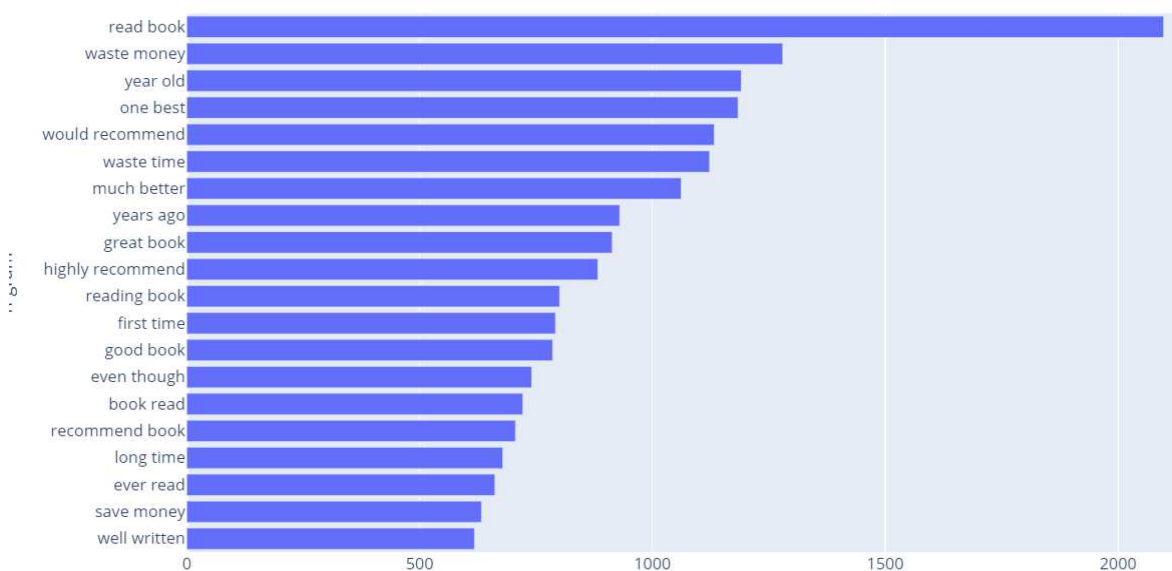
For customer review study I have found out the most frequent n-grams present in the customer review dataset with countvectorizer and TFIDF vectorizer, when the stopwords are present as shown in the below diagram.



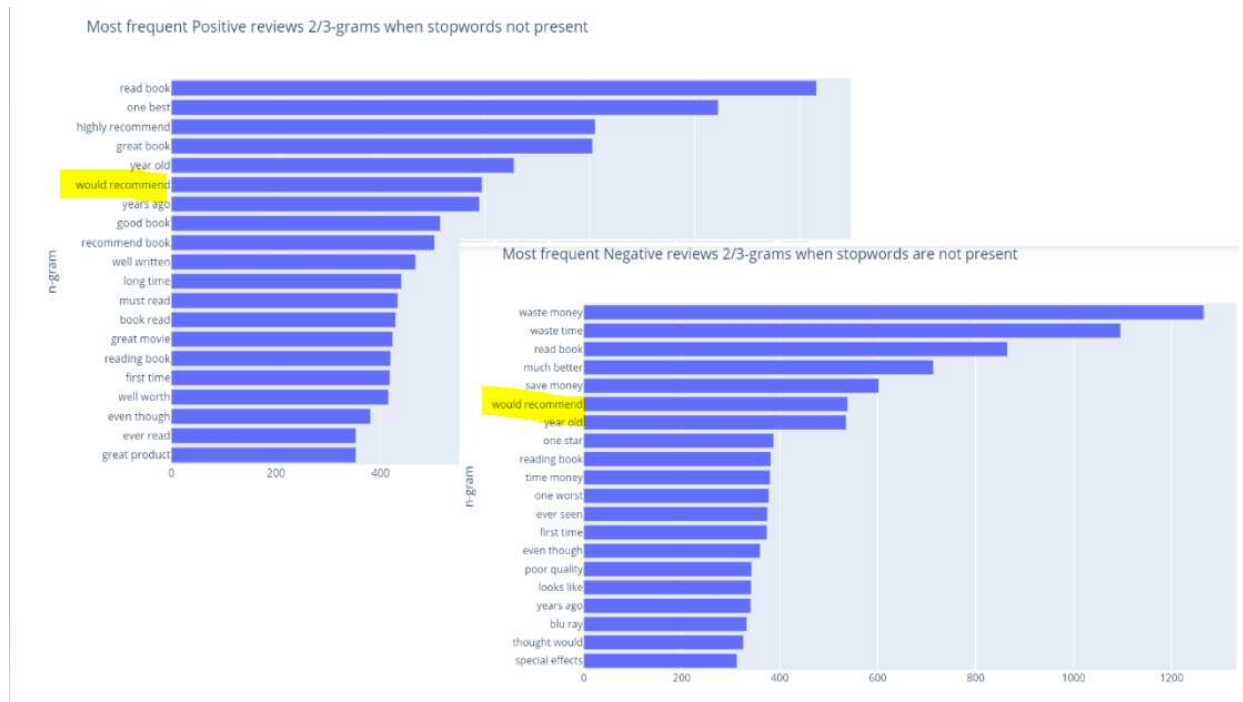
And here getting the n-grams like “of the”, “this book”, “in the”, “it is” and this word doesn’t make decision of positivity or negativity.

And then I removed the stopwords and again find out the most frequent n-grams present without stopwords below is the result for the same.

Most frequent n-grams with Bag of words when stop words are not present



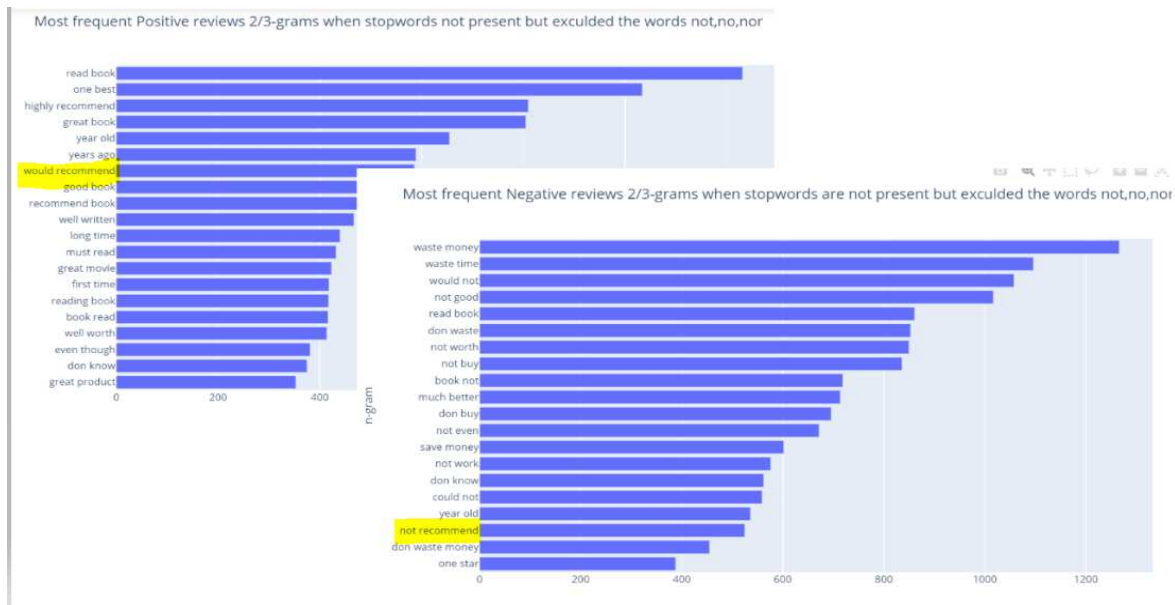
And then it has been seen that the better finding of the n-grams has been found from which it is easy to identify the negativity or the positivity and then I have decided to find out most frequent positive and negative n-grams it means most frequent positive and negative reviews result as shown below.



As per the above image the positive and negative reviews result is having the same word which is “would recommended” and its seems the same meaning and it is again difficult to identify the review is negative or positive. Then I found the stop words list also contains the words “no”, “not”, “nor” and as discussed earlier in this study this words plays an important role in sentiment analysis study so that should not get removed from the dataset.

I have then prepared the dataset which removes the stopwords but not removed the words “not”, “nor”, “no”.

And the result for the most frequent positive and negative reviews as shown below.



As per above image it has been shows that how the stopword removal with careful consideration plays an important role in the result of positive and negative reviews.

Now will move towards the discuss of the model result and will see how the model will get the different result in different cases.

For customer review: As discussed in methodology section of the study I have built the model with stopwords and then without stopwords for naïve bayes using Countvectorizer and TFIDF Vectorizer. And again as stopwords contains the word like “no”, “not”, “nor” and this words are very important for sentiment analisys as most of the time it directly indicates the negativity

Below is the table for the results showing the accuracy with and without stopwords and after excluding the words “no”, “not”, “nor” from the stopword variable.

Naïve Bayes Accuracy %	With stopword	Without stopword	Excluding No. Not. Nor from stopwords
<b>CountVectorizer</b>	88.30%	86.00%	99.68%
<b>TFIDF Vectorizer</b>	88.33%	86.25%	94.98%

As per the above table accuracy of the model is getting better when the proper manipulation of the stopword has been done and here I am getting good accuracy score for naive bayes in case of customer review sentiment analysis. and one more thing can be observed that with count vectorizer getting the better accuracy than the TFIDF vectorizer.

For news headlines I have used the Label and “NewHeadline” corpus (it is nothing but the list of the string”) I have train the Naïve bayes model on fully prepared data and below is the result of the accuracy.

I have also calculated the other metrics mentioned below.

1. Accuracy
2. Recall
3. Precision
4. F1 Score
5. ROC
6. AUC

## 5.1 Result Table

### 5.1.1 Naïve Bayes

Naïve Bayes	Accuracy	Recall		Precision		F1 Score		AUC
		Positive	Negative	Positive	Negative	Positive	Negative	
Customer Review	99.68%	1	1	1	1	1	1	99.68%
News Headlines	86.24%	0.95	0.77	0.81	0.94	0.88	0.85	86.00%

From the above table Naïve bayes accuracy for customer review is better than the accuracy for news headlines dataset.

Let's discuss the result of the other models implemented in the study.

### 5.1.2 Logistic regression

Logistic Regression	Accuracy	Recall		Precision		F1 Score		AUC
		Positive	Negative	Positive	Negative	Positive	Negative	
Customer Review	92.02%	0.92	0.92	0.92	0.92	0.92	0.92	94.98%
News Headlines	84.66%	0.96	0.73	0.78	0.95	0.86	0.82	84.47%

### 5.1.3 Decision Tree

Decision Tree	Accuracy	Recall		Precision		F1 Score		AUC
		Positive	Negative	Positive	Negative	Positive	Negative	
Customer Review	92.02%	0.92	0.92	0.92	0.92	0.92	0.92	94.98%
News Headlines	75.13%	0.70	0.81	0.79	0.72	0.74	0.76	75.22%

### 5.1.4 Random forest

Random forest classifier	Accuracy	Recall		Precision		F1 Score		AUC
		Positive	Negative	Positive	Negative	Positive	Negative	
Customer Review	100%	1	1	1	1	1	1	100%
News Headlines	84%	0.99	0.68	0.76	0.98	0.86	0.80	83.35%

### 5.1.5 SVM

SVM	Accuracy	Recall		Precision		F1 Score		AUC
		Positive	Negative	Positive	Negative	Positive	Negative	
Customer Review	86.20%	0.85	0.87	0.86	0.86	0.86	0.87	86.18%

### 5.1.6 KNN

KNN	Accuracy	Recall		Precision		F1 Score		AUC
		Positive	Negative	Positive	Negative	Positive	Negative	
Customer Review	81.11%	0.84	0.79	0.80	0.83	0.82	0.81	81.50%

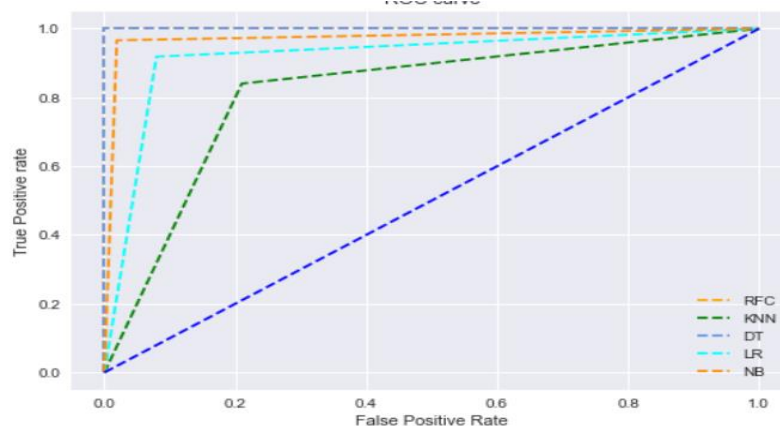
## 5.2 Accuracy and AUC Result Table

	Algorithm	Accuracy	AUC
<b>Customer Review</b>	Naïve Bayes	99.68%	99.68%
	Logistic Regression	92.02%	94.98%
	Decision Tree	100%	100%
	Random Forest classifier	100%	100%
	SVM	86.20%	86.12%
	KNN	81.11%	81.50%
<b>News Headlines</b>	Naïve Bayes	86%	86%
	Logistic Regression	84.66%	84.47%
	Decision Tree	75.13%	75.22%
	Random Forest classifier	84%	83.35%

From the accuracy and AUC table out of 6, 4 model are best fitted 1. Naïve Bayes 2. Logistic Regression 3. Decision Tree 4. Random Forest classifier and for news headlines is naïve bayes is best fitted model. Also, from the ROC curve and AUC the best fitted model has been identified as shown below:

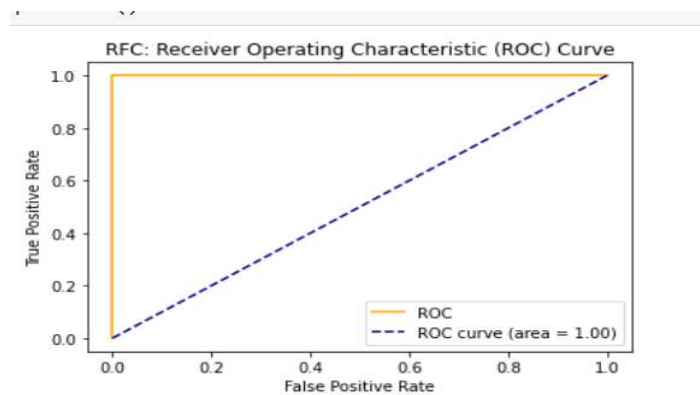
## 5.3 Customer Review ROC curve

Below is the ROC curve for Naive bayes, Logistic regression, Decision Tree and KNN

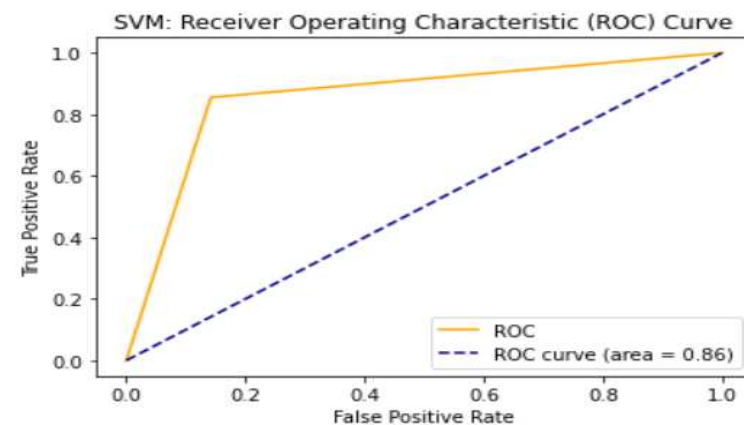


ROC curve for Random forest classifier and SVM

Random forest classifier

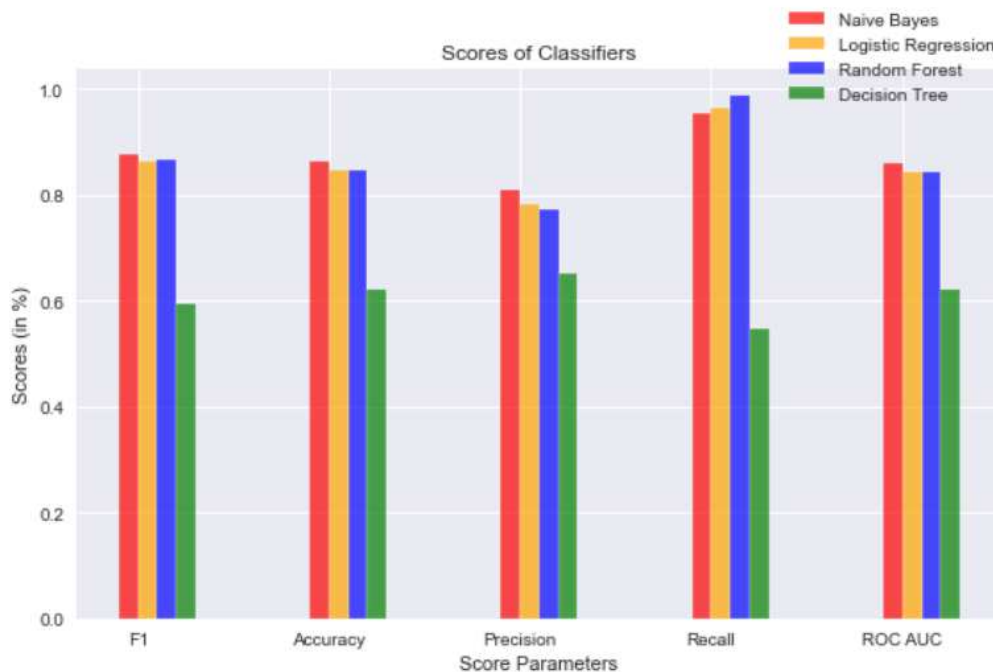


SVM



As per the ROC curve for customer review the AUC is showing the best fitted model as the Area under ROC is maximum for the model 1. Naïve Bayes 2. Logistic Regression 3. Decision Tree 4. Random Forest classifier and as per the AUC concept the model which is having the maximum area under roc is the best model.

For news headline, below is the plot for the metrices and the models. Here it has been shown the model wise metrices



## 5.4 Summary

In this study I have analyze the data and prepare the data for training the model also implement the data visualization technique. The data has been split into train and test dataset for both the experiment customer review and news headlines then different models has been train and test using the test data.

I have then analysis the model on the basis of the metrices accuracy and the AUC also confusion matrix.

And then find out the best fitted model.

With this experiment or the project any organization can find out the point of view of their customer regarding their products and the services. With news headlines can identify the news is positive or negative

In result I have compare the result of all this model and find out the best fitted model here the comparison of the different metrices has been done.



Comparison of past work and this study:

Past work

1. [https://www.researchgate.net/publication/348159352\\_Sentimental\\_Analysis\\_of\\_Amazon\\_Product\\_Reviews\\_Using\\_Machine\\_Learning\\_Approach](https://www.researchgate.net/publication/348159352_Sentimental_Analysis_of_Amazon_Product_Reviews_Using_Machine_Learning_Approach)  
dataset of 48500 records.

	Model	F1-Score	Recall	Accuracy	Precision
1	Multinomial Naïve Bayes	82.75%	81.33	84.72%	67.70%
2	Support Vector Machine	85.35%	84.59%	86.59%	86.51%

This study:

Naïve bayes:70000 records

SVM : 5000 records

Customer Review	F1 Score	Recall	Accuracy	Precision
Naïve Bayes	97.36%	98.09%	97.35%	96.64%
SVM	86.03%	85.70%	85.80%	86.37%

2. <https://www.diva-portal.org/smash/get/diva2:1241547/FULLTEXT01.pdf>

	Naïve Bayes
On reviews	90.16%
On summaries	92.72%

This study:

Naïve Bayes	Accuracy
Customer Review	97.35%
News Headlines	86.24%

## 6 Conclusion

In this study I have done the analysis of the Machine learning model that in such type of requirement of sentiment analysis which model is working best and then it has been achieved that the Decision tree for customer review is working best also naïve bayes and logistic regression and random forest classifier are giving the very good AUC.

With this study and experiment if anyone or any organization would like to check the sentiment of the customer reviews if its positive or negative then they can pass the data to the model built in this study and analyze the sentiment as sentiment analysis plays an important role in finding the opinion of the public it helps to improve the support to the customers and it's an automation

task which doesn't need the human intervention which directly save the time and money also. For individual sentiment analysis provides the real time result that helps to take any decision like for buying the products and investing in the stock as the data use in this study for news headlines is the stock news headlines data from Kaggle.

And the main and important thing is with this experiment of sentiment analysis we can get the sentiment of all over internet data at the center place. For example, the stock news is spread all over the internet on different sites, over the online news sites or the video channels so if this data is getting collected in one excel file and pass to the model built in this study for prediction this will give the result for all the data or the particular news is positive or negative.

Same in the case of customer reviews.

As discuss I have used Amazon review data but with this architecture of the study the no of client's requirements can be achieved for customer review sentiment as well as news headlines data.

This experiment can be use by multiple organization and the individuals.

## 6.1 Further Works

It will be very exciting to see how the deep learning techniques like LSTM bidirectional LSTM will work in this study.

It will be interesting to create the app for the sentiment analysis in which any organization or the customer will feed the data or upload the file of the reviews or the news headlines and then the output generated will be the positive or negative review of the particular product.

With this can add the only specific product related data and decide should buy the product or not

Also, for different areas of the organization like finance, culture, business, overall feedback like topic modelling can be implemented.

With this the review analysis website can be developed where any organization or the individual can measure the sentiment of the people over the internet at the central place.

## 7 References

- [1] J. Islam and Y. Zhang, Visual Sentiment Analysis for Social Images Using Transfer Learning Approach, 2016 IEEE Int. Conf. Big Data Cloud Comput. (BDCloud), Soc. Comput. Netw. (SocialCom), Sustain. Comput. Commun., pp. 124130, 2016.
- [2] [https://www.researchgate.net/publication/348159352\\_Sentimental\\_Analysis\\_of\\_Amazon\\_Product\\_Reviews\\_Using\\_Machine\\_Learning\\_Approach](https://www.researchgate.net/publication/348159352_Sentimental_Analysis_of_Amazon_Product_Reviews_Using_Machine_Learning_Approach)
- [3] Jason B. (2017) Gentle Introduction to Models for Sequence Prediction with RNNs
- [4] <<https://machinelearningmastery.com/models-sequence-prediction-recurrent-neural-networks/>> viewed 26 August 2021
- [5] Jason B. (2021) How to Choose an Activation Function for Deep Learning
- [6] <<https://machinelearningmastery.com/choose-an-activation-function-for-deep-learning/>> viewed 10 September 2021
- [7] Khan, Mudassir & Malviya, Aadarsh & Yadav, Suryakant. (2020). Big data approach of sentiment analysis of twitter data using k-mean clustering approach. 10. 6127-6134.
- [8] Kristain, B. 2018 understanding sentiment analysis: what it is and why it's used, viewed on 2 September 2021 < Sentiment Analysis: How Does It Work? Why Should We Use It? | Brandwatch>
- [9] Bhatt, R. and Gupta, P. (2019). Sentiment Analysis. Indian Journal of Science and Technology, 12(41), pp.1–6.
- [10] Chockalingam, N. (2018a). Simple and Effective Feature Based Sentiment Analysis on Product Reviews using Domain Specific Sentiment Scores. Polibits, 57, pp.39–43.
- [11] Chockalingam, N. (2018b). Simple and Effective Feature Based Sentiment Analysis on Product Reviews using Domain Specific Sentiment Scores. Polibits.
- [12] De, S.R. (2017). Sentiment analysis on product purchase through e commerce. International Journal of Scientific Research and Management.

- [13] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems* (pp. 3111-3119).
- [14] Miedema, F. (2018). Sentiment analysis with long short-term memory networks. *Vrije Universiteit Amsterdam*, 1.
- [15] <https://www.diva-portal.org/smash/get/diva2:1241547/FULLTEXT01.pdf>
- [16] *No online customer reviews means BIG problems* 2016, <<https://fanandfuel.com/no-online-customer-reviews-means-big-problems-2017> > viewed 10 August 2021
- [17] Ohana, B., & Tierney, B. (2009). Sentiment classification of reviews using SentiWordNet. *Proceedings of IT&T*.
- [18] The imbalanced-learn developers. (2021) Common pitfalls and recommended practices <[https://imbalanced-learn.org/stable/common\\_pitfalls.html](https://imbalanced-learn.org/stable/common_pitfalls.html)>
- [19] The university of Edinburgh (2019) < <https://www.ed.ac.uk/informatics/news-events/stories/2019/king-man-woman-queen-the-hidden-algebraic-struct> >
- [20] Vanishing gradient problem < [https://en.wikipedia.org/wiki/Vanishing\\_gradient\\_problem](https://en.wikipedia.org/wiki/Vanishing_gradient_problem)>
- [21] What is the Difference Between CNN and RNN? (2021) <<https://www.telusinternational.com/articles/difference-between-cnn-and-rnn>>
- [22] X. Ouyang, P. Zhou, C. H. Li, and L. Liu, Sentiment Analysis Using Convolutional Neural Network, Comput. Inf. Technol. Ubiquitous Comput. Commun. Dependable, Auton. Secur. Comput. Pervasive Intell. Comput. (CIT/IUCC/DASC/PICOM), 2015 IEEE Int. Conf., pp. 23592364, 2015
- [23]
- [24] Hamdan, H., Bellot, P. and Bechet, F. (2015). Sentiment Lexicon-Based Features for Sentiment Analysis in Short Text. *Research in Computing Science*, 90(1), pp.217–226.
- [25] Sentiment Analysis - A Review. (2015). *International Journal of Science and Research (IJSR)*, 4(12), pp.1842–1845.
- [26] YUE, G., DONG, Y., CHEN, H. and LAI, K. (2013). Online Textual Sentiment Analysis Technology and It's Applications. *Advances in Psychological Science*, 21(10), pp.1711–1719.
- [27] ZHAO, Y.-Y., QIN, B. and LIU, T. (2010). Sentiment Analysis. *Journal of Software*, 21(8), pp.1834–1848.
- [28]
- [29] Kurniasari, L., & Setyanto, A. (2020, February). Sentiment analysis using recurrent neural network. In *Journal of Physics: Conference Series* (Vol. 1471, No. 1, p. 012018). IOP Publishing.
- [30] KURNIASARI, L., & SETYANTO, A. (2020). Sentiment analysis using recurrent neural network-lstm in bahasa indonesia. *Journal of engineering science and technology*, 15(5), 3242-3256.
- [31] Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1), 1-167.
- [32] M. Day and C. Lee, Deep Learning for Financial Sentiment Analysis on Finance News Providers, no. 1, pp. 11271134, 2016.
- [33] Madhu R., (2015) Bi-directional RNN & Basics of LSTM and GRU <<https://medium.com/analytics-vidhya/bi-directional-rnn-basics-of-lstm-and-gru-e114aa4779bb>>
- [34] Peter D. Turney (2002) Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. *Computational Linguistics (ACL)*, Philadelphia,
- [35] Prabu Palanisamy, Vineet Yadav and Harsha Elchuri (2019) Serendio: Simple and Practical lexicon based approach to Sentiment Analysis -
- [36] Pranjal Srivastava (2017) *Essentials of Deep Learning : Introduction to Long Short Term Memory* pp 4 – 7
- [37] Polanyi L and Zaenen A (2006). Contextual valence shifters. In: Shanahan J, Qu Y and Wiebe J, eds., *Computing attitude and affect in text: Theory and applications*. Springer, pp. 1–10.
- [38] Ryan E., (2018) Want to Improve Your Glassdoor Rating? An External Agency May Be The Answer <<https://www.forbes.com/sites/ryanerskine/2018/02/14/want-to-improve-your-glassdoor-rating-an-external-agency-may-be-the-answer/?sh=1940074c2493> >

- [39] Sammi C., 2020, How to Use Social Media for Customer Service  
<<https://www.businessnewsdaily.com/5917-social-media-customer-service.html>>
- [40] Satyam K.,(2020) Overview of various Optimizers in Neural Networks  
<<https://towardsdatascience.com/overview-of-various-optimizers-in-neural-networks-17c1be2df6d5>>
- [41] Sepp Hochreiter, Jürgen Schmidhuber; Long Short-Term Memory. *Neural Comput* 1997; 9 (8): 1735–1780
- [42] Shiha, M., & Ayvaz, S. (2017). The effects of emoji in sentiment analysis. *Int. J. Comput. Electr. Eng.(IJCEE.)*, 9(1), 360-369
- [43] Singh, A. Recurrent Recursive Neural Networks for Sentiment Analysis.
- [44] Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A. Y., & Potts, C. (2013, October). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing* (pp. 1631-164

# Research Ethics Form

## Section 1

### Your details

---

**a) Researcher**

Ashwini Sawarkar  
(20211120)

**b) Title of Proposed Project**

Sentiment analysis on Customer review and News headlines

**c) Programme Title and Level of Study**

DATA SCIENCE

**d) Research Dates**

Start Date:

I confirm that I will not start my research before ethical approval has been given

I confirm that I will not start my research before ethical approval has been given

End Date:

2021-10-16

**e) Department**

First Subject

MATHEMATICS, COMPUTER SCIENCE AND ENGINEERING

First Supervisor Name

Dr Neil Buckley

**f) Professional Guidelines Referenced**

*No Guidelines Referenced*

## Section 2

### Who will be taking part in your research?

-----

**a) Will other people be taking part in your research?**

No human participants

## Section 3

### This section is for research which does not involve human participants

-----

**a) Does the research present a risk to you as the researcher?**

The research involves no risk to me as the researcher

**b) Summary of Research Project**

Please provide a brief but clear 200-word summary of the project

## **Section 4**

### **Consent Forms, Research Information Sheet and Additional Documentation**

---

**c) Research Information Sheet**

**Uploaded Files:**

No Files Uploaded

**d) Additional Documentation**

**Uploaded Files:**

No Files Uploaded